Emulator Reference

About This Book

HCL Z and I Emulator for Windows reference books are comprised of this *Emulator User's Reference* and an *Administrator's Guide and Reference*. This book is intended for users of Z and I Emulator for Windows.

Note:

1. *PC/3270* refers to the 3270 portion of the combined package.
2. *PC400* refers to the 5250 portion of the combined package.
3. *Workstation* refers to all supported personal computers.
4. *Windows®* refers to Windows® 7, Windows® 8/8.1, Windows® 10, Windows® Server 2008, and Windows® Server 2012. When information applies only to a specific operating system, this is indicated in the text.

Who Should Read This Book

This book is intended for the person who uses Z and I Emulator for Windows on a workstation to access hosts using 5250, 3270, or VT terminal emulation.

How to Use This Book

This book contains reference information that you might need to refer to when installing or operating Z and I Emulator for Windows.

Z and I Emulator for Windows is designed to use various communication adapters and to work with other workstation and host system software. Refer to the appropriate documentation for the products you use.

Command Syntax Symbols

Parentheses, brackets, ellipses, and slashes have the following meanings or uses:

( )

Parentheses enclose operands that govern the action of certain command options.

[ ]

Brackets indicate an optional command argument. If you do not use the optional item, the program selects a default.

...

Ellipsis after an argument indicates that you can repeat the preceding item any number of times.

/

For 3270, a slash must precede the Time Sharing Option Extensions (TSO/E) password. A slash must also precede parameters of DOS commands entered from the command line. For 5250, a slash must precede parameters of IBM® DOS commands entered from the command line.

\

A backslash is included as part of any directory name. An initial backslash indicates the first-level directory, and an additional backslash is inserted in the directory name to indicate another level.

All directives, operands, and other syntax can be typed in either uppercase or lowercase, unless otherwise indicated.

Where to Find More Information

The following sections discuss getting help when you are installing, configuring, or using Z and I Emulator for Windows.

Information Center

You can find documentation and links to other resources at the Z and I Emulator for Windows Information Center, at the following address:

https://help.hcltechsw.com/zie/ziewin/3.0/index.html

The Information Center contains reference material that is not found in this book, such as keyboard layouts and host code page tables.

The Z and I Emulator for Windows Information Center provides information in English.

Online Help

The help facility describes how to install, configure, and use Z and I Emulator for Windows. Online help is very extensive and includes information about every aspect of configuring and using Z and I Emulator for Windows. You can use Z and I Emulator for Windows online help just as you use the online help for Windows®.

Use help to obtain information about:

- Menu choices
- Operation procedures
- Operations in windows
- Meanings of the terms displayed in windows
- Causes of errors and the corresponding actions to take
- Mouse-based operations
- Operation without a mouse
- Detailed explanations of specific terms
- Further technical information about Z and I Emulator for Windows
- Detailed explanations of operator information area (OIA) messages

Z and I Emulator for Windows Library

The Z and I Emulator for Windows library includes the following publications:

- *Installation Guide*
- *Quick Beginnings*
- *Emulator User's Reference* (this document)
- *Administrator's Guide and Reference*
- *Emulator Programming*
- *Host Access Class Library*

In addition to the PDF documents, there are HTML documents provided with Z and I Emulator for Windows:

Quick Beginnings

The HTML form of *Quick Beginnings* contains the same information as the PDF version. The HTML files are installed automatically and can be accessed from the Help menus in the Session Manager and .WS session panels.

Related Publications

For information about local area networks (LANs), refer to the following publications:

- *IBM Local Area Network Technical Reference*
- *AS/400 Communications: Local Area Network (LAN) Guide Version 2*

Contacting HCL

This section lists ways you can reach HCL in case you encounter a problem or concern with Z and I Emulator for Windows. Depending on the nature of your problem or concern, we ask that you be prepared to provide the following information to allow us to serve you better.

- The environment in which the problem occurs:
  - Z and I Emulator for Windows configuration
    - Z and I Emulator for Windows version and manufacturing refresh level
    - The name of the workstation profile
  - Workstation configuration
    - The machine type and model, the system memory, the video adapter
    - The communication adapter you are using
    - Other adapters (especially communication adapters) installed
    - The printer type and model
    - Other devices installed, such as sound cards, modems, or fax machines
  - Software configuration

        - Windows® version and level
        - Communication and device-driver version and level
        - Other communication programs (such as Microsoft® Data Link Control) that are running and using resources
        - Printer driver version and level
    - Host configuration
        - The upstream host connection and configuration
    - FTP client configuration
        - The name of the FTP client configuration
        - The trace files
- Problem analysis information
    - Symptoms
    - Type of problem
    - OIA messages or error messages (if any)
    - Key factors related to the problem

If you have a technical problem, take the time to review and carry out the actions suggested here. Use your local support personnel before contacting HCL. You can also check the Hints and Tips at the Z and I Emulator for Windows support Web page for more information. Only persons with in-depth knowledge of the problem should contact HCL; therefore, support personnel should act as the interface with HCL.

Support Options

If you determine that you need to contact HCL, you can do any of the following:

- Access the Z and I Emulator for Windows Support page.

Part I. General Information

Chapter 1. Z and I Emulator for Windows Highlights

Z and I Emulator for Windows brings the power of personal networking to your workstation by providing a variety of connectivity options supporting local area network (LAN) and wide area network (WAN) environments. Whether you need host terminal emulation, client/server applications, or connectivity, Z and I Emulator for Windows offers a robust set of communications, networking, and administrative features.

Z and I Emulator for Windows is a full-function emulator package with an easy-to-use graphical interface, which includes many useful features such as file transfer and dynamic configuration, and emulator APIs including the Host Access Class Library.

Z and I Emulator for Windows provides the following functions:

- **zSeries™ Connections**

  LAN

  Telnet3270

  VT-over-Telnet (TCP/IP)

  COM port

  Telnet 3270

  VT-over-Telnet (TCP/IP)
- **iSeries™ Connections**

  LAN

  Telnet5250 over TCP/IP

  VT over Telnet
- **ASCII Emulator Connections**

  LAN

  VT over Telnet

  COM port

  VT over Telnet
- **Log Viewer**
  - View Message Log, Trace Log, and Merged Log files
  - Summary and Detail views
  - Set default Message Log size and location
  - Filter and search Log files
  - Message Log entries Help
- **Trace Capability**

    - 3270/5250 emulator data
    - Connectivity data, such as LAN
    - User services data, such as node initialization
- **Sample Programs**
    - Located in \Z and I Emulator for Windows\samples subdirectory
- **Installation and Configuration**
    - Partial installation option
    - Program sharing on a network server
    - Automatic detection of installed communication adapters
    - Dynamic change of communication configurations
    - Automatic Dial Facility)
    - Silent Installation
    - Verification of ASCII configuration
- **Host Session Function**
    - Up to 52 sessions
    - Variable screen size and automatic font scaling
    - Function settings (of the host code page, for example) for each session
- **Host Graphics Support**
    - Built-in vector graphics support for GDDM® and other graphics applications
- **File Transfer Function**
    - Easy operation through graphical user interface (GUI) windows
    - Batch transfer of multiple files
    - Concurrent file transfer through multiple sessions
    - Background file transfer
    - File transfer invocation by macro
    - VT File Transfer (XModem and YModem)
- **Edit (Cut and Paste) Function**

  You can use the clipboard to cut, copy, and paste a selected area. In addition, you can paste data in other applications, such as spreadsheet programs, that support the PasteLink function.
    - Support of spreadsheet data format (Sylk, BIFF3, Wk3 formats)
    - Copy Append
    - Paste Next
    - Paste to Trim Rectangle
    - Paste Stop at Protected Line
- **Graphical User Interface (GUI)**
    - Customizable 3D iconic tool bar
    - 3D-button hotspots
    - Pop-up keypad
    - Macro function, including record and play
    - VBScripts, including record and play
    - Keyboard-function setup and remapping
    - Mouse-button-function setup and remapping

  - Display setup (cursor type, graphics, sound, colors, for example)
  - Automatic font size adjustment or fixed font size
  - Window-appearance setup
  - Menu-bar customization
  - 3270 Light Pen emulation by using a mouse
  - Status bar with history
  - Page setup (Text and Graphics)
  - Revised Configuration Dialog
  - Online help
- **Print Function**
  - Printer session (for PC/3270: SCS, LU 3)
  - Graphics local print
  - Printing with the Windows printer drivers
  - Print function by printer definition table (PDT)
  - Multiple host-print functions in multiple sessions
  - PDF-to-PDT conversion tool
  - PC400 print function by OS/400® and i5/OS™ Host Print Transform (HPT)
  - PC400 printing supported by the iSeries™, eServer™ i5, and System i5™ Advanced Print Support Utility
- **Programming Interfaces**
  - 32-bit Emulator High-Level Language Application Programming Interface (EHLLAPI)
  - 32-bit Z and I Emulator for Windows API (PCSAPI)
  - 32-bit Automation Object API
- **PC400 Client Function**
  - Data transfer
  - Text Assist
  - Enhanced Programmable Terminal User Interface (ENPTUI)

Chapter 2. Problem Analysis

This chapter describes the information that will help you analyze problems with Z and I Emulator for Windows, and ways to report a problem to HCL. For detailed information about contacting HCL, refer to *Quick Beginnings*.

For information about Z and I Emulator for Windows and support, refer to the following Web site(s):

- The Z and I Emulator for Windows support page provides links to code fixes, tips, newsgroups, support options, and services. To view this page or to submit a software defect report, go to the following Internet address:

  https://hclpnpsupport.hcltech.com/csm

Z and I Emulator for Windows provides several utilities to help you with problem analysis. They can be invoked by clicking their icons from the **Programs → HCL Z and I Emulator for Windows → Administrative and PD Aids** subfolder on the Windows® **Start** menu.

The following sections describe these utilities and how to use them.

Log Viewer

The Z and I Emulator for Windows log viewer utility enables you to view, merge, sort, search, and filter information contained in message and trace logs. Use the log viewer during problem analysis to work with message and trace log entries. The default name of the message log output file is PCSMSG.MLG; its file extension must be .MLG. The file extension for trace logs must be .TLG.

To view message or trace logs:

1. From the Administrative and PD Aids subfolder, click **Log Viewer**; or, from an active session, click **Actions → Launch → Log Viewer**.
2. From the list of logged messages, double-click a message to display the message text.

For more information about log viewer functions, refer to *Administrator's Guide and Reference*.

Trace Facility

The Z and I Emulator for Windows trace facility enables you to log trace information for certain Z and I Emulator for Windows functions.

To start a trace, perform the following steps:

1. From the **Administrative and PD Aids** folder, click **Trace Facility**; or, from an active session, click **Actions → Launch → Trace Facility**. The trace status on the title bar displays the current state:

   Active

   Trace data is being collected by the trace facility.

Inactive

No trace data is being collected.

2. From the main dialog box, click **Set Up** to set the desired trace system parameters.
3. Click **OK** to return to the main trace dialog box.
4. From the main trace dialog box, select the type of data you want to trace from the **Function Name**, **Component Name**, and **Trace Option** list boxes.

   Function Name

   A specific set of Z and I Emulator for Windows features, such as 3270/5250 Emulator or User Services.

   Component Name

   The name of a specific part of a function, such as API data (for the 5250 Emulator function) or Node Initialization (for the User Services function).

   Trace Options

   The options associated with a particular component, such as EHLLAPI (for the API component) or API trace (for the Node Initialization component).
5. Start tracing data by clicking **Start**, or apply changes to the trace options by clicking **Apply**.
6. Run the operation that you want to trace.
7. Optionally, stop the trace by clicking **Stop**.
8. Save the trace data to your hard disk by clicking **Save**.
9. Click **Format** to specify a formatted trace file name and to format the trace data. The Information Bundler utility should be used immediately after the trace is complete to ensure that the correct information is gathered.

   Note: If you have changed the default path for the formatted trace file, the Information Bundler will not find the trace information. Copy the trace files to the system-class application data directory.
10. Click **OK**.
11. Click **Clear** to clear the trace buffer where you saved a trace.
12. Use the log viewer to view the formatted trace log.

Enhanced trace buffers

Additional CSTrace buffers are provided, so that all trace records are captured during heavy trace loads.

Information Bundler

The Z and I Emulator for Windows Information Bundler utility gathers system files, trace and log files, and registry information into a .ZIP file. This file can be sent to support personnel, using the Internet Service utility. The

Information Bundler should be executed immediately after the trace is complete to ensure that the correct information is gathered.

Start Information Bundler using one of the following methods:

- Click **Administrative and PD Aids → Information Bundler** from the Z and I Emulator for Windows program menu.
- In an active emulator session, click **Actions → Launch → Information Bundler** from the menu bar.

The X12345.ZIP file is created in the Z and I Emulator for Windows system-class application data directory. This file contains system and Z and I Emulator for Windows information. Refer to the installation documentation for the location of the system-class application data directory for each Windows® operating system.

NOTE : Information Bundler utility requires dotnet version 4.6.1 to work.

Chapter 3. Considerations for Using Z and I Emulator for Windows Sessions

This chapter contains general hints and tips for using Z and I Emulator for Windows sessions. Supplementary information other than the items described in this book are included in the Readme HTML file in the Z and I Emulator for Windows directory.

Usage Notes for Sessions in OLE Documents

Changing Fonts

If you are using an In-Place embedded session, then changing the font face name, switching between automatic sizing and fixed size, or changing the size for a fixed size font can result in an incorrect display on the screen. To correct the display, adjust the size of the session object window slightly.

Initial Selection of Font

The initial font selection for a embedded or linked session is determined by its Session ID (a letter A - Z or a - z) just like a regular session. Therefore, the initial font might change if other sessions are already active. Further, to prevent In-Place embedded sessions from having adverse effects on subsequent sessions, font changes made during use of In-Place embedded sessions are not saved.

WordPad

Z and I Emulator for Windows session objects created in Microsoft® WordPad documents by the drag-and-drop method cannot be used after the document has been saved and closed. You should only create Z and I Emulator for Windows objects in WordPad by using the **Insert → Object** menu option.

Some versions of Microsoft® Word and Microsoft® WordPad incorrectly save the state of embedded objects that are displayed as icons. If you open a document that contains a Z and I Emulator for Windows session object that was created to display as an icon, and the object is activated, then it might activate in-place, instead of as a separate window.

WordPro

If you attempt to open a link to a Z and I Emulator for Windows session in a Lotus® WordPro document, WordPro might give unpredictable results. You should only use embedded Z and I Emulator for Windows objects in WordPro documents. You can use the **Display as Icon** option if a separate window is desired.

Updating Linked Files

Files that are linked into Word 97 or Excel 97 do not update automatically. You must manually save the linked file before your edits are reflected in the container window.

Inactivity Timeout for Communication Links

The Inactivity Timeout automatically disconnects a link after it has been idle for a specified length of time. Its purpose is to avoid excessive charges on dial-up links, such as switched-line connections . Inactivity Timeout is not recommended for other types of connections.

To enable Inactivity Timeout, add the following statement to the PU section of your workstation profile (.WS file):

```
[PU]
InactiveTimeout=xxx
```

The value `xxx`, in the range 1 to 999, is the number of minutes a link remains connected when there is no activity over it. The default value, InactiveTimeout=0, disables Inactivity Timeout.

Note: The Inactivity Timeout function monitors only attention keys (that is, the Enter, Clear, and PFx keys). It is recommended that you set a comparably longer value for `xxx` if, for example, you expect to key in large amounts of data on the screen before pressing the Enter key.

Environment Considerations

The following are environmental considerations for Z and I Emulator for Windows.

Virtual Memory

If you receive a message stating that the system is low on virtual memory, increase the size of the virtual memory paging file. If you receive this message while trying to open new host sessions or starting a Z and I Emulator for Windows function such as File Transfer, increase the amount of virtual memory.

Refer to the operating system documentation for instructions on how to increase the size of the paging file.

Emulator Session Icons

Emulator session icons that were not migrated during installation of Z and I Emulator for Windows Version 3.0 will not function correctly if they were not created in the application data directory that was specified during installation. The icons can be updated by using the **File → Import** option from the Session Manager. This option will not copy the prior icons to the application data directory specified during installation; the icons must be moved manually

Disabling CDRA Tables

This release uses the standard IBM® CDRA translation tables when converting between ASCII and EBCDIC. From some code page pairs, the standard tables differ from those that Z and I Emulator for Windows has used in the past. For code pages that were supported in prior releases, you can configure Z and I Emulator for Windows to use the old tables. A switch is available in PCSWIN.INI to disable the use of CDRA tables. This switch is located in the Translate section and is named UseOnlyZIEWin. This switch takes a binary value and is defaulted to FALSE (except for Japan, Korea, Taiwan, and PRC where it defaults to TRUE). For the code pages that are new to Version 3.0, you must use the standard tables. Setting the switch will apply to all sessions, as well as data transfer and command line file transfer.

Chapter 4. Printing

You can use Z and I Emulator for Windows to print from display or printer sessions. This chapter provides details about printing and page setup options.

Setting Up the Printer

This section describes how to set up your printer with either a Windows® printer driver or a PDT file.

For an explanation of how to customize a PDT for PC/3270, see PDT Files (3270 and 5250) on page 24. For an explanation of how to customize a PDT for PC400, refer to *Administrator's Guide and Reference*.

For information about print processing for PC400, see Print Processing on page 95. For information about print processing for PC/3270, see Print Processing on page 53.

You can map a key sequence to bring up the Printer Setup dialog. There is no default key combination for this function. See Default Key Function Assignments on page 29 for more information about keyboard functions.

Defining a Printer for a Session

To choose or change the Windows® printer driver to be used, follow these steps:

1. Click **File → Printer Setup** in the session window.

   The Printer Setup window lists the available printers.
2. Select a printer from the list box. If desired, select the check box **Show this dialog before every print**.

   Note: The **Default** selection causes the current Windows® default printer to be used.
3. Confirm that **Use PDT** is not selected.

Page Setup Parameters

You can set Page Setup parameters, such as the maximum number of lines per page, the maximum number of columns, and fonts. These parameters are initially set to the defaults. Use this function to change specific control items.

Note: When a PDT file is used, this function cannot be used.

To set or change setup items:

1. Select **File → Page Setup** from the menu bar of the session window.

   The Page Setup window opens.

2. Select the tab that contains the parameters you want to change.

   Note: To switch from the current settings back to the defaults, select **Default**.

3. When all the items have been set, click **OK** or **Apply**.

Text Parameters

You can set the following general parameters for 3270, 5250, and VT display sessions.

Table 1. Page Setup Parameters — Text Tab

Parameter	Description
CPI	Specifies the number of characters to be printed per inch. If CPI was not set previously, a CPI value suitable for the font selected from the **Font** list box is assumed as the default.
LPI	Specifies the number of lines to be printed per inch. If LPI was not set previously, an LPI value suitable for the font selected from the Font list box is assumed as the default.
Maximum Number of Lines per Page	Specifies the maximum number of lines per page. A number in the range 1–255 can be specified. The default is *66*. When you set this to a value other than the default, Z and I Emulator for Windows uses it to scale the LPI and font to the best fit for the page.
Maximum Number of Characters per Line	Specifies the maximum number of characters per line. A number in the range 1–255 can be specified. The default is *132*. When you set this to a value other than the default, Z and I Emulator for Windows uses it to scale the CPI and font to the best fit for the page.
Font	Lists the fonts that can be used with the currently selected printer. The fonts in brackets [ ] are device fonts specific to the printer driver. The other fonts are graphics display interface (GDI) fonts printed in bit map formats.
Margins	This option maps the text on the screen to the printed page size. This option is disabled by default. It is available for the following sessions: • 3270 display and printer • VT display, including Printer Controller Mode
Top or Left Margin	Specify the decimal number (either inches or centimeters, depending upon your location), which represents the offset that will be reserved as the margin. **Note:** The value entered must represent the distance from the paper's edge. However, most Print Drivers enforce a nonprintable border (area). The default (original value shown) represents the Print Driver's enforced margin. Your input value cannot be less than that value; if it is, the Driver's value is substituted for

Table 1. Page Setup Parameters — Text Tab (continued)

Parameter	Description
	your value. Some printers have an additional nonprintable border not represented in the Driver's value.
Use best fit scaling	This option maps the text on the screen to the printed page size. This option is disabled by default. It is available for the following sessions: • 3270 display and printer • VT display, including Printer Controller Mode

Text Options Parameters

You can set the following parameters by selecting the **Text Options** tab.

Table 2. Page Setup Parameters — Text Options Tab

Group	Options	Setting
Print Options These options are not available for the printer session.	**Suppress Null Lines**	Determines whether to delete lines containing only null or non-printable characters (null or non-printable field characters, and field attributes) or to print them as null lines. **Selected** Null lines are not printed. **Not selected** Null lines are printed. This option can be used when: • The **Print Screen** command is used • The combination of bits 2 and 3 of the WCC is not 00
	Print Nulls as spaces	Determines whether to print NULL control codes as null characters. **Selected** The codes are printed as blanks.

Table 2. Page Setup Parameters — Text Options Tab

(continued)

Group	Options	Setting
		Not selected The codes are treated as null characters. This option can be used when: • The **Print Screen** command is used
	Replace FF by LF	Select this option to replace a form feed by the number of lines entered in the edit box. The default is unchecked. **Note:** This option is only available on TN3270/TN3270E and TN5250.

Page Header and Footer Parameters

You create your own header and footer, and save up to five header and five footer configurations. Apply a saved header or footer by selecting it from the drop-down list.

Note:

1. A custom header or footer is associated with the specific session. A newly configured session will not have a header or footer.
2. If BestFit is enabled, the header and footer will be truncated at the Maximum Print Position (MPP), as determined by the BestFit parameters. You can allow multiple lines to prevent truncation (from the **Advanced** options).

To add items to a custom header or footer, do the following:

1. Select the desired alignment for the item (**Left**, **Center**, or **Right**).
2. Double-click on the item in the **Choices** box.

   The item is added to the alignment box.

You can manually reorder the items in an alignment box. Remove an item by manually deleting it from the box.

You can set the following parameters by selecting the **Header and Footer** tab.

Table 3. Page Setup Parameters — Header and Footer Tab

Group	Category	Parameter
Customize Header **Customize Footer**	**General**	The following information can be added to a header or footer: • **Date** • **New Line** • **Page Number** • **PC Name** • **Time** • **PC User Name**
	Host Information	The following host details can be added to a header or footer: • **3270 Application Name** (3270 sessions only) • **Host Name** • **LU Name** • **Workstation ID** (5250 sessions only)
	Session Information	The following session details can be added to a header or footer: • **Short ID** • **Short Name**
Advanced Options	**Multiple Lines**	The following customization options are available: • **Allow multiple lines in header** • **Allow multiple lines in footer** If the header or footer does not fit on a single line, then it will be truncated at the Maximum Print Position. Select this option to allow multiple lines on the header or footer and prevent truncation.
	Page Number	The **Always start from** parameter specifies the starting value for the page number to be included in the header or footer. By default, the page number begins at 1.

Graphics Parameters (3270)

From a Z and I Emulator for Windows 3270 session, you can set additional parameters by selecting the **Graphics** tab.

Table 4. Page Setup Parameters — Graphics Tab (3270)

Parameter	Description
Scaling	By default, the screen size (display resolution) is mapped to the printed page size (printer resolution)—this is called **BestFit**. It is done automatically if you change either resolution (including changing printers). The **/2**, **/3**, and **/4** values reduce the printed page size.
Black-on-White	Determines how the black pixels on the screen are printed. **Yes** Black pixels are printed as white pixels. Pixels other than black are printed as black pixels when you use a monochrome printer. When you use a color printer, they are printed in the same color as on the screen. **No** Black pixels are printed in black. Pixels other than black are printed as white pixels when you use a monochrome printer. When you use a color printer, they are printed in the same color as on the screen.

Orientation Parameters (5250)

When you use a PC400 printer session, you can set the following additional parameters by clicking the **Orientation** tab.

Table 5. Page Setup Parameters — Orientation Tab (PC400 Printer Session)

Group	Parameter	Description
Margins These margin settings are used only if **Use best fit scaling** is selected.	Top Margin	Bottom Margin is assumed to be equal to Top Margin.
	Left Margin	Right Margin is assumed to be equal to Left Margin.
Page Orientation Changes the default page orientation to specify how to print a document on the workstation printer. If the orientation is explicitly set by the iSeries™, eServer™ i5, or Sys-	Use automatic page orientation	If selected and the host does not explicitly set the orientation, the best orientation based on the host specified CPI, LPI, and page size will be used. If not selected and the host does not explicitly set the orientation the following drawer orientation will be used.

Table 5. Page Setup Parameters — Orientation Tab (PC400 Printer Session)

(continued)

Group	Parameter	Description
tem i5™ page setup code, the explicit orientation is used.		
Drawer 1 orientation The default page orientation for the paper from drawer 1.	**Computer output reduction**	The document is printed in landscape. The font, pitch, and margins are set to appropriate values to fit on a page.
	Portrait	The document is printed in portrait.
	Landscape	The document is printed in landscape.
Drawer 2 orientation The default page orientation for the paper from drawer 2.	**Computer output reduction**	The document is printed in landscape. The font, pitch, and margins are set to appropriate values to fit on a page
	Portrait	The document is printed in portrait.
	Landscape	The document is printed in landscape.

When you use a PC400 printer session, you can set the following additional parameters by clicking the **Form Settings** tab. This option is available only when the printer and its driver support the change-source function.

Table 6. Page Setup Parameters — Form Settings Tab (PC400 Printer Session Only)

Parameter	Description
Form Settings	Specifies the form that should be selected when an application program specifies one of the following paper sources: • Drawer-one form • Drawer-two form • Envelope-hopper form Before using this function, you must configure the paper trays and forms in the printer-driver setup.

Table 7. Page Setup Parameters — Advanced Options Tab (PC400 Printer Session Only)

Option	Item to be set
Printer Font Code Page	Represents the code page being used for printing and displaying on the workstation.
No CR between fields	Represents not sending a CR when printing other fields on the same line.
Print bold as normal	Represents printing bold characters as not bold.
Display print status dialog	Represents showing a dialog window that will display showing printer status.
Use raster fonts	Represents allowing bitmap fonts for display and printing.

Display Sessions (3270 and 5250)

From display sessions, you can print all (**Print Screen**) or part (**Trim Print**) of the screen of your session window on a workstation printer. **Trim Print** is not available for PC400 sessions. For more information, refer to *Quick Beginnings* or the online help.

From a 3270 display session, you can also use the ZipPrint utility to print PROFS® notes, calendars, documents, CMS files, XEDIT workspaces, and host-session screens. See the online help for more information.

Print Screen Collection

The Print Screen Collection functions are available for 3270 and 5250 display sessions. You can capture all or part of the screen and add it to a collection of screen captures, and then print all the collected screen captures at the same time.

Collect Screens

Using the **File → Print Screen Collection → Collect Screen** feature, you can add a capture of all or part of the screen to a collection of captures.

To capture part of a screen, take the following steps:

1. Use the marking rectangle to mark an area of the screen.
2. Click **File → Print Screen Collection → Collect Screen** in the usual way. Z and I Emulator for Windows captures the entire screen image (including the white marking rectangle) and adds the screen image to the list of collected screen captures in the usual way, so that you can see the context of the area enclosed by the white marking rectangle.
3. Print the screen image. Z and I Emulator for Windows prints only the area that lies inside the white marking rectangle.

Note: The **Collect Screens** feature works independently of the normal **Print Screen** function. You can still use **Print Screen** to print an individual screen, while collecting multiple screens.

Print and Purge Collection

Using the **File → Print Screen Collection → Print and Purge Collection** feature, you can send the collected print screens to the printer. A status bar message indicates how many screens have been printed. The current Page Setup settings are applied to the printed screens. All the collected screens are purged.

Print and Keep Collection

Using the **File → Print Screen Collection → Print and Keep Collection** feature, you can send the collected print screens to the printer. A status bar message indicates how many screens have been printed. The current Page Setup settings are applied to the printed screens. All the collected screens are available for reprint.

Process Collection

Using the **File** → **Print Screen Collection** → **Process Collection** feature, you can preview the collected screens and select collected screens to be printed or purged.

On the **Process Print Screen Collection** window, you can perform the following tasks:

- Review the screens that have been collected.
- Select the check box next to one or more screens for processing.
- Print or delete selected screens.
- Scroll through the collected screens using the scroll bar.
- Click a collected screen to view a larger version of the screen.

Purge Collection

All the collected screens can be deleted without printing by clicking **File** → **Print Screen Collection** → **Purge Collection**. A confirmation message will be displayed. Click **Yes** to purge the collected screens.

Print Collection on Exit

The **File** → **Print Screen Collection** → **Print Collection on Exit** option ensures that the collected screens are printed before you close or disconnect the session. This option is enabled by default. To end the session without printing the collected screen, clear the **Print Collection on Exit** option. All the collected screens are then deleted when you close or disconnect the session.

When this option is disabled and you disconnect the session, a confirmation message will be displayed. Click **Yes** to purge the collected screens on exit.

You can add the **Collect Screens**, **Print and Purge Collection**, **Print and Keep Collection**, **Process Collection**, **Purge Collection**, and **Print Collection on Exit** functions to the toolbar, a popup keypad, a custom keyboard map, or a mouse customization, using the **Edit** → **Preferences** menu in the session window. The settings in the Page Setup dialog are used (shared with the normal Print Screen function).

In PDT mode, there is an option available for printing more than one screen on a page. Refer to *Administrator's Guide and Reference* for more information.

Replace FF with LF in GDI Print Mode

In Multiple Print Screen functionality, 3270/5250 host screens can be collected and then released to the physical printer. Each print screen that is collected is printed on a separate page, like individual print screens. However, in PDT mode using the BEL command, the Form Feeds (between two screens) can be converted to specified number of Line Feeds that is defined in the PDF/PDT file.

With this feature, the same functionality has been extended to GDI print mode, using the following workstation profile keyword:

```
[Printers]
ReplaceFFbyLF=<Byte value>
```

The possible byte values are as follows:

Byte Value	Action
00	No LF between screens
01 to 0xFE	LFs between screens
0xFF	FF after each screen
No keyword	Default (FF after each screen)

Printer Sessions (3270 and 5250)

From printer sessions, you can direct printing from a zSeries™, iSeries™, eServer™ i5, or System i5™ to a workstation printer.

Note: When you use a host application which prints to your workstation's LPT1, you must first select the printer in the **Printer Setup** dialog of the **File** menu.

Configure a printer session to designate a workstation printer as a system printer that will use either a Windows® printer driver or a printer definition table (PDT) provided with Z and I Emulator for Windows.

- Use Windows® printer drivers for Z and I Emulator for Windows to print files based on printer setup parameters, such as scaling, duplex options, and page orientation, that you define in **Printer Setup**.
- Use PDT files for Z and I Emulator for Windows to print files based on page setup information, such as control codes and the printer output format, defined in the PDT. You can customize PDTs to define your own controls, by editing the corresponding printer definition file (PDF) and converting it to a PDT.

PDT Files (3270 and 5250)

PDTs (printer definition tables) are compiled from PDFs (printer definition files). PDFs contain printer commands that must be understood and supported by your printer.

The following are the basic printer languages:

PCL

Printer Control Language (Hewlett-Packard)

PPDS

ProPrinter Data Stream (IBM®)

ESC/P

Printer Control Language (Epson)

POSTSCRIPT

(No PDFs for this language)

Many printers support two or more of these languages. Most print drivers use a PJL (Printer Job Language) to switch between languages and to perform other job control functions, such as setting the number of copies.

You do not need a PDF for each different printer model; with the increasing number of models, PDFs are named for the printer language, not the printer model.

Older SBCS PDFs and PDTs are not shipped with Z and I Emulator for Windows, but are available at the product Web site. If you already have modified PDFs, any PDF and PDTs other than those in Table 8: Old Printer Definition Files on page 25 are retained during an install.

The End_Job statement in a PDF contains the printer commands that are sent to the printer at the end of each print job. If the End_Job contains a character defined as form feed (`FFF` in the PDT), a form feed (FF) is sent to the printer. It is not needed if the host application ends the job with the FF, as is commonly done. Some print drivers add the FF if needed, and most print drivers ignore extra FFs. So the FFF usually is protection against the host application not using a FF, and usually causes no problem. However, if you get an extra blank page, remove the FFF.

Table 8. Old Printer Definition Files

Printer Definition File (PDF) Name	Remarks
`ibm5577a`	(No FFF)
`ibm5577b`	(No FFF)
`lbp4`	

The supplied basic_ascii PDF does not contain any printer commands, which results in only ASCII text being sent to a printer or file. An accompanying PDT is also shipped. This PDF is for SBCS only.

PFT Migration

You can migrate a PC Support/400 Workstation Feature Printer Function Table (PFT) to a PDF for PC400. Refer to *Administrator's Guide and Reference* for more information.

Using PDT Files

To use a PDT file:

1. Click **File → Printer Setup** from the menu bar of the session window.

   The Printer Setup window opens.
2. Click the printer to be used from the list box.
3. Click **Setup**; specify the paper size.
4. Click **OK**.
5. Select the **Use PDT** check box, then click **Select PDT**.

   The Select PDT file window opens.

6. Do one of the following:
    - To use an existing PDT, select the PDT file to be used; then click **OK**.
    - To use a PDF that you have modified, you must first convert it to a PDT. To do so:
        a. Click **Convert PDF**.
        b. Select the PDF file to be converted from the list, then click **Convert**.

           The window displays the result of the conversion. If there are any errors during the conversion, they are listed in the window.
        c. When you select **Save List**, the window list is saved in *.LST file in the PDFPDT subdirectory.

           To close without saving the list, click **Close**.

           After the file is converted, control returns to the Select PDT file window and the converted PDT file appears in the list.
        d. Select the PDT file; then click **OK**.
7. Click **OK** in the Printer Setup window.

Windows print driver for VT host printing

You can use the Windows print driver for VT host printing. This functionality adds to the existing PDT printing capability.

Collecting Print Jobs (5250 Printer Session)

You can collect 5250 print jobs and print them as a single job or in a group. The collected print jobs are stored in a .SCS file.

Note: This functionality is not supported in Host Print Transform mode.

You can set the following .WS profile keywords to specify the path and file name for the .SCS file.

```
[Printers]
SCSFile=<filename>.scs
SCSPath=<local path>
```

The functions associated with this feature are listed below. The functions can be mapped to the keyboard, popup keypad, mouse button, or toolbar button.

- **Collect Mode**

  When Collect Mode has been started, print jobs that have been sent are saved in the .SCS file. They are not printed immediately.
- **Print Collection**

  The print jobs that have been saved are sent to the printer as a single job.
- **Purge Collection**

  The collected print jobs are deleted.

Refer to the online help for details about mapping the functions.
The CombineJobs profile keyword enables you to collect the jobs for printing, while maintaining them as individual jobs (instead of one job in the .SCS file). Specify the .WS keyword as follows:

```
[Printers]
CombineJobs=N
```

If you set CombineJobs to **N**, the Print Collection function sends the separate, collected jobs to the printer. While in Collect Mode, if the keyword is set to **Y** or is not specified, the print jobs are combined as a single job in the .SCS file.

Printing to Disk

If you are using a PDT, you can save a host print-job or the contents of the session window (Print Screen) to a workstation file instead of printing it.

Two types of Print-to-Disk function are provided by Z and I Emulator for Windows:

Print-to-Disk Append

Appends multiple host print jobs or print screen jobs to a single workstation file.

Print-to-Disk Separate

Saves each host-print job or screen to a separate workstation file. You can specify the file name, but the extension is automatically assigned as a decimal number from 000 to 999. If you delete a file, its number will be re-used. When all 999 numbers have been used, the extension is automatically assigned a decimal number from 1000 to 9999.

Note:

1. Print-to-Disk is not available for the Print-Graphics function.
2. Print-to-Disk can be used only when you use a printer definition table (PDT) file.

To set up Print-to-Disk:

1. Click **File → Printer Setup** from the menu bar in the session window.

   The Printer Setup window lists the supported printers.
2. Select **Print to Disk Append** or **Print to Disk Separate** from the list box.
3. Click **Select PDT**.

   The Select PDT file window opens.
4. Select a PDT file from the list; then click **OK**.

   The Printer Setup window reopens.
5. Click **Setup**.

   The Select Print-to-Disk File window opens.

6. Specify a file name, drive, and path; then click **OK**.

   Note: If you specify the name of an existing file, subsequent print jobs are appended to the data in the original file in the case of Print to Disk Append.

Workstation Profile Parameter for Code Page

Occasionally a font does not support the desired code page. The wrong characters may be printed within the specific character set (Latin 2, for example). Z and I Emulator for Windows has a workstation profile parameter that allows the program to use a different code page that is supported by the desired font.

You can use the **PrinterFontCodePage** parameter if the following conditions are met:

- You can specify the printer font code page with which the desired font is encoded.
- Z and I Emulator for Windows provides the translation table for the host code page and the printer font code page.

However, because some Z and I Emulator for Windows releases might require manual adjustment of the workstation profile, try using different fonts before altering the .ws file. Fonts are listed in the Z and I Emulator for Windows Page Setup panel for all display sessions and 3270 host print sessions. For 5250 print sessions, the PCSPD.DAT file can be manually changed to control the fonts used. The Courier New font should support most languages and corresponding code pages.

To edit the .ws file, you must change the PrinterFontCodePage parameter to the value of the supported code page you wish to use. This option must be put in the [Printers] section, and is case-sensitive. See the following example for the proper parameter syntax. The parameter does not need to be placed immediately after the [Printers] section label.

```
[Printers]
PrinterFontCodePage=852
```

In this case, the desired font is encoded with code page 852. Z and I Emulator for Windows uses a different, existing translation table to translate data from EBCDIC to 852, versus using the standard Windows® code page.

This option is on the Page Setup panel for Z and I Emulator for Windows 5250 print sessions.

Chapter 5. Key Functions and Keyboard Setup

This chapter contains information about keyboard setup and customizing mapped key functions.

Default Key Function Assignments

This section lists the functions assigned, by default, to each key on your keyboard.

For more information about each function, refer to the **Keyboard** choice on the **Help** menu.

You can change the default key assignments to the following default function tables, by selecting **Keyboard Setup** from the **Assist** menu.

When the Keyboard Setup window opens, select one of the following choices:

- 3270 for a 3270 keyboard layout
- 5250 for a 5250 keyboard layout
- 3270+5250 for a combined keyboard layout
- VT for a DEC VT220 keyboard layout

Z and I Emulator for Windows includes two .KMP keyboard map files that map the standard Win32 hotkeys for Cut, Copy, and Paste. You can use these keyboard map files or add the key values to an existing map file. See Win32 Cut, Copy, and Paste Hotkeys on page 38 for more information.

Setting the 3270 Keyboard Layout Default

To make the 3270 keyboard layout defaults available, do the following:

1. Click **Preferences → Keyboard** from the **Edit** menu. The Keyboard dialog box is displayed.
2. Select the **Default** radio button next to Current Keyboard.
3. Click **OK**.

Default Key Functions for a 3270 Layout

Table 9: Default Key Functions for a 3270 Layout on page 29 shows the default key functions for PC/3270. The key used is the same for all the supported keyboard types.

Table 9. Default Key Functions for a 3270 Layout

Function of Key	Key
APL	Ctrl+F8
Attention	Esc
Alternate Cursor	Alt+F11
Backspace	← (Backspace)
Back Tab	Shift+→|

Table 9. Default Key Functions for a 3270 Layout

(continued)

Function of Key	Key
Back Tab Word	Alt+←
Break	Break
Change Format Toggle	Alt+F3
Change Screen	Ctrl+PageUp
Clear	Pause
Cursor Blink	Ctrl+F10
Cursor Down	↓ or 2(pad)
Cursor Left	← or 4(pad)
Cursor Right	→ or 6(pad)
Cursor Select	Ctrl+F9
Cursor Up	↑ or 8(pad)
Delete Character	Delete or .(pad)
Delete Word	Ctrl+Delete or Ctrl+.(pad)
Document Mode Toggle	Alt+F1
Dup	Shift+Insert[2]
Edit Copy	Ctrl+Insert
Edit Cut	Shift+Delete
Edit Paste	Shift+PageDown or Ctrl+Shift+Insert
Edit Undo	Alt+◄— (Backspace)
End Field	Pad End
Enter/Control	Shift+Ctrl
Erase EOF	End[2]
Erase Field	Shift+End[2]
Erase Input	Alt+End[2]
Fast Cursor Down	Alt+↓ or Alt+2(pad)
Fast Cursor Up	Alt+↑ or Alt+8(pad)
Field Mark	Shift+Home[2]
Graphic Cursor	Alt+F12
Highlighting Field Inherit	Alt+3(pad)
Highlighting Reverse	Alt+*(pad)
Highlighting Underscore	Alt+6(pad)

Table 9. Default Key Functions for a 3270 Layout

(continued)

Function of Key	Key
Home	Home or 7(pad)
Insert	Insert or 0(pad)
Jump Next	Alt+PageUp
Mark Down	Shift+↓
Mark Left	Shift+←
Mark Right	Shift+→
Mark Up	Shift+↑
Move Mark Down	Ctrl+↓ or Ctrl+2(pad)
Move Mark Left	Ctrl+← or Ctrl+4(pad)
Move Mark Right	Ctrl+→ or Ctrl+6(pad)
Move Mark Up	Ctrl+↑ or Ctrl+8(pad)
PA1	Alt+Insert[2]
PA2	Alt+Home[2]
PA3	Shift+PageUp[2]
Pause	Ctrl+F7
PF1 to PF12	F1 to F12
PF13 to PF24	Shift+F1 to F12
Play	Ctrl+F6
Print (Local Copy)	Not assigned
Quit (Device Cancel)	Alt+Left Ctrl
Record	Ctrl+F5
Reset/Control	Left Ctrl
Response Time Monitor	Ctrl+F11
Rule	Ctrl+Home
Sys Request	Shift+Esc
Tab Field	→| or Shift+→|(pad)
Tab Word	Alt+→
Test	Ctrl+PageDown
Word Wrap Toggle	Alt+F2

[2] Indicates the key on the main keyboard.

Table 9. Default Key Functions for a 3270 Layout

(continued)

Function of Key	Key
(*pad*) Indicates a key on the numeric keypad.	
Note: The Enhanced keyboard has some duplicated keys. The functions of the duplicated keys are the same except when you specify a single key. For example, Del means any Delete key, whereas Pad Del specifies only the Delete key on the numeric keypad.	

Setting the 5250 Keyboard Layout Default

To make the 5250 keyboard layout defaults available, do the following:

1. Select **Preferences → Keyboard** from the **Edit** menu. The Keyboard dialog box is displayed.
2. Select the **Default** radio button next to Current Keyboard.
3. Click **OK**.

Default Key Functions for a 5250 Layout

Table 10: Default Key Functions for a 5250 Layout on page 32 shows the default key functions for iSeries™, eServer™ i5, or System i5™. The key used is the same for all the supported keyboard types.

Note:

1. If you use iSeries™ from the combined package, see Default Key Functions for the Combined Package on page 35.
2. The default key functions for a 5250 layout are not available by default. To make these functions available, perform the procedures in Setting the 5250 Keyboard Layout Default on page 32.

Table 10. Default Key Functions for a 5250 Layout

Function of Key	Key
Alternate Cursor	Ctrl+F11
Attention	Esc
Backspace	← (Backspace)
Backtab	Shift+→|
Backtab Word	Alt+←

Table 10. Default Key Functions for a 5250 Layout

(continued)

Function of Key	**Key**
Begin Bold*	Ctrl+B
Begin of line*	Ctrl+4(pad)
Begin Underscore*	Ctrl+U
Bottom of Page*	Ctrl+2(pad)
Carrier Return	Ctrl+Enter or Ctrl+-(pad) or Ctrl++(pad)
Center Text*	Ctrl+C
Clear	Pause
Cursor Blink	Ctrl+F10
Cursor Down	↓ or 2(pad)
Cursor Left	← or 4(pad)
Cursor Right	→ or 6(pad)
Cursor Up	↑ or 8(pad)
Delete Character	Delete or .(pad)
Delete Word	Ctrl+Delete or Ctrl+.(pad)
Display Text Code	Alt+Insert
Dup	Shift+Insert
Edit Copy	Ctrl+Insert
Edit Cut	Shift+Delete
Edit Paste	Shift+PageDown or Ctrl+Shift+Insert
Edit Undo	Alt+◄— (Backspace)
End Bold/Underscore*	Ctrl+J
End of line*	Ctrl+6(pad)
End of page*	Ctrl+P
Enter/Control	Right Ctrl
Erase EOF	End or 1(pad)
Erase Input	Alt+End
Fast Cursor Down	Alt+↓ or Alt+2(pad)
Fast Cursor Up	Alt+↑ or Alt+8(pad)
Field Exit	Enter(pad) or ↵ (Enter)

Table 10. Default Key Functions for a 5250 Layout

(continued)

Function of Key	**Key**
Field Mark	Shift+Home
Field Minus (-)	-(pad)
Field Plus (+)	+(pad)
Half Index Down*	Ctrl+H
Half Index Up*	Ctrl+Y
Help	Alt+F1
Home	Home or 7(pad)
Host Print	Ctrl+Pause
Insert	Insert or 0(pad)
Insert Symbol*	Ctrl+A
Jump Next	Alt+PageUp
Mark Down	Shift+↓
Mark Left	Shift+←
Mark Right	Shift+→
Mark Up	Shift+↑
Move Mark Down	Ctrl+↓
Move Mark Left	Ctrl+←
Move Mark Right	Ctrl+→
Move Mark Up	Ctrl+↑
Next Column*	Ctrl+D
Next Stop*	Ctrl+N
Pause	Ctrl+F7
PF1 to PF12	F1 to F12
PF13 to PF24	Shift+F1 to F12
Play	Ctrl+F6
Quit	Alt+Left Ctrl
Record	Ctrl+F5
Required Backspace	Ctrl+◄— (Backspace)
Required Space*	Ctrl+Space
Required Tab*	Ctrl+—►|
Reset/Control	Left Ctrl
Roll Down	9(pad) or PageUp
Roll Up	3(pad) or PageDown
Rule	Ctrl+Home
Stop Code*	Ctrl+S

Table 10. Default Key Functions for a 5250 Layout

(continued)

Function of Key	**Key**
System Request	Shift+Esc
Tab Field	→|
Tab Word	Alt+→
Test Request	Alt+Pause
Top of Page*	Ctrl+8(pad)
Word Underscore*	Ctrl+W

(*pad*)

Indicates a key on the numeric keypad.

*****

Indicates a Text Assist Key (SBCS only).

Default Key Functions for the Combined Package

Table 11: Default Key Functions for the Combined Package on page 35 shows the default key functions for the combined package. The key used is the same for all the supported keyboard types.

When you use the 3270+5250 keyboard layout, the key definitions for the 3270 and 5250 layouts are combined with those listed here.

Table 11. Default Key Functions for the Combined Package

Function of Key	**Key**
Change Screen	Not assigned
Character Advance	Shift+BackSpace
Help	Not assigned
Host Print	Not assigned
PA3	Not assigned
Roll Down	PageUp
Roll Up	PageDown
Printer Setup	Not assigned

Setting the VT Keyboard Layout Default

To make the VT keyboard layout defaults available, do the following:

1. Click **Preferences → Keyboard** from the **Edit** menu. The Keyboard dialog box is displayed.
2. Select the **Default** radio button next to Current Keyboard.
3. Click **OK**.

Default Key Functions for the VT Emulator Layout

Table 12: Default Key Functions for a VT Emulator Layout on page 36 shows the default key functions for VT220, VT100 and VT52. The key used is the same for all the supported keyboard types. The VT emulator keyboard gets selected as the default only when the VT Component is selected in the installation path.

Table 12. Default Key Functions for a VT Emulator Layout

Function of Key	Key
Backspace	← (Backspace)
Break	Ctrl+Pause
CAN	Ctrl+← (Backspace)
Cursor Down	↓ or 2(pad)
Cursor Left	← or 4(pad)
Cursor Right	→ or 6(pad)
Cursor Up	↑ or 8(pad)
Edit Copy	Ctrl+Insert
Edit Cut	Shift+Delete
Edit Paste	Shift+PageDown or Ctrl+Shift+Insert
Edit Undo	Alt+← (Backspace)
ESC	ESC
Jump Next	Alt+PageUp
New Line	↵ (Enter)
Mark Down	Shift+↓
Mark Left	Shift+←
Mark Right	Shift+→
Mark Up	Shift+↑
Move Mark Down	Ctrl+↓ or Ctrl+2(pad)
Move Mark Left	Ctrl+← or Ctrl+4(pad)
Move Mark Right	Ctrl+→ or Ctrl+6(pad)
Move Mark Up	Ctrl+↑ or Ctrl+8(pad)
PF6 to PF12	F6 to F12
PF13 to PF20	Shift+F1 to F8
Rule	Ctrl+Home
Tab Field	→| or Shift+→|

Table 12. Default Key Functions for a VT Emulator Layout

(continued)

Function of Key	Key
VT Enter	Shift+Enter(pad)
VT Find	End[2] or 1(pad)
VT Hold	Pause
VT Insert	Insert or 0(pad)
VT Next	Page Down[2] or 3(pad)
VT Numpad 0 to VT Numpad 9	Shift+0(pad) to Shift+9(pad)
VT Numpad Comma	Shift++(pad)
VT Numpad Minus	-(pad) or Shift+-(pad)
VT Numpad Period	Shift+.(pad)
VT PF1 to VT PF4	F1 to F4
VT Prev	Page Up[2] or 9(pad)
VT Remove	Delete or .(pad)
VT Select	Home[2] or 7(pad)
VT User F6 to VT User F12	Ctrl+F6 to F12
VT User F13 to VT User F20	Ctrl+Shift+F1 to F8

[2] Indicates the key on the main keyboard.

(pad)

Indicates a key on the numeric keypad.

Note: The Enhanced keyboard has some duplicated keys. The functions of the duplicated keys are the same except when you specify a single key. For example, Del means any Delete key, whereas Pad Del specifies only the Delete key on the numeric keypad.

Keyboard Setup (3270 and 5250)

You can use Keyboard Setup to modify the function defined for each key on the keyboard, except some reserved keys.

You can define the following functions for the keys:

- Performing a key function
- Playing a macro
- Entering characters

Note: For 3270, the Enter function is assigned to the Ctrl key, by default. To change this assignment or, if you are using a non-IBM compatible keyboard and the Enter key does not work properly, you need to customize your keyboard. For 3270 and 5250 sessions, you can use the keyboard map files provided with Z and I Emulator for Windows (see Win32 Cut, Copy, and Paste Hotkeys on page 38).

Keyboard File

When you specify a key, you can save the new keyboard layout in a file (.KMP). If you create two or more keyboard files, you can alternate between them as required.

To assign a function to a key on the keyboard:

1. Click **Preferences → Keyboard** from the **Edit** menu or click the map icon on the tool bar.
2. When the Keyboard Setup window appears, select **Customize**.

   Note: Select **Spain** from the **Language** menu during keyboard setup if you want Catalan support.

3. Assign the key functions, referring to the online help for detailed instructions.
4. Save your changes and exit the Customize Keyboard window.
5. Click **OK** after completing the setup.

You can reset either the entire keyboard or specific keys to defaults:

- To reset the entire keyboard, set the current keyboard to **Default** in the Keyboard Setup window.
- To reset specific keys, select a key in the Customize Keyboard window and then select **Default** from the Current Actions for Selected Key box.

Note: You cannot redefine the following keys: Alt, AltGr, Print Screen, Scroll Lock, CapsLock, NumLock, and Shift.

Win32 Cut, Copy, and Paste Hotkeys

Z and I Emulator for Windows includes two .KMP keyboard map files that map the standard Win32 hotkeys for Cut, Copy, and Paste to Ctrl+X, Ctrl+C and Ctrl+V, respectively. You can use these keyboard map files or add the key values to an existing map file.

For 5250 sessions, the .KMP file provided is pcswinkb5.kmp. The remapping is given in Table 13: Win32 Keyboard Map Functions for a 5250 Layout on page 39.

Table 13. Win32 Keyboard Map Functions for a 5250 Layout

Function of Key	Key
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Enter	Enter
New Line	Right Ctrl

The keys PF7 and PF8 remains mapped to Roll Up and Roll Down, respectively.

For 3270 sessions, the .KMP file provided is pcswinkb3.kmp. The remapping is given in Table 14: Win32 Keyboard Map Functions for a 3270 Layout on page 39.

Table 14. Win32 Keyboard Map Functions for a 3270 Layout

Function of Key	Key
Edit Cut	Ctrl+X
Edit Copy	Ctrl+C
Edit Paste	Ctrl+V
Page Up	PF7
Page Down	PF8
Enter	Enter
New Line	Right Ctrl

Part II. Using Z and I Emulator for Windows 3270

Chapter 6. Considerations for Using PC/3270 Sessions

This chapter contains hints and tips for using PC/3270 sessions. Supplementary information other than the items described in this book may be included in the Readme HTML file in the Z and I Emulator for Windows directory.

TN3270E Contention Resolution

Using the TN3270E function negotiation mechanism, Z and I Emulator for Windows negotiates with servers to enable the CONTENTION-RESOLUTION function described in the IETF *TN3270E Functional Extensions* Internet-Draft document. As with any other such negotiation, the server might accept or reject this function.

The CONTENTION-RESOLUTION function is supported only for display sessions.

Z and I Emulator for Windows negotiation for this function is enabled by default. It can be disabled by adding the following keyword to the .WS profile.

```
[Telnet3270]
TN3270EContentionResolution=N
```

Host-Session Window Operations

Cursor Color

By default, PC/3270 draws the underline cursor in white. If the background color of the current field is white, the color of the underline cursor automatically switches to black. PC/3270 draws the block and half-block cursors in the same color as the current field, reversing the background and forground colors. This behavior is identical to a 327x terminal.

If you want to change the default cursor color assignment, modify the pcswin.ini file and add the CursorColor value to the Session stanza, as follows:

```
[Session]
CursorColor=<red_value> <green_value> <blue_value>
```

Here, `<red_value>`, `<green_value>`, and `<blue_value>` are integers from 0 to 255 that specify the color intensity for each color primitive respectively. The color values range from `0 0 0` for black to `255 255 255` for white. PC/3270 then draws the underline cursor and the block cursors in this new color, mixing this new color with the existing screen colors using an XORed (exclusive or) operation. The value `0 0 0` (black) is not recommended because XORing 0 0 0 with any existing color results in the existing color, which makes the cursor invisible.

If you select a blinking cursor, it will always be drawn white, mixing the white with the existing screen colors using an XORed operation. You cannot change the cursor color when it is blinking.

Note: Cursor blinking is disabled by default in Windows Terminal Services sessions and under virtualization environments such as Citrix XenDesktop and Citrix XenApp. Refer to the appropriate vendor documentation to enable cursor blinking.

Releasing Insert Mode with Attention Keys

As on a non-programmable terminal, you can release insert mode when you press an Attention key. If you want this to happen, add this parameter to the [Keyboard] section of the workstation profile (`*.WS`):

```
[Keyboard]
ResetInsertByAttn=Y
```

Scroll Bar

If you choose **Font** from the Appearance menu in the host session window and choose **Fixed Size** from the Select Display Font window, the entire operator information area might not appear on the screen. If you specify **With Scroll Bar**, the OIA will not scroll. The session window size is restricted to be smaller than the screen size.

Scroll-Lock Key

When the Scroll Lock keyboard indicator is turned on, the cursor movement keys and the Page Up and Page Down keys are used to scroll windows only when you specify **With Scroll-Bar** in the Window Setup window. If you specify **Without Scroll-Bar**, you cannot use the Scroll Lock key, because the entire screen is displayed. For example, cursor-movement keys do nothing in Scroll Lock mode.

Customizing a Display Translation Table

PC/3270 displays the host EBCDIC character using the workstation (ANSI) graphic symbol so that the character defined by the zSeries™ EBCDIC host code page is displayed correctly using the same graphic symbol defined by ANSI. However, you might need your original translation, because your host or workstation application is not designed to use the standard translation.

You can use your original translation table if you refer to the following procedure as an example. Note that the data integrity caused by the user-defined table is your responsibility.

The following procedure is an example of how to remap left and right brackets.

1. Terminate all running 3270 sessions
2. Modify the PC/3270 workstation profile (*.WS).

   ```
   [Translation]
   IBMDefaultView=N
   DefaultView=C:\Z and I Emulator for Windows\PRIVATE\BRACKET.XLT
   ```
3. Create the display translation table file (.XLT). In this example, the following BRACKET.XLT file is created in the Z and I Emulator for Windows private subdirectory.

   ```
   [Profile]
   id=XLT
   Description=User-defined Display Translation Table

   [Option]
   ```

```
Replace=Y

[SB Xlate]
; EBCDIC=ANSI
; The next line displays EBCDIC X'AD' as
; an ANSI X'5B' (left bracket)
AD=5B
; The next line displays EBCDIC X'BD' as
; an ANSI X'5D' (right bracket)
BD=5D
```

4. Create your own keyboard layout (.KMP) if you need to enter your new left and right brackets graphic symbols:

   ```
   [Keyboard]
   KEY27=ansi dd
   KEY28=ansi a8
   ```

   The information on the right should be lowercase characters. PC/3270 translates ANSI X'dd' into EBCDIC X'ad'. It is displayed as [by the table created in step 3.
5. Click on the PC/3270 icon corresponding to the modified workstation profile.

Support for Long File Names

Like Windows®, Z and I Emulator for Windows supports long file names. You can give any name (up to 255 characters) to a file; you are not limited to eight characters with a three-character extension. You can use spaces in the file name, but not the symbols /, \, :, *, ?, “, <, >, or |. In addition, the tilde (~) character should not be used in CMS or MVS host file names.

File Transfer Function

Host File Name and Reserved Words

You should not use the following words as a VM file name or file type, as a MVS data set name, or as a CICS® file name, because they are reserved for use as option commands:

- ASCII
- APPEND
- TIME
- CLEAR
- NOCLEAR
- SILENT
- QUIET
- PROGRESS
- BLANK
- CRLF

- BINARY
- NOCRLF

Changing the Packet Size When Import/Export Is Idle

When import/export is idle, select **Preferences → Transfer** from the **Edit** menu. When you change the packet size on the Setup window, end import/export, and then rerun it.

Wait Option for Multiple File Transfer

If multiple file transfers do not succeed, insert the following statement into the [Transfer] section of your .WS file:

```
[Transfer]
wait=1000
```

This parameter causes a 1000 msec (1 sec) delay between file transfers. If this does not help, you might need to increase the value again.

NOTRUNC and BLANK Options (SBCS Only)

If you want to add trailing blanks (spaces) to fill the logical record length for each record when downloading a text file, use the following options in the **Additional Options** edit field of the Transfer-Type Definition window.

NOTRUNC : for VM/CMS (PTF# UR35492)
NOTRUNC : for MVS/TSO (PTF# UR34797)
BLANK : for CICS®

Setting the VTAM® PSERVIC Statement

File transfer problems can occur if extended attribute support has not been set on in the VTAM® PSERVIC statement. For extended attribute support, set on the high order bit in PSERVIC byte1 (zero byte origin) as follows: x'xx80xxxxxx...'.

Entry assist feature in 3270 display session

The Entry Assist (DOC mode) features enable easier editing of text documents. The Entry Assist features are available only for 3270 display sessions.

Enabling DOC mode

Enabling DOC mode The Entry Assist features can be enabled by adding the keyword Docmode=Y to the .WS session profile. Set Docmode=N to disable Entry Assist.

When Entry Assist is enabled, the DOC indicator is displayed in positions 67, 68, and 69 of the Operator Information Area (OIA). Also, all the pasting functions on the Paste tab are ignored (see Pasting in DOC mode on page 46).

You can assign the mode toggle to a keyboard key or button in the Popup Keypad. The default key assignment for DOC mode is Alt+F1.

Word wrap

You can enable this function by adding the keyword DocmodeWordWrap=Y to the .WS session profile. When DOC mode is enabled, this function is enabled by default. When word wrap is enabled, the Word Wrap indicator is displayed in position 71 in the OIA.

When word wrap is enabled, a word that is being typed at the right margin is moved in its entirety to the first unprotected field in the next row, assuming that the unprotected field has enough left side blank space (spaces and nulls) to contain the word. The vacated area on the previous row is filled with spaces. See the following examples.

When word wrap is disabled:

```
Look in the diction
ary, please.
```

When word wrap is enabled:

```
Look in the
dictionary, please.
```

If the unprotected field does not have sufficient blank space at the left, then the word is not moved (same as when word wrap is disabled).

Start Column and End Column

You can set these values using the keywords DocmodeStartColumn=<value> and DocmodeEndColumn=<value> in the .WS profile. These settings control the left and right margins while the session is in DOC mode. For example, you can set the left margin to 10 and the right margin to 60. If insert mode is on, the following conditions apply:

- When you type a character at the right margin, the cursor returns to the left margin on the next row.
- Protected fields are skipped.

If insert mode is off, the following conditions apply:

- When you are typing data and a character is pushed beyond the right margin, the pushed character is wrapped to the beginning of the next row.
- Protected fields are skipped.

When insert mode is enabled, the ^ character is displayed at position 52 in the OIA.

New Line key

If you press the key (default is Enter key) for New Line, the cursor skips to the first unprotected character position in the next row (or in a subsequent row if necessary) that lies within the margins.

Tab stops

You can set tab stops values with the keyword DocmodeTabStops=<column1,column2,...> in the .WS profile. You can set the tab stops by specifying the number of the columns that you want for the tab stops, separated by commas (for example, 5,10,15,20,25).

When tab stops are set, pressing the Tab key causes the cursor to skip to one of the following, whichever comes first:

- The next tab stop in the same unprotected field on the same row. Note that tab stops cannot be defined outside the left or right margin.
- The first character position in the next unprotected field on the same row, if that character position is within the margins.
- The first character position in the next unprotected field in a subsequent row, if that character position is within the margins.

The following conditions apply when using tab stops:

- Characters that are skipped over as the result of a tab key are not set to blanks.
- Characters that lie within an unprotected field and that the cursor skips over as the result of a tab key are not set to blanks. However, nulls that the cursor skips over as the result of a tab key are set to blanks.

Nulls in an unprotected field straddling a margin

When the tab key causes the cursor to skip to the left margin of the next line, and an unprotected field straddles the left margin, nulls that lie outside the left margin in the unprotected field are converted to spaces. However, when the tab key causes the cursor to skip to the next line, and an unprotected field straddles the right margin of the preceding line, nulls that lie outside the right margin in the unprotected field are not converted to spaces.

Enable audible End of Line signal

You can enable this function by adding the keyword DocmodeEndofLineSignal=Y to the .WS profile. The function enables an audible signal when the cursor enters the column set for the End of Line signal column.

End of Line signal column

You can enable this function by adding the keyword DocmodeEndofLineSignalColumn=<value> to the .WS profile. You can set the column value at which you want the End of Line signal to be sounded. You could type in a value that would cause the audible signal to occur when the cursor approached the right margin. For example, if the right margin is 70, you might set the End of Line signal column to 65.

Pasting in DOC mode

When DOC mode is enabled, all pasting functions on the Paste tab are ignored and pasting is done according to the following rules.

If the cursor is within the margins:

- Start pasting characters at the cursor.
- Paste characters according to the same rules that are used when the user is typing characters.
- Follow the word wrapping rules if Word Wrap is enabled.
- Display the Too Much symbol in the OIA if the end of the last row is reached before all the data is pasted.

If the cursor is outside the right margin:

- Start pasting characters at the cursor.
- Paste characters into unprotected fields until the end of the row is reached.
- Start at the left margin of the next row and continue pasting according to the rules that apply when the cursor is within the margins.

If the cursor is outside the left margin:

- Start pasting characters at the cursor.
- Paste characters into unprotected fields until the left margin is reached.
- Continue pasting according to the rules that apply when the cursor is within the margins.

Graphic Functions

This section provides information, restrictions, and considerations for graphic functions.

Graphics Protocols

Z and I Emulator for Windows allows you to use host graphics applications, such as GDDM® and others. Two types of graphics are supported:

- Vector
- Programmed symbols

Two protocols are supported for vector graphics:

- Advanced
- Native

See Configuring Graphics in the online helps for a description of these protocols and to learn how to configure your sessions for graphics.

The following functions are supported:

- Multiple mixed alphanumeric and graphics host sessions
- Use of standard OS/2® printing and plotting facilities
- Creation of PIF (Picture Interchange Format) files
- Clipping graphics data into the clipboard

Vector Graphics

Vector graphics are computer graphics in which display images are generated from display commands and coordinate data. Z and I Emulator for Windows provides vector graphics support for the OS/2-Link (advanced) or the 3179G or GOCA (native) protocols. Choose the protocol that is appropriate for your host applications.

Advanced Protocol

Use the advanced protocol when you have GDDM® Version 2 Release 3 or later and are using any of the following operating systems:

- MVS
- VSE
- VM/SP
- VM/XA SP™

Note: The advanced protocol is not supported by the CICS® pseudo-conversational mode with versions of GDDM® earlier than Version 3, and not by IMS/VS at all. It is, however, supported by the CICS® pseudo-conversational mode with GDDM® Version 3 Release 1 or later.

The advanced protocol is equivalent to that used by OS/2-Link, so it supports the same subsystems. However, no download of code from the host system is required for Z and I Emulator for Windows because all the OS/2-Link graphics modules are integrated into the program.

Native Protocol

Choose the native protocol when you intend to use older GDDM® versions or non-GDDM host-graphics applications, such as those originally intended for use on 3270 nonprogrammable terminals as the 3179G, 3192G or 3472G. The native protocol also allows IMS/VS users to display GDDM® graphics.

Programmed Symbols

Raster graphics are displayed with programmed symbols, which are downloaded to your workstation. Z and I Emulator for Windows supports up to six sets (PSA through PSF) of triple-plane and multiple-color programmed symbols.

Use programmed symbols as the graphics type when you intend to use host graphics applications originally written for the 3279G terminal.

Graphics applications use one or both of these methods to display graphical screens. Z and I Emulator for Windows allows you to enable or disable support for vector graphics and programmed symbols. Choose the type of support that our host applications require.

Note: If you use the OS2-Link (advanced) protocol under the GDDM® program, do not choose programmed symbols. Also, do not choose programmed symbols when you use the OS2-Link protocol with other applications.

Enabling Programmed Symbol Sets

PC/3270 provides up to six sets of triple-plane programmed symbols, depending on the type of graphics support that you choose. By default:

- Two sets (PSA and PSB) of single-plane programmed symbols are usable if you choose both programmed symbols and vector graphics.
- Three sets (PSA, PSB, and PSE) of single-plane programmed symbols and three sets (PSC, PSD, and PSF) of triple-plane programmed symbols are usable if you choose programmed symbols, but not vector graphics.

You can change the number of programmed-symbol sets and triple or single planes available for each programmed-symbol set by editing the [3270] section of the workstation profile:

```
PSSPlanes=xxxxxx
```

Each *x* represents a number (0, 1, or 3) that indicates how many planes are to be available for each set; the first column indicates the number of planes for PSA, the second column for PSB, and so on. For example, to enable six triple-plane programmed symbol sets, enter the following:

```
PSSPlanes=333333
```

To enable two single-plane and two triple-plane sets, enter the following:

```
PSSPlanes=113300
```

How to Handle Errors Caused by Insufficient Memory

Graphic execution module PCSGRP.DLL uses a large amount of global memory for graphic drawing or printing. When the workstation has insufficient installed memory, results might not be correct. For example, an area might not be clearly shaded.

In this case, increase the amount of installed workstation memory by at least 1 MB. For host graphic printing, add 1 more megabyte.

Memory might have to be further extended depending on the host graphic application and printer driver used.

Drawing-Buffer Size

The drawing-buffer size varies depending on the contents set for Redraw of a graphic function.

To set **Redraw**, click **Preferences → Appearance → Display Setup** from the **Edit** menu in the session window. Select **Graphics** from **Category**.

Selecting **Host** from the optional items of **Redraw** requires no buffer.

If you select **Retained**, the graphic execution module stores all redrawing data into a buffer. Such a buffer is called a *retained buffer*. The buffer size varies depending on the complexity of the graphic data from an application program. For example, a simple table has a buffer size of 10 KB to 20 KB, while a complicated graphic image has a buffer size of 200 KB to 300 KB.

When you select **Bitmap** to set **Redraw**, the buffer size will be the same as the sum of the retained buffer size and compatible bit map size:

(Height) x (Width) x (Number of planes) x (Bits/Pixel) / 8 bytes

For example, when you select a 7x12 font for a VGA 16-Color Display Model 2 (24x80), the bitmap size is:

(7x80) x (12x24) x 1 x 4 / 8 = 80 KB

When you select a 12x20 font for an IBM® PS/55 High-Resolution 256-Color Display Model 2, the bitmap size is:

(12x80) x (20x24) x 1 x 8 / 8 = 460 KB

Using Bitmaps for Drawing

The graphic execution module uses a bit map compatible with the display unit to draw an area instruction in overpaint mode. An image instruction requires one plane bit map.

(Buffer for area)	=	(Area width) x (Area height) x (Number of planes) x (Bits/Pixel) / 8
(Image buffer)	=	(Image width) x (Image height) / 8

Print Buffer Size

The retained buffer must be used for printing. The retained buffer is the same size as that used for redrawing. This is also applied when you specified **Bit Map** for **Redraw** on the Display Setup window.

If graphic printing is called in Bitmap mode, the graphic printing module generates a bitmap compatible with the connected printer, draws an image on the bitmap, and transfers the bit image to the printer.

This operation is generally performed quickly. When memory is frequently swapped, the process slows down in proportion to the number of swap operations. If a large bit map is not allocated, the graphic printing module prints a graphic image normally using only the retained buffer.

Example:

Proprinter (240x144 DPI) character size:
Bitmap size = (240x8) x (144x11) x 1 x 1 / 8 = 380 KB

Example:

EPSON (ESC/P) (360x180 DPI color) character size:
Bitmap size = (360x8.5) x (180x11) x 3 x 1 / 8 = 2.3 MB

Note: With some printers, different printing results might be obtained in bitmap mode and non-bitmap mode. If the desired results are not obtained, change the current bitmap mode. For example, specify non-bitmap mode to print in bitmap mode.

Edit-Copy Buffer

An editing operation causes the graphic execution module to copy a bit map and DIBitmap to the clipboard. The bit map is compatible with the display; DIBitmap is a 4-bit/pixel bit map.

Printer Fonts

The printer driver can handle two font sets, the device font and GDI font. The device font is a hardware font built into the printer. The GDI fonts are System (without brackets) or other software fonts for Windows®.

When you select a font set for graphic printing from the Printer Control window, use the GDI fonts for the following reasons:

- In bitmap mode, a GDI font can be used for printing. However, the device font cannot be used, because an image cannot be drawn on a memory bit map when using the device font.
- In bitmap mode, the device font cannot be used for printing when OR and exclusive OR attributes are mixed.

Plotter

Because a plotter does not support a raster, the following restrictions are imposed on drawing. Use a plotter for figures and tables that have mainly lines.

- No shading is supported.
- Some shading patterns cannot be distinguished.
- Image order drawing requires much time, and the final printout is of poor quality.
- The OR and exclusive OR are not correctly reflected.

Hole in Screen Caused by Clearing a Graphic Character

When a character overlaps a graphic image, the graphic image is cleared at the position where the character is to be displayed. When you enter a null character or space having the transparent attribute at the position where a graphic image is displayed, the graphic image in that character cell is not cleared.

If you select **Host** or **Retained** to set **Redraw** on the Display Setup window, when characters in a graphic image are cleared, a hole appears in the graphic area. This is because these two modes do not have a bitmap image, and partial redrawing cannot be performed on the screen.

If you select **Bitmap** mode as **Redraw Graphics**, you can find no hole on the graphic region by the application that overrides any alphanumeric characters (as well as NULL and SPACE) on the graphic image.

To restore the screen, perform either of the following actions:

- Press the PA3 key to have the application program redraw the screen.
- Minimize and restore the graphic image retained in Retained mode, then redraw it or select Bitmap mode.

Note: When you change the setting of **Redraw** in the Display Setup window, the set contents are valid from the next drawing.

Miscellaneous Restrictions for Graphic Functions

If advanced protocol is selected, graphic functions cannot be used in the IMS/VS and CICS® pseudo-conversational mode with versions of GDDM® earlier than Version 3.

Considerations for Graphics Functions

Native-Graphics Datastream

If the host sends an Object Structured Field (Object Picture, Object Data, Object Control) with a zero value in the length field, Z and I Emulator for Windows rejects it and displays PROG754.

Printout to LPT1

When you use a host application that prints to your PC's LPT1, you must first select the printer in the Printer Setup dialog of the File menu.

Print Processing

Chapter 7. Transferring Files

Z and I Emulator for Windows File Transfer enables you to transfer one or more files between a host system and workstation at the same time. Transfer types and translation tables can be defined in advance.

You can perform the following file transfer functions:

- Send files to the host system
- Receive files from the host system
- Use lists of files
- Create templates to define file names and transfer types
- Define transfer types
- Set transfer options
- Modify translation tables
- Import or export files (PC/3270 CICS only)
- Create interactive document profile (IDP) files (PC/3270 CICS only)
- Transfer files via the XMODEM or YMODEM protocols

Note:

PCT400 was withdrawn from marketing 3/98.

Host Requirements

For PC/3270 File Transfer in SBCS mode, you need one or more of the following host file-transfer programs (referred to as IND$FILE):

- IBM 3270-PC File Transfer Program, 5665-311 (MVS/TSO)
- IBM 3270-PC File Transfer Program, 5664-281 (VM/SP 2.1)
- IBM CICS/VS 3270-PC File Transfer Program, 5798-DQH (CICS/VS 1.5)

Sending Files to the Host System

To send a file from your workstation to the host system:

1. Sign on to the host system.
2. Click **Send File to Host** from the **Actions** menu of the session window. (You can also select the **Send** button on the tool bar.)

   The Send File to Host window opens.
3. To use a list file, click **Open List**. Select the list to be used for transfer. See Creating List Files on page 56 for details of how to create list files.

If you do not want to use a list file, proceed to the next step.

4. Type the name of the **PC File** to be sent to the host system, or click **Browse** to select the file. If a template is provided for the file type you are transferring, the host file name and transfer type appear automatically.
5. Type the **Host File Name**. For MVS/TSO, you can click **Browse** to view the datasets and members on the host (3270 only). Select the files to send, then click **OK** to add the files to the transfer list.
6. Select the **Transfer Type**.
7. Click **Send**.

   The file is sent to the host system. The send status appears in the Send a File Status window.

Receiving Files from the Host System

To transfer a file from the host system to your workstation:

1. Sign on to the host system.
2. Click **Receive File from Host** from the **Actions** menu. (You can also select the **Receive** button from the tool bar.)

   The Receive File from Host window opens.
3. To use a list file, click **Open List**. Select the list to be used for transfer. See Creating List Files on page 56 for details of how to create list files.

   If you do not want to use a list file, proceed to the next step.
4. Type the name of the **Host File** to be received. You can also specify the host file name as follows:
   - **Using the Clipboard button**

     If you have copied one or more host file names to the clipboard, you can click the **Clipboard** button and paste the names into the transfer list. Select one or more of the pasted file names to be transferred and click **OK**.
   - **Using the Browse button**

     For MVS/TSO, you can click **Browse** to view the datasets and members (3270 only). Select one or more of the files to receive, then click **OK** to add the files to the transfer list.

   If a template is provided for the file type you are transferring, the PC file name and transfer type appear automatically.
5. Type the **PC File Name** or click **Browse** button to select a location for the file.
6. Select the **Transfer Type**.
7. Click **Receive**.

   The receive status appears in the **Receive a File Status** window.

Using List Files

If the same files are transmitted frequently, you can create a list of the files and save it.

A list file can be used for both Send and Receive. The default list file extension is .SRL.

Creating List Files

To create a list file:

1. Select **Receive File from Host** from the **Actions** menu or **Send File to Host** from the **Actions** menu of the session window; or click the **Send** or **Receive** buttons on the tool bar.

   The corresponding window opens.
2. Select a file to be transferred from the **Host-File Name** or **PC-File Name** list box by pointing to the name of a file to be selected. While holding down the Ctrl key, click the left mouse button.

   The file name, its corresponding workstation or host file name (according to the available templates), and the transfer type appear in the **Transfer List** part of the window.

   Note: You can also click the **Browse** button (for sending files) or the **Clipboard** button (for receiving files) to open the corresponding dialog box, which allows you to select files for transferring; when you click **OK**, the selected files are shown in the **Transfer List**.
3. Click the **Add to List** button to include a selected file in the **Transfer List**.
4. After all desired files have been selected, click **Save List**.

   The Save File-Transfer List File As window opens.
5. Enter or select a list name, and click **OK**.

Editing Lists

To edit the contents of a previously created list:

1. As explained in Sending Files to the Host System on page 54 and Receiving Files from the Host System on page 55, display the Send File to Host or Receive File from Host window.
2. Select **Open List**.

   The Open File-Transfer List File window opens.
3. Select the name corresponding to the list file to be edited, then click **OK**.
4. The contents of the selected list appear in the Send File to Host or Receive File from Host window.
5. Edit the contents of the list file.

Changing the contents of a list: Choose the file to be changed from the list, and overwrite the items to be changed in the text box; then click the **Update in List** button.

Removing a file from the list: Choose the file to be removed, and click **Remove from List**.

Adding a file to the list: Double-click the file to be added from the list of host or workstation files.

6. Select **Save List**.

   The Save File-Transfer List File As window opens.
7. Enter a name and then click **OK**.

Managing Templates

A *template* is a set of rules to be used by the workstation to automatically generate a workstation or host file name and transfer type when you specify a file to be sent or received.

You can have up to 32 templates. They are automatically numbered from 1 to 32.

When you specify a file to be transferred, the workstation scans the templates, starting from template 1. It uses the first matching template to generate a name for the transferred file and the transfer type.

To manage a template:

1. Click **Receive File from Host** from the **Actions** menu or **Send File to Host** from the **Actions** menu of the session window; or click the **Send** or **Receive** buttons on the tool bar.

   The Send File to Host or Receive File from Host window opens.
2. Select **Template**.

   The Template window opens. The contents of the window depend on the connected host system.

Adding Templates

The list box for the Template window lists the currently stored templates.

To add a template:

1. Select any template from the list box.

   The contents of the selected template appear under the list box.
2. Change the workstation or host file names or extensions by overwriting them; then select the transfer type. (For details of the transfer types, see Defining Transfer Types on page 58.)
3. Click **Add**.

   The window for determining where in the list to display the new template opens.

4. Select a template number and specify whether to display the new template before or after the template that has that number. Click **OK**.

   The new template is added to the list in the appropriate position.

Replacing and Deleting Templates

To change the contents of a currently stored template, or to delete a template:

1. Select the template to be changed or deleted.

   The contents of the selected template appear under the list box.
2. To change the contents, overwrite the appropriate part and then click **Replace**.

   To delete a template, click **Delete**.

   The selected template is changed or deleted, and the contents of the template list box are changed.

Testing Templates

To test the contents of an added or changed template:

1. Select the template to be tested from the list box.

   The number of the selected template appears in the Test Templates box in the lower part of the window.
2. Select or enter data for the following items:

   Test Mode

   Determine which mode is to be used for the test: the mode in which a file is transmitted from the workstation to the host system (send), or the mode in which a file is transmitted from the host system to the workstation (receive).

   Templates

   Determine which templates to test: only the template selected in step 1, or all registered templates.

   Source File

   Enter the name of the file to be used for the test.
3. Click **Test**.

   Target File indicates the name that has been generated by the template.

Note: Testing a template does not transfer a file.

Defining Transfer Types

Transfer types define the option information used for controlling file transfer. Up to 32 transfer types can be defined for each host system. Text, binary, and append (excluding CICS) are the defaults.

To add or change transfer types:

1. Click **Edit → Preferences → Transfer** from the session window.
2. Click the tab for your host type or modem protocol.

   The property page for the selected host or modem protocol opens. The items that appear depend on the selected host system.
3. Enter transfer-type names in the **Transfer Type** box, or select them from the drop-down list.
4. Select or enter the required items (see Items to Be Specified on page 59).

   To add or replace a transfer type, click **Save**. To delete a transfer type, click **Delete**.
5. A dialog box displays, asking for confirmation. Click **OK**.

Items to Be Specified

Choosing the appropriate property page enables you to set the items described in the following sections.

File Options

The file options that can be used depend on the type of the connected host system and the host code page selected when the session was configured. Table 15: Mode Values for File Transfer Options on page 59 lists the mode values for the file transfer options. Table 16: Transfer File Options on page 59 lists the transfer options.

Table 15. Mode Values for File Transfer Options

Mode	Host Code Page
SBCS	Others

Table 16: Transfer File Options on page 59 lists the options for PC/3270.

Table 16. Transfer File Options

File Option	Host System	Mode	Conversion Details
ASCII	VM/CMS MVS/TSO ICS	SBCS	Converts codes as follows when a file is sent: • Converts 1-byte workstation codes to EBCDIC codes • Converts RS (hex 1E) and US (hex 1F) to SO (hex 0E) and SI (hex 0F) Converts codes as follows when a file is received: • Converts EBCDIC codes to 1-byte workstation codes

Table 16. Transfer File Options (continued)

File Option	Host System	Mode	Conversion Details
CRLF	VM CMS MVS/TSO CICS	SBCS	Converts codes as follows when a file is sent: • Does not remove CRLF (hex 0D0A) from the end of each line. The code is treated as a delimiter for each record. • Removes EOF (hex 1A) from the end of the file. Converts codes as follows when a file is received: • Adds CRLF (hex 0D0A) to the end of each line. • Adds EOF (hex 1A) to the end of the file. Removes EOF from the existing file, and appends EOF to the end of the added file when APPEND is specified.
APPEND	VM/CMS MVS/TSO	SBCS	Appends the sent file to the existing host file. Appends the received file to the existing workstation file.
SRC	OS/400 i5/OS	SBCS	This option is good only for Send. When SRC is checked, the target file is stored as a member of the physical source file. If the file already exists on the Host system, this option is ignored.

Record Format

Valid only for VM/CMS and MVS/TSO when APPEND is not specified for file transmission. You can select any of the following:

- **Default**
- **Fixed** (fixed length)
- **Variable** (variable length)
- **Undefined** (undefined mode for MVS/TSO only)

If you select the **Default** value, the record format is selected automatically by the host system.

Specifying **Variable** for VM file transfer enables host disk space to be used efficiently.

Logical Record Length (LRECL)

Valid only for VM/CMS and MVS/TSO when APPEND is not specified for file transmission.

Enter the **logical record length** to be used (host record byte count) in the **LRECL** text box. If **Variable** and **Undefined Mode** are specified as the record format, the logical record length is the maximum record length within a file. The maximum value is 32767.

The record length of a file sent from a workstation to the host system might exceed the logical record length specified here. If so, the host file transfer program divides the file by the logical record length.

To send a file containing long records to the host system, specify a sufficiently long logical record length.

Because the record length of a workstation file exceeds the logical record length, a message does not appear normally if each record is divided. To display a message, add the following specification to the [Transfer] item of the workstation profile:

```
DisplayTruncateMessage = Y
```

TSO Allocation Parameter (MVS/TSO)

Valid only for MVS/TSO when **APPEND** is not specified for file transmission. The following items can be specified:

[Allocation Amounts]

Primary

Enter the number of tracks or cylinders allocated to this file transfer.

Secondary

If the primary allocation is not sufficient for the entire file transfer, enter additional storage capacity allocated to the file transfer.

[Allocation Units]

Tracks

Specify this parameter to allocate a host file by track. Ask your system manager whether to use tracks or cylinders as the unit.

Cylinders

Specify this parameter to allocate a host file in units of cylinders.

AVblocks

Specify this parameter to allocate a host file in units of blocks.

[Block size]

This item is used only to create a new data set. Enter the block size of a new host data set, in bytes, in the text box. If this item is omitted, the workstation assumes the value that appears in the **Logical Record Length** box. The maximum value is 32767. If **AVblocks** is selected, the block size is the block size of the new data set.

Additional Options

You can enter the required host command options in the **Additional Options** text box.

Setting General Transfer Options

To set advanced options:

1. Click **Edit → Preferences → Transfer** from the session window.

   The setup dialog is displayed.
2. Change the required settings on the property page labeled **General**.
3. Click **OK**.

The following sections contain information about the items which can be defined for file transfer options.

Host Type

You can specify from the drop-down list box the type of host (MVS/TSO, VM, or CICS) to which your workstation is connected.

Host Command

You can specify host command to be called when file transfer starts. If nothing is entered in this text box, IND$FILE or its equivalent for other countries is used for 3270 SBCS sessions.

Default PC Directory

You can specify the default directory that appears in the Send File to Host or Receive File From Host window. To select the directory, click the **Browse** button.

Default Partitioned Data Set (MVS/TSO Only)

You can specify the MVS partitioned data set to be used as the default.

Default VM Disk (VM Only)

You can specify the VM disk to be used as the default.

PC Code Page

When a file is transferred, EBCDIC codes are converted to 1-byte workstation codes, and vice versa. A valid value is automatically selected from among the following values for SBCS sessions: 437, 737, 806, 813, 819, 833, 850, 852, 854, 857, 858, 860, 861, 862, 863, 864, 865, 866, 869, 874, 912, 915, 916, 920, 921, 922, 1008, 1089, 1124, 1125, 1127, 1129, 1131, 1133, 1153, 1155, 1156, 1157, 1158, 1160, 1164, 1250, 1251, 1252, 1253, 1254, 1255, 1256, 1257, and

1258; —according to the host code page specified when the workstation is configured. For an explanation of how to select host code pages, see the online help for the host code page.

Packet Size

The amount of memory (in bytes) used by the workstation for transmission and reception. If a large value is entered, a file is transferred more quickly, but the memory overhead is larger. The default value is 12288. In the case of Telnet3270, you can specify a packet size larger than 8000 bytes by adding the following line to the Telnet3270 stanza in your workstation profile:

```
SendBufferSize=nnnn
```

File-Transfer Timeout

You can define the time the workstation waits for a response from the host system (in seconds). If the host system does not respond, the transfer is canceled, and an error message appears. A number in the range 20–65535 (or 0) can be specified. The default is 60 seconds for ASCII sessions; for all others, it is 30 seconds. Specify an appropriate value such that the error message does not appear too early. If you specify 0, a timeout is not set.

If a packet or block size is relatively large for low-speed lines, such as COM port lines, it is recommended that 150 seconds or greater be specified.

Extension for List-Files

You can change the default extension (.SRL) of file-transfer list files.

Clear Session Before Transfer

You can specify whether a Clear command is sent to the host system before a file is transferred. Choose any of these option buttons:

Default

A Clear command is sent before a file is transferred (VM/CMS or CICS only).

Yes

A Clear command is sent for MVS also.

No

A Clear command is not sent for any host system.

Show Status Window

You can choose the method of displaying the file-transfer-progress status.

In Session

When file transfer starts, the status window opens. The name of the file being transferred and the transfer progress appear.

In icon

When file transfer starts, the status icon appears on the screen. If the icon is restored, the status window opens.

Setting Up the Translation Table

You can create or edit the translation table to be used for sending or receiving files.

Changing the Translation Table

To change the translation table:

1. Click **Edit → Preferences → Transfer** from the session window.
2. Click the **Translation Table** tab on the resulting window.

   The Translation-Table Setup property page opens.
3. The table currently being used (default or the name of a user-defined table) is shown. Choose either **Default** or **User-Defined**.
4. If you choose **User-Defined**, enter a translation-table name in the **File Name** text box, or select a name by clicking **Browse**.
5. Click **OK**.

Customizing the Translation Table

You can create a user-specific translation table for transmission or reception, or you can edit an existing translation table.

To create or edit a translation table:

1. On the **Translation Tables** property page, click **Customize** in the Upload or Download window.

   The Customize Translation window opens.

   If you chose **Default** or if you chose **New** from the File menu, the default values appear in the table.

   Translation source codes

   PC code-points when an upload translation table is edited. Host code-points when a download translation table is edited.

Translation target codes

Host code-points when an upload translation table is edited. PC code-points when a download translation table is edited.

2. Double-click the code to be changed in the table, and change the value in the entry field that subsequently appears.
3. Click **Save** or **Save As** from the File menu.
4. If asked, enter a name in the Save Translation File As window and click **OK**.
5. Click **Exit** from the File menu of the Customize Translation window.

Import/Export (3270 CICS Only)

Import/Export is an office system communication program and an application program executed under the IBM Customer Information Control System (CICS).

Clicking **Import/Export** loads a module into workstation memory. You can then start Import or Export from a menu on the host screen.

When you export a document from the host, the workstation receives two files: one is the file itself, and the other is the interchange document profile (IDP) file, which contains document header information.

When you **Import** a file to a host system, it must be accompanied by an IDP file of the same name. If the necessary IDP file does not exist, you can create it as described in .

To transmit files using Import/Export:

1. Verify that the window of the host session is active and ready for file transfer.
2. Click **Import/Export** from the **Actions** menu of the session window.

   The minimized Import/Export Status window opens.
3. Click **Import** or **Export** from the host application menu.
4. Specify the host and workstation file names of the file to be transferred. Run Import or Export.

   When Import or Export starts, the Import/Export Status window is maximized.

   After the file is transferred, the window is closed.

File Transfer Commands for PC/3270

You can send data files to and receive them from IBM host systems that are running:

CICS/MVS

Customer Information Control System running under MVS

CICS/VSE

CICS running under Virtual Storage Extended

MVS/TSO

Multiple Virtual Storage/Time Sharing Option

VM/CMS

Virtual Machine/Conversational Monitor System

For more information on using these commands, click **Send File to Host** from the **Actions** menu and **File Transfer from Command prompt** in the help panel.

File Transfer Methods

You can transfer files in the following ways with PC/3270:

- By clicking **Receive File from Host** from the **Actions** menu or **Send File to Host** from the **Actions** menu of the workstation window
- By using the **SEND** and **RECEIVE** commands at the DOS command prompt
- By using an EHLLAPI application that invokes file transfer
- By using a macro that has send or receive commands as macro statements
- By clicking the **Send** or **Recv** icon on the tool bar

Requirements and Restrictions

Install the file transfer program, IND$FILE, on your host system. Ask your system administrator for additional file transfer procedures and precautions. An alternate host command name can be used by defining a DOS environment variable IND_FILE in AUTOEXEC.BAT or in a particular DOS box. For example:

```
SET IND_FILE = MYXFER
```

You should not use the following words as a VM file name or file type, as an MVS data set name, or as a CICS file name, because they are reserved for use as option commands.

ASCII, APPEND, TIME, CLEAR, NOCLEAR, SILENT, QUIET, PROGRESS,
BLANK, CRLF, BINARY, NOCRLF

If you want to send to or receive from a subdirectory other than \Z and I Emulator for Windows, you must specify the full path name.

Sending and Receiving Files from the DOS Command Prompt

The workstation is the point of reference for the SEND and RECEIVE commands: You send from the workstation to the host and receive from the host to the workstation.

To send or receive a file:

1. Make sure you are logged on to your host.
2. Make sure the **Ready** message of the host system is displayed, except if you are transferring files through the command option of the ISPF application.

   Note: In the latter case, you must specify the NOCLEAR option for the file transfer command.

   If your screen is blank, make sure that no applications are running and that your host session is not in a *holding* state.

   Note: If you receive any messages from host application programs while you are transferring files, the transfer might not succeed. To prevent messages from interfering, enter the appropriate host command to set messages off temporarily. When file transfer is finished, set messages on again.
3. Switch to your DOS window session or DOS full-screen session.
4. If you use a hard disk, make sure the SEND.EXE and RECEIVE.EXE files are in your current directory or in your path. If you want to send to or receive from a subdirectory other than \Z and I Emulator for Windows, you must specify the full path name.
5. Type the appropriate SEND or RECEIVE command at the DOS command prompt.

   Details on the SEND and RECEIVE commands and their options are explained in the following sections.

Using the VM/CMS SEND Command

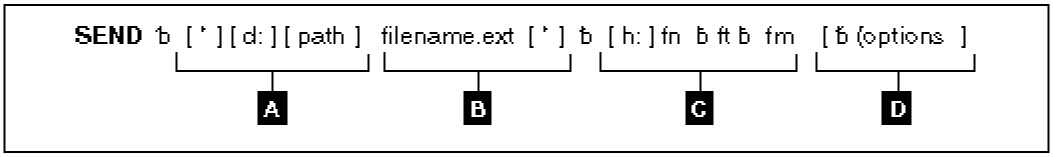
Use the following information when sending a file to VM/CMS:

Figure 1: VM/CMS SEND Command Syntax on page 67 shows the command and information that you must provide. Enter it as shown (including parentheses). You can use either uppercase or lowercase letters.

Note:

1. ƀ means to insert a space. There must *not* be a space between **h:** and **fn**.

Figure 1. VM/CMS SEND Command Syntax



```
SEND ƀ [ ' ][ d: ][ path ]  filename.ext [ ' ] ƀ [ h: ] fn ƀ ft ƀ fm  [ ƀ (options ]
```

A

The workstation drive and path of the file to send.

B

The name of the workstation file to send.

C

Host session specifications for the file to be sent to the host.

`h:`

The short name of the session (which can be omitted if it is **a**)

`fn`

File name

`ft`

File type

`fm`

File mode

D

Optional changes made to the file during transfer. More than one option can be selected. Valid options are:

- APPEND
- ASCII
- CLEAR
- CRLF
- LRECL `n`
- NOCLEAR
- PROGRESS
- QUIET
- RECFM `x`
- TIME(`n`)

The parts of the VM/CMS SEND command are:

SEND

The command.

`d:`

The name of the diskette or hard disk drive on which the file is located.

`path`

The path to the subdirectory that the file is in.

`filename.ext`

The name of the file to be sent, including the extension.

`h:`

The short name of the host session to which you want to send the file. The default is `a:`.

`fn ft fm`

The name the file is to have on your VM/CMS disk. You must specify the file name (`fn`) and file type (`ft`). You can omit file mode (`fm`) if you want the file placed on your A-disk. You can create a new name or use a name that is already on your disk. If you use a new name, the file that you send is added to your disk. If you use the name of an existing file, the file that you send either replaces or is added to the old file. (Refer to the description of the APPEND option.)

(`options`

These options can be specified:

APPEND

Specifies that the file being sent is to be added to the end of an existing VM/CMS file. Omit this option if you want the file to replace an existing file. You cannot specify the LRECL `n` or RECFM `x` option if you use the APPEND option.

ASCII

Performs the following:

- Converts 1-byte workstation codes to EBCDIC codes.

CLEAR

Clears the workstation window at the beginning of the file transfer. **CLEAR** is the default.

CRLF

Specifies preserving of the carriage return and line feed codes. You need the ASCII and CRLF options for text or source files that you want to view or edit, such as SCRIPT files. You do not need them for binary files, such as programs.

LRECL `n`

Specifies the file’s record length. Include a record length only if you want the file to have a record length on your VM/CMS disk other than 80. Replace `n` with the record length you want. If you omit this option, the record length is set to 80 for fixed-length records or to a maximum of 80 for variable-length records.

NOCLEAR

Suppresses the sending of a Clear command at the beginning of the file transfer.

PROGRESS

Shows a message indicating that the file transfer is in progress or has ended. Such messages do not show the current transferred bytes.

QUIET

Does not show any messages.

RECFM `x`

Specifies the file record format. Use this parameter to specify variable-length or fixed-length records in the file. Replace `x` with V for variable or F for fixed. By default, the file has fixed-length records unless you specify the CRLF option; then the file has variable-length records unless you specify otherwise.

TIME(`n`**)**

Specifies the length of time `n`, in units of 30 seconds, that the program waits for a response from the host before it sends an error message. Replace `n` with an integer value in the range from 0 through 2184. If you specify 0, timeout will not be set. The default is 1. To avoid a premature error message, specify an adequate value. In cases of large packet sizes, of large block sizes, or for slow communication lines (such as and COM port), *5* (150 seconds) is recommended. There should be no blank spaces is between TIME and (`n`).

Command Syntax for Sending Files to VM/CMS

The following examples show the command syntax you can use to send files to a VM/CMS host. The parameters of the SEND command can be combined into a single set of parentheses.

- To send a workstation file from your default drive and add it as a new file on your VM/CMS A-disk:

  ```
  SEND pc.txt a:cmsfile script a (ASCII CRLF LRECL 72 RECFM V
  ```

  Note: If you use a command that exceeds one line, do not press Enter when you fill that line; continue typing your command.

  This command sends a workstation file named PC.TXT from your default drive to your host in your host session named **a**. You do not need to specify the workstation drive if the file you are sending is on the current drive. The command creates a new file, named CMSFILE SCRIPT, on your A-disk. The records in the file can vary in length up to 72 characters.
- To send a workstation file from your default drive to replace a file on your VM/CMS A-disk:

  ```
  SEND pc.txt a:cmsfile script a (ASCII CRLF
  ```

  This command sends a workstation file named PC.TXT from your default drive to your VM/CMS A-disk in your host session named **a**. You do not need to name the workstation drive if the file you are sending is on the default drive. The file replaces a SCRIPT file named CMSFILE. The new CMSFILE has the same record length and format as the old CMSFILE.

  If you do not have a file called CMSFILE SCRIPT on your A-disk, PC.TXT is added to your A-disk as a new file called CMSFILE SCRIPT. The records in the file are 80 characters long and have fixed length.
- To send a binary workstation file from a drive other than your default drive:

```
SEND a:pc.exe c:cmsfile exebin b (recfm v
```

This command sends a workstation file named PC.EXE from a diskette in drive A to your VM/CMS B-disk in your host session named **c**. It is a new file, or it replaces a file named CMSFILE.

When transferring a binary file, you must specify a variable record format (**recfm v**), otherwise, blank characters are added to the file.

- To send a file from your hard disk and add it to the end of a file on your VM/CMS A-disk:

```
SEND c:pc.txt cmsfile script a (ASCII CRLF APPEND
```

This command sends a workstation file named PC.TXT from your hard disk to your host session. You do not need to name the host session if you are sending to the **a** session. The file is added to the end of a script file named CMSFILE on your VM/CMS A-disk.

- To send a file from a subdirectory on your hard disk to your VM/CMS A-disk:

```
SEND c:\sd1\pc.txt cmsfile script a (ASCII CRLF
```

This command sends a file named PC.TXT from subdirectory SD1 on your hard disk to your host session. It replaces a SCRIPT file named CMSFILE on your VM/CMS A-disk.

Using the VM/CMS RECEIVE Command

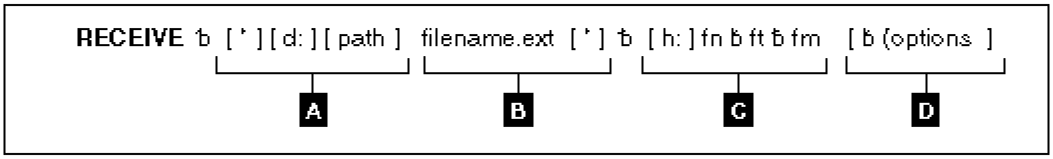
Use the following information when receiving a file from VM/CMS:

Figure 2: VM/CMS RECEIVE Command Syntax on page 71 shows the command and information you must provide. Enter it as shown (including parentheses), except that you can use either uppercase or lowercase letters.

Note:

1. ƀƀ means to insert a space. There must *not* be a space between **h:** and **fn**.

Figure 2. VM/CMS RECEIVE Command Syntax


RECEIVE ƀ ['] [d:] [path] filename.ext ['] ƀ [h:] fn ƀ ft ƀ fm [ƀ (options]
A
B
C
D

A

The workstation drive and path of the file to be received.

B

The name of the workstation file to be received.

C

Host session specifications for the file to be received from the host.

`h:`

The short name of the session (which can be omitted if it is **a**)

`fn`

File name

`ft`

File type

`fm`

File mode

D

Optional changes made to the file during transfer. More than one option can be selected. Valid options are:

- APPEND
- ASCII
- BLANK
- CLEAR
- CRLF
- NOCLEAR
- PROGRESS
- QUIET
- TIME(`n`)

The parts of the VM/CMS RECEIVE command are:

RECEIVE

The command.

`d:`

The name of the diskette or hard disk drive on which the file is to be received.

`path`

The path indicating the directory to which the file is to be stored.

`filename.ext`

The name of the workstation file, including the extension. Use a new name or one that already exists. If you use a new name, the file that you receive is added to your diskette or hard disk. If you use the name of an existing file, the file that you receive either replaces or supplements the existing file. (Refer to the APPEND option.)

`h:`

The short name of the host session from which you want to get the file. The default is `a:`.

`fn ft fm`

The name of the file you want to receive from your VM/CMS disk. The file name `fn` is required.

(`options`

These options can be specified:

APPEND

Specifies that the file being received is to be added to the end of an existing file. Omit this part of the VM/CMS file that is received to replace an existing file.

ASCII

Performs the following:

- Converts EBCDIC codes to 1-byte workstation codes.

BLANK

This option is valid with the CRLF option. Use it to retain BLANK (x'40') at the end of each line.

CRLF

Specifies the carriage return and line feed codes. You need ASCII and CRLF for text or source files that you want to view or edit, such as SCRIPT files. You do not need them for binary files, such as programs.

CLEAR

Clears the workstation window at the beginning of the file transfer.

NOCLEAR

Suppresses the sending of a Clear command at the beginning of the file transfer.

PROGRESS

Shows a message indicating that the file transfer is in progress or has ended. Such messages do not show the current transferred bytes.

QUIET

Does not show any messages.

TIME(`n`)

Specifies the length of time, in units of 30 seconds, that the program waits for a response from the host before it sends an error message. The value `n` is an integer value in the range from 0 through 2184. If you specify 0, timeout is not set. The default is 1. To avoid a premature error message, specify an adequate value. In cases of large packet sizes, of large block sizes, or for slow communication lines (such as and COM port), *5* (150 seconds) is recommended. There should be no blank spaces between TIME and *(n)*.

Command Syntax for Receiving Files from VM/CMS

The following examples show the command syntax you can use to receive files from a VM/CMS host. The parameters of the RECEIVE command can be combined into a single set of parentheses.

- To receive a file from your VM/CMS A-disk to your default drive for a workstationsession:

  ```
  RECEIVE pc.txt a:cmsfile script a (ASCII CRLF
  ```

  This command sends a SCRIPT file CMSFILE from your VM/CMS A-disk in a host session named A to your workstation session. It adds the file to your default drive (diskette or hard disk) with the name PC.TXT.
- To receive a file from your VM/CMS B-disk and replace a file on a drive other than your default:

  ```
  RECEIVE a:pc.txt a:cmsfile script b (ASCII CRLF
  ```

  This command sends a SCRIPT file named CMSFILE SCRIPT from your VM/CMS B-disk in a host session named A to a drive other than the default for your PC session. It replaces a file named PC.TXT on a diskette in drive A.
- To receive a file from your VM/CMS A-disk and add it to the end of a file on your hard disk:

  ```
  RECEIVE c:pc.txt a:cmsfile script a (ASCII CRLF APPEND
  ```

  This command sends a SCRIPT file named CMSFILE SCRIPT from your VM/CMS A-disk in a host session named A to your workstation session. It adds the contents of CMSFILE to the end of a file named PC.TXT on your hard disk.
- To receive a file from your VM/CMS A-disk and place it in a subdirectory on your default drive:

  ```
  RECEIVE \sd1\pc.txt a:cmsfile script a (ASCII CRLF
  ```

  This command sends a SCRIPT file named CMSFILE SCRIPT from your VM/CMS A-disk to your default drive. It creates or replaces a file named PC.TXT in a subdirectory named \SD1.

Using the MVS/TSO SEND Command

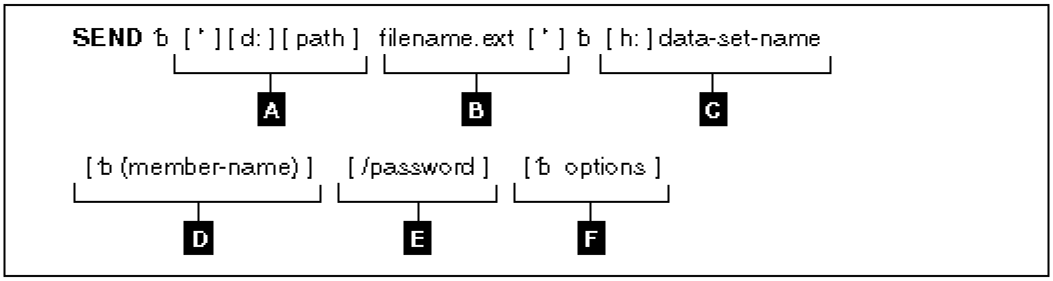
Use the following information when entering the SEND command to the MVS/TSO host:

Figure 3: MVS/TSO SEND Command Syntax on page 75 shows the command and information you must provide. Enter text as shown (including parentheses), except that you can use either uppercase or lowercase letters.

Note:

1. ƀƀ means to insert a space. There must ***not*** be a space between **h:** and **fn**.

Figure 3. MVS/TSO SEND Command Syntax



```
SEND ƀ [ ' ] [ d: ] [ path ]  filename.ext  [ ' ]  ƀ  [ h: ] data-set-name
[ ƀ (member-name) ]  [ /password ]  [ ƀ options ]
```

A

The workstation drive and path of the file to send.

B

The name of the workstation file to send.

C

The short name of the host session and the data set name of the file to send.

D

The member name if the file is in a partitioned data set.

E

The password of the data set if it has one.

F

Optional changes made to the file during transfer. More than one option can be specified. Valid options are:

- APPEND
- ASCII
- BLKSIZE(n)
- CLEAR
- CRLF
- LRECL(n)
- NOCLEAR

  You must use the NOCLEAR option when you are transferring files while in ISPF command mode on the host.
- PROGRESS

- QUIET
- RECFM(`x`)
- SPACE(`n[,n1]`) `unit`
- TIME(`n`)

The parts of the MVS/TSO SEND command are:

SEND

The command.

`d:`

The name of the diskette or hard disk drive where the file is located.

`path`

The path indicating the directory where the file is located.

`filename.ext`

The name of the file to be sent. Include the extension if the file has one.

`h:`

The name of the MVS/TSO host session to which you want to send the file. You can omit this name if you have only one host. If you have more than one host, this is the short name of the MVS/TSO host session. The default short name is A.

`data-set-name`

The data set name that the file you send is to have on your MVS/TSO volume; this name is required. Enclose the data set name with the member name in single quotation marks if you are using a fully qualified data set name.

This option creates a new name or uses a data set name already on your TSO volume. If you use a new name, the file that you send is added to your MVS/TSO volume. If you use the name of an existing data set, the file you send either replaces or supplements the existing data set. Refer to the APPEND option.

(`member-name`**)**

The member name if the file is to be put into a partitioned data set. If you use member-name, you cannot use LRECL(`n`), BLKSIZE(`n`), RECFM(`x`), and SPACE(`n, [n1]`) `unit`.

Note: If someone else is using the partitioned data set, you cannot send a file to your MVS/TSO host.

/`password`

The password of the data set, if the data set has a password.

`options`

These options can be specified:

APPEND

Specifies that the file being sent is added to the end of an existing MVS/TSO data set. Omit this option if you want the file to replace an existing MVS/TSO data set. You cannot use LRECL(`n`), RECFM(`x`), SPACE(`n[,n1]`) `unit`, or BLKSIZE(`n`) options if you use the APPEND option.

Note: This option is not valid when sending data to a member of a partitioned data set.

ASCII

Performs the following:

- Converts 1-byte workstation codes to EBCDIC codes.

BLKSIZE(`n`)

Specifies the size of the blocks of data in a new data set on your MVS/TSO volume. This part is optional. To set the block size for a new data set, replace `n` with the new size. If you omit this option, the block size is determined in the following manner:

- If the record format is variable, the block size is 6233.
- If the record format is fixed, the block size is the largest multiple of the record length that is less than 6233:

```
BLKSIZE = LRECL * (6233/LRECL)
```

If you use the (member-name) or APPEND option, do not use this option.

CLEAR

Clears the workstation window at the beginning of the file transfer.

CRLF

Specifies the global use of carriage return and line feed codes. You need to specify ASCII and CRLF options for sending text or source files that you want to view or edit, such as SCRIPT files. You do not need them for binary files.

LRECL(`n`)

Specifies the record length for a new data set on your MVS/TSO volume, where `n` is a whole number from 1 through 32760 representing the number of characters per record. If you want to set the record length for a new data set, replace `n` with the new length. If you omit this option, the record length is set to 80 for fixed-length records and to 255 for variable-length records. If you use the (member-name) or APPEND options, do not use this option.

NOCLEAR

Suppresses the sending of a Clear command at the beginning of the file transfer. This option is required for ISPF command mode.

PROGRESS

Shows a message indicating that the file transfer is in progress or has ended. Such messages do not show the current transferred bytes.

QUIET

Does not show any messages.

RECFM(`x`)

Specifies the record format for a new data set on your MVS/TSO volume, where `x` = V, F, or U. For variable-, fixed- or undefined-length records in the data set, replace the `x` with V, F, or U, respectively.

If you omit this option, the record format of the host data set is determined by the setting of the CRLF parameter: if you specify CRLF, the data set has variable-length records; if you do not specify CRLF, it has fixed-length records. If you use the (member-name) or APPEND options, do not use this option.

SPACE(`n[,n1]`) `unit`

Specifies an amount of space to be set aside for a new data set on your MVS/TSO volume. To set aside a certain number of blocks, tracks, or cylinders for the new data set:

- Provide `unit` as the type of space you want (AVBLOCK, TRACKS, or CYLINDERS).
- Give `n` as the amount of space that you want the data set to occupy (in the unit of measure you select).
- If the data set needs more space than you ask for with `n`, give `n,n1` where `n1` is the size of additional space to be used only when necessary.

These values are similar to the values on the ALLOCATE command of MVS/TSO.

If you omit this option, you get space for one block. The length of the block is set by the BLKSIZE(`n`) or LRECL(`n`) options. If you use the (`member-name`) or APPEND options, do not use this option.

TIME(`n`)

Specifies the length of time, in units of 30 seconds, that the program waits for a response from the host before it sends an error message. The value `n` is an integer value in the range from 0 through 2184. If you specify 0, timeout is not set. The default is 1. To avoid a premature error message, specify an adequate value. In cases of large packet sizes, of large block sizes, or for slow communication lines (such as and COM port), *5* (150 seconds) is recommended. There should be no blank spaces between TIME and *(n)*.

Command Syntax for Sending Files to MVS/TSO

The following examples show the command syntax you can use to send files from your workstation to an MVS/TSO host:

- To send a file from your default drive to replace a file on the MVS/TSO host:

  ```
  SEND pc.txt g:ds.script ASCII CRLF
  ```

  This command sends a workstation file named PC.TXT from your default drive to your MVS/TSO host in a host session named G. It creates or replaces a data set named DS.SCRIPT on your MVS/TSO volume.
- To send a file from a drive other than the default to your MVS/TSO host:

  ```
  SEND a:pc.txt g:ds.script ASCII CRLF
  ```

  This command sends a workstation file named PC.TXT from a diskette in drive A to your MVS/TSO host in a host session named G. It replaces a data set named DS.SCRIPT on your MVS/TSO volume.
- To send a file from your default drive to your MVS/TSO host and add it to the end of an MVS/TSO data set:

  ```
  SEND a:pc.txt g:ds.script ASCII CRLF APPEND
  ```

  This command sends a workstation file named PC.TXT from a diskette in drive A to your MVS/TSO host in a host session named G. It adds the file to the end of a data set named DS.SCRIPT on your MVS/TSO volume.
- To send a file to your MVS/TSO host and add it to the end of a data set that has a password:

  ```
  SEND a:pc.txt g:ds.script/odyssey8 ASCII CRLF APPEND
  ```

  This command sends a workstation file named PC.TXT from a diskette in drive A to your MVS/TSO host in a host session named G. It adds the file to the end of a data set named DS.SCRIPT on your MVS/TSO volume. This data set has a password of odyssey8.
- To send a file from a subdirectory on your hard disk to a partitioned data set on your MVS/TSO host:

  ```
  SEND c:\sd1\pc.txt g:ds.script (m1) ASCII CRLF
  ```

  This command sends a workstation file named PC.TXT from a subdirectory named \SD1 on your hard disk to your MVS/TSO host in a host session named G. It creates or replaces a member named M1 in a partitioned data set named DS.SCRIPT on your MVS/TSO volume.
- To send a file from your default drive and add it as a new data set on your MVS/TSO volume:

  ```
  SEND pc.txt g:ds.script/aeneid20 ASCII CRLF LRECL(132)
       BLKSIZE(132) RECFM(V) SPACE(20,10) TRACKS
  ```

  This command sends a workstation file named PC.TXT from your default drive to your MVS/TSO host. It adds the file as a new data set named DS.SCRIPT on your MVS/TSO volume. A password of aeneid20 is assigned. The records in the data set can vary in length up to 132 characters. Data blocks are the same length as the records. Twenty tracks are set aside for this data set. If more tracks are needed, they are added in groups of 10.

Using the MVS/TSO RECEIVE Command

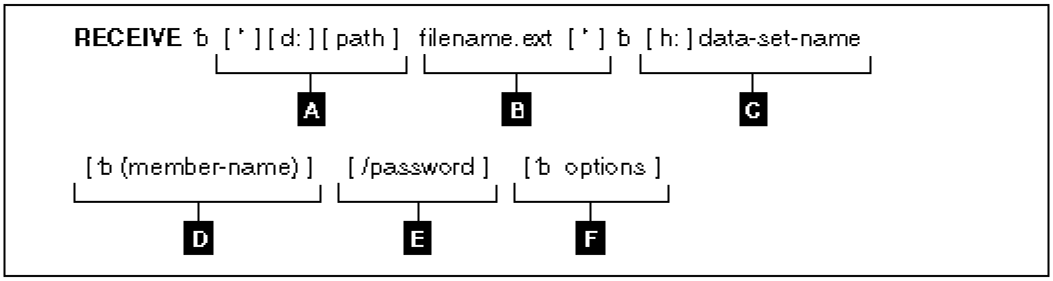
Use the following information when receiving a file from MVS/TSO:

Figure 4: MVS/TSO RECEIVE Command Syntax on page 80 shows the command and information you must provide. Enter it as shown (including parentheses), except that you can use either uppercase or lowercase letters.

Note:

1. ƀ means to insert a space. There must *not* be a space between **h:** and **fn**.

Figure 4. MVS/TSO RECEIVE Command Syntax


RECEIVE ƀ ['] [d:] [path] filename.ext ['] ƀ [h:] data-set-name
A
B
C
[ƀ (member-name)]
[/password]
[ƀ options]
D
E
F

A

The workstation drive and path to the directory where the file is to be stored.

B

The name of the workstation file to receive.

C

The short name of the host session, and the data set name of the file you are receiving.

D

The member name if the file is put in a partitioned data set.

E

The password of the data set, if any.

F

Optional changes made to the file during transfer. More than one option can be specified. Valid options are:

- APPEND
- ASCII

- BLANK
- CRLF
- PROGRESS
- QUIET
- TIME(`n`)

The parts of the MVS/TSO RECEIVE commands are:

RECEIVE

The command.

`d:`

The name of the diskette or hard disk drive where the file is to be located. Use A:, B:, C:, D: through Z:. This part is optional if the file is received on the current drive.

`path`

The subdirectory where you want the data set located. This part is optional.

`filename.ext`

The name the file is to have on your diskette or hard disk. Creates a new name or uses a name that is already on your diskette or hard disk.

If you use a new name, the data set that you receive is added to your diskette or hard disk. If you use the name of an existing file, the data set that you receive either replaces or supplements the existing file. (Refer to the APPEND option on page APPEND on page 82.)

`h:`

The short name of the MVS/TSO session where the data set is located. If you have only one host, this part is optional. Use this option if you have more than one host. The default short name is A.

`data-set-name`

The name of the data set or the partitioned data set that contains the member you want to send to your workstation session. You must use the qualified name. Enclose the data set name with the member name in single quotation marks if you are using a fully qualified data set name.

`(member-name)`

The member name of a partitioned data set to send to your workstation session. This part is optional. Use it only if the data set is a member of a partitioned data set.

`/password`

The password of the data set. Use it only if the data set has a password.

(`options`

These options can be specified:

APPEND

Adds the data set to the end of an existing file. Omit this part if you want the MVS/TSO data set to replace an existing workstation file.

ASCII

Performs the following:

- Converts EBCDIC codes to 1-byte workstation codes.

BLANK

This option is valid with the option CRLF; it retains BLANK (hex 40) at the end of each line.

CRLF

Specifies the use of carriage return and line feed codes. You need ASCII and CRLF for text or source files that you want to view or edit, such as SCRIPT files. You do not need them for binary files.

PROGRESS

Shows a message indicating that the file transfer is in progress or has ended. Such messages do not show the current transferred bytes.

QUIET

Does not show any messages.

TIME(`n`)

Specifies the length of time, in units of 30 seconds, the program waits for a response from the host before it sends an error message. Replace `n` with an integer value in the range from 0 through 2184. If you specify 0, timeout is not set. The default is 1. To avoid a premature error message, specify an adequate value. In cases of large packet sizes, of large block sizes, or for slow communication lines (such as and COM port), *5* (150 seconds) is recommended. There should be no blank spaces between TIME and (`n`).

Command Syntax for Receiving Files from MVS/TSO

The following examples show the command syntax you can use to receive files from your MVS/TSO host to your workstation:

- To receive a data set from an MVS/TSO host to the default drive for your workstation session:

  ```
  RECEIVE pc.txt g:ds.script ASCII CRLF
  ```

  This command sends a data set named DS.SCRIPT from your MVS/TSO volume in a host session named G to your OS/2 session. It creates or replaces the file on the default drive with the name PC.TXT.
- To receive a data set from an MVS/TSO host to a drive other than your default drive:

  ```
  RECEIVE A:pc.txt g:ds.script ASCII CRLF
  ```

This command sends a data set named DS.SCRIPT from your MVS/TSO volume in a host session named G. It replaces a file named PC.TXT on a diskette in drive A.

- To receive a data set from an MVS/TSO host and add it to a workstation file:

```
RECEIVE a:pc.txt g:ds.script ASCII CRLF APPEND
```

This command sends a data set named DS.SCRIPT from your MVS/TSO volume in a host session named G. It adds the data set to the end of a file named PC.TXT on the diskette in drive A.

- To receive a data set from an MVS/TSO host and place it in a subdirectory on your hard disk:

```
RECEIVE c:\sd1\pc.txt ds.script ASCII CRLF
```

This command sends a data set named DS.SCRIPT from your MVS/TSO volume in a host session named G. It creates or replaces a file named PC.TXT in a subdirectory named \SD1 on your hard disk.

- To receive a data set that has a password from an MVS/TSO host to your default drive:

```
RECEIVE A:pc.txt g:ds.script/odyssey8 ASCII CRLF APPEND
```

This command sends a data set named DS.SCRIPT from your MVS/TSO volume in a host session named G. The data set has the password odyssey8. The data set is added to the end of a file named PC.TXT on the diskette in drive A.

- To receive a member of a partitioned data set from an MVS/TSO host to your DOS session:

```
RECEIVE c:\sd1\pc.txt g:ds.script (m1) ASCII CRLF
```

This command sends a member named M1 from a partitioned data set named DS.SCRIPT in a host session named G. The member is placed on your hard disk in a subdirectory named \SD1. It replaces or creates a file named PC.TXT.

- To receive a member of a partitioned data set that has a password to your Windows session:

```
RECEIVE a:pc.txt g:ds.script (m2)/ili1 ASCII CRLF APPEND
```

This command sends a member named M2 from a partitioned data set named DS.SCRIPT in a host session named G. The data set has a password of ili1. The member is added to a file named PC.TXT on the diskette in drive A.

Using the CICS SEND Command

Please note the differences between the Z and I Emulator for Windows GUI and Command Line syntaxes. These two syntaxes are not interchangeable.

Using CICS SEND with the Z and I Emulator for Windows GUI

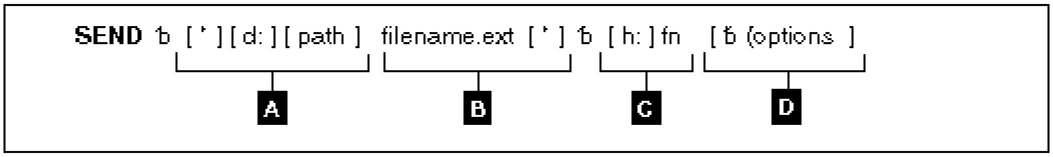
Use the following information when sending a file to CICS using the Z and I Emulator for Windows graphical user interface (GUI):

Figure 5: CICS SEND Z and I Emulator for Windows GUI Syntax on page 84 shows the command and information you must provide. Enter it as shown (including parentheses), except that you can use either uppercase or lowercase letters.

Note:

1. ƀ means to insert a space. There must *not* be a space between **h:** and **fn**.

Figure 5. CICS SEND Z and I Emulator for Windows GUI Syntax


SEND ƀ ['] [d:] [path] filename.ext ['] ƀ [h:] fn [ƀ (options]
A
B
C
D

A

The workstation drive and path of the file to send.

B

The name of the workstation file to send.

C

The short name of the host session, and the host file name of the file to send.

D

Optional changes made to the file during transfer. More than one option can be specified. Valid options are:

- ASCII
- BINARY (for SBCS sessions)
- CLEAR
- CRLF
- NOCLEAR
- NOCRLF (for SBCS sessions)
- PROGRESS
- QUIET
- TIME(n)

Note: For SBCS sessions, the default options are ASCII and CRLF.

Using CICS SEND with the Z and I Emulator for Windows Command Line

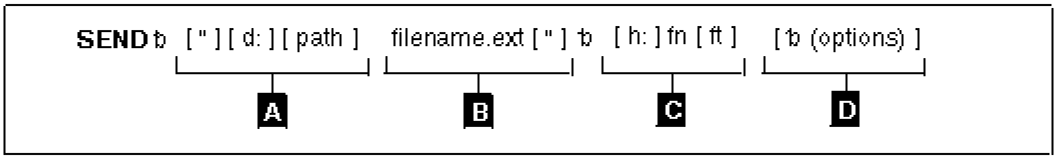
Use the following information when sending a file to CICS using the Z and I Emulator for Windows command line:

Figure 6: CICS SEND Command Line Syntax on page 85 shows the command and information you must provide. Enter it as shown (including parentheses), except that you can use either uppercase or lowercase letters.

Note:

1. ƀ ƀ means to insert a space. There must *not* be a space between **h:** and **fn**.

Figure 6. CICS SEND Command Line Syntax


SEND ƀ ["] [d:] [path] filename.ext ["] ƀ [h:] fn [ft] [ƀ (options)]
A
B
C
D

A

The workstation drive and path of the file to send.

B

The name of the workstation file to send.

C

The short name of the host session (h:), the host file name (fn), and the file type (ft).

D

Optional changes made to the file during transfer. More than one option can be specified. Valid options are:

- ASCII
- BINARY (for SBCS sessions)
- CLEAR
- CRLF
- NOCLEAR
- NOCRLF (for SBCS sessions)
- PROGRESS
- QUIET
- TIME(n)

Note: For SBCS sessions, the default options are ASCII and CRLF.

CICS SEND Command Description and Options

The parts of the CICS SEND command are:

SEND

The command.

`d:`

The name of the diskette or hard disk drive where the file is located.

`path`

The path to the subdirectory that the file is in.

`filename.ext`

The name of the file to be sent, including the extension.

`h:`

The short name of the host session where you want to send the file. If you have only one host, this part is optional. The default is session **A**.

`fn`

The name the file is to have on your CICS disk. You must specify the file name. You can create a new name or use a name that is already on the disk.

ft

The type of file in CICS. For use only with command line syntax, see Using CICS SEND with the Z and I Emulator for Windows Command Line on page 84.

(`options`

These options can be specified:

ASCII

Performs the following:

- Converts 1-byte workstation codes to EBCDIC codes.

The default is ASCII CRLF. You need these control terms for text or source files that you want to view or edit, such as SCRIPT files. You do not need them for binary files.

Note:

1. CRLF and NOCRLF are mutually exclusive options.
2. BINARY and ASCII are mutually exclusive options.
3. The assumed defaults, if the optional parameters are omitted, are CRLF ASCII.

BINARY

Specifies that the data in the file is binary data. The data can be encrypted, compiled programs, or other data. It is not translated by the host file transfer program but copied unaltered into a temporary storage queue.

This option is valid for SBCS sessions only.

CLEAR

Clears the workstation window at the beginning of the file transfer.

CRLF

Specifies carriage return and line feed codes in the text file.

NOCLEAR

Suppresses the sending of a Clear command at the beginning of file transfer. This option is required for ISPF command mode.

NOCRLF

Specifies that the PC file does not consist of logical records delimited by carriage return and line feed characters. No concatenation or splitting of records is performed by the CICS file transfer program.

The file is written into a temporary storage using one item on the queue to represent each inbound data buffer. The items on the CICS temporary storage queue can be of different lengths, but none can be more than 32767 characters.

This option is valid for SBCS sessions only.

PROGRESS

Shows a message indicating that the file transfer is in progress or has ended. Such messages do not show the current transferred bytes.

QUIET

Does not show any messages.

TIME(`n`)

Specifies the length of time, in units of 30 seconds, the program waits for a response from the host before it sends an error message. Replace `n` with an integer value in the range from 0 through 2184. If you specify 0, timeout is not set. The default is 1. To avoid a premature error message, specify an adequate value. In cases of large packet sizes, large block sizes, or for slow communication lines (such as and COM port), 5 (150 seconds) is recommended. There should be no blank spaces between TIME and (n).

Command Syntax for Sending Files to CICS

The following examples show the command syntax you can use to send files from your workstation to your CICS host.

- To send a workstation file from your default drive and add it as a new file on your CICS host:

  ```
  SEND pc.txt a:cicsfile (ASCII CRLF)
  ```

  Note: Enter the complete CICS SEND command on one line.

  This command sends a workstation file named PC.TXT from your default drive on your workstation to your host session A. You do not need to provide the workstation drive name if the file you are sending is on the current drive. The command creates a new file named CICSFILE.
- To send a basic workstation file from a drive other than your default to replace a file on your CICS host:

  ```
  SEND a:myprog.exe a:basprog
  ```

  This command sends a workstation file named MYPROG.EXE from a diskette in drive A to your CICS host in your host session named A. It is written to a file named BASPROG, replacing any existing file by that name in host session A.

Using the CICS RECEIVE Command

Please note the differences between the Z and I Emulator for Windows GUI and Command Line syntaxes. These two syntaxes are not interchangeable.

Using CICS RECEIVE with the Z and I Emulator for Windows GUI

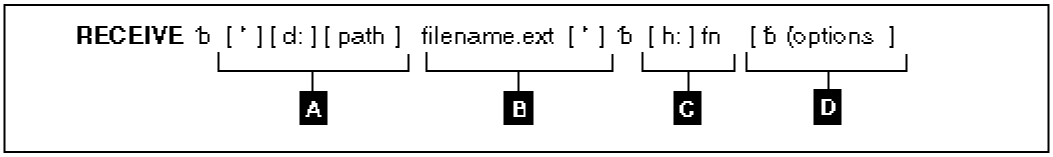
Use the following information when receiving files from CICS using the Z and I Emulator for Windows GUI:

Figure 7: CICS RECEIVE Z and I Emulator for Windows GUI Syntax on page 89 shows the command and information you must provide. Enter it as shown (including parentheses), except that you can use either uppercase or lowercase.

Note:

1. ƀ means to insert a space. There must *not* be a space between **h:** and **fn**.

Figure 7. CICS RECEIVE Z and I Emulator for Windows GUI Syntax


RECEIVE ƀ ['] [d:] [path] filename.ext ['] ƀ [h:] fn [ƀ (options]
A
B
C
D

A

The workstation drive and path where the file is to be received.

B

The name of the workstation file.

C

The short name of the host session (h:) from which you are receiving the file, and the host file name (fn).

D

Optional changes made to the file during transfer. More than one option can be specified. Valid options are:

- ASCII
- BINARY (for SBCS sessions)
- BLANK
- CLEAR
- CRLF
- NOCLEAR
- NOCRLF (for SBCS sessions)
- PROGRESS
- QUIET
- TIME(n)

Note: The default options for SBCS sessions are ASCII and CRLF.

Using CICS RECEIVE with the Z and I Emulator for Windows Command Line

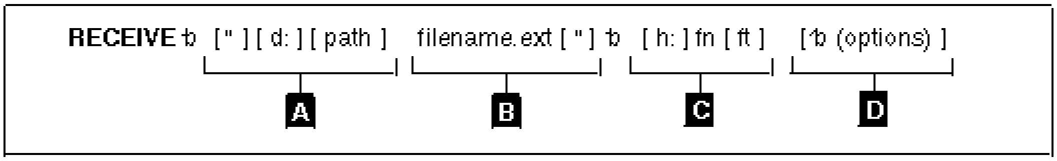
Use the following information when receiving files from CICS using the Z and I Emulator for Windows command line:

Figure 8: CICS RECEIVE Command Syntax on page 90 shows the command and information you must provide. Enter it as shown (including parentheses), except that you can use either uppercase or lowercase.

Note:

1. ƀ means to insert a space. There must *not* be a space between **h:** and **fn**.

Figure 8. CICS RECEIVE Command Syntax



```
RECEIVE ƀ [" ] [ d: ] [ path ] filename.ext [ " ] ƀ [ h: ] fn [ ft ] [ƀ (options) ]
```

A

The workstation drive and path where the file is to be received.

B

The name of the workstation file.

C

The short name of the host session (h:) from which you are receiving the file, the host file name (fn), and the file type (ft).

D

Optional changes made to the file during transfer. More than one option can be specified. Valid options are:

- ASCII
- BINARY (for SBCS sessions)
- BLANK
- CLEAR
- CRLF
- NOCLEAR
- NOCRLF (for SBCS sessions)
- PROGRESS
- QUIET
- TIME(n)

Note: The default options for SBCS sessions are ASCII and CRLF.

CICS RECEIVE Description and Options

The parts of the CICS RECEIVE command are:

RECEIVE

The command.

`d:`

The name of the diskette or hard disk drive where the file is to be received.

`path`

The path to the subdirectory where the file is to be located.

`filename.ext`

The name of the workstation file, including the extension. You can create a new name or use a name that is already on your workstation diskette or hard disk. If you use a new name, the file that you receive is added to your diskette or hard disk. If you use the name of an existing file, the file that you receive either replaces or supplements the existing file. Refer to the APPEND option.

`h:`

The short name of the CICS session where the data set is located. If you have only one host, this part is optional. The default session is A.

`fn`

The name of the file you want to receive from your CICS host.

ft

The type of file in CICS. For use only with command line syntax, see Using CICS RECEIVE with the Z and I Emulator for Windows Command Line on page 89.

(`options`

These options can be specified:

ASCII

Performs the following:

- Converts EBCDIC codes to 1-byte workstation codes.

The default is ASCII CRLF. You need ASCII and CRLF control terms for text or source files that you want to view or edit, such as SCRIPT files. You do not need them for binary files.

Note:

1. CRLF and NOCRLF are mutually exclusive options.
2. BINARY and ASCII are mutually exclusive options.
3. The assumed defaults, if the optional parameters are omitted, are CRLF ASCII.

BINARY

The data in the file is binary data. The data can be encrypted data, compiled programs, or other data. It is not translated by the host file transfer program but is copied without changes into the workstation file.

This option is valid for SBCS sessions only.

BLANK

This option is valid only when used with the CRLF option; it retains BLANK (hex 40) at the end of each line.

CLEAR

Clears the workstation window at the beginning of the file transfer.

CRLF

Specifies the use of the carriage return and line feed codes.

NOCLEAR

Suppresses the sending of a Clear command at the beginning of file transfer. This option is required for ISPF command mode.

NOCRLF

Specifies that the host computer file does not consist of logical records. The items in the temporary storage queue are sent in order and concatenated in your workstation into a single string of data.

This option is valid for SBCS sessions only.

PROGRESS

Shows a message indicating that the file transfer is in progress or has ended. Such messages do not show the current transferred bytes.

QUIET

Does not show any messages.

TIME(`n`)

Specifies the length of time, in units of 30 seconds, the program waits for a response from the host before it sends an error message. Replace `n` with an integer value in the range from 0 through 2184. If you specify 0, timeout is not set. The default is 1. To avoid a premature error message, specify an adequate value. In cases of large packet sizes, of large block sizes, or for slow communication lines (such as and COM port), 5 (150 seconds) is recommended. There should be no blank spaces between TIME and (n).

Command Syntax for Receiving Files from CICS

The following examples show the command syntax you can use to receive files from your CICS host to your workstation.

- To receive a file from your CICS host to your default drive for a workstation session:

  ```
  RECEIVE pc.txt A:cicsfile (ASCII CRLF)
  ```

  This command sends a file named CICSFILE from your CICS host in session A to your workstation session. It adds the file to your default drive (diskette or hard disk) with the name PC.TXT.
- To receive a basic file from your CICS host and replace a file on a drive other than your default:

  ```
  RECEIVE a:myprog.exe a:myprog
  ```

  This command sends a file named MYPROG from your CICS host in session A to a drive other than the default for your workstation session. It replaces a file named MYPROG.EXE on a diskette in drive A.

Configuring File-Transfer Code Translation

When you transfer a file between the host and the workstation using the ASCII option, the host-system file-transfer program performs translation from EBCDIC to ASCII or vice versa, according to the host and PC code pages specified during PC/3270 configuration. However, you might want to use different translation from that supplied. For the details, refer to Transferring Files on page 54.

Part III. Using Z and I Emulator for Windows 5250

Chapter 8. Considerations for Using PC400 Sessions

This chapter contains hints and tips for using PC400 sessions. Supplementary information other than the items described in this book is included in the Readme HTML file in the Z and I Emulator for Windows directory.

Scroll Bar

When you click **Font** from the **Appearance** menu in the host session window and select **Fixed Size** from the Select Display Font window, the entire operator information area might not appear on the screen; the session-window size is restricted to be smaller than the screen size. If you specify **With Scroll Bar**, the OIA will not scroll.

Print Processing

Following are some additional considerations when printing with PC400.

Printing Bar Codes

This function requires OS/400® Version 4.2 or i5/OS™.

CPI/LPI of Device Fonts

If the printer driver cannot print with device fonts associated with the user-specified CPI/LPI, the print output can be generated with incorrect CPI/LPI values.

PCSERR999 Error Messages

Message `PCSERR999 - Z and I Emulator for Windows internal error:`*`module-name - xxxx`* might appear if there is insufficient memory. If any print jobs are queued in the print manager, delete those print jobs.

Disconnect in Testrequest to iSeries, eServer i5, or System i5 on Telnet 5250

Executing a Testrequest function when connected to an iSeries™, eServer™ i5, or System i5™ might cause the session to be disconnected. If you experience this problem, make sure that OS/400® APAR MA15053 has been applied on the iSeries™, eServer™ i5, or System i5™.

iSeries, eServer i5, or System i5 Host Print Problem

If you attempt to use the host print function (mapped to CTRL-Pause by default) while viewing a spooled print file, the ends of some of the lines might be wrapped incorrectly in the second generated spool file. This problem occurs with both 24X80 and 27X132 display modes. This problem has been fixed by a PTF on OS/400®. The APAR number is SA57195 and is available on PTF MF13596 for OS/400® V3R1.

Printable Area

Depending on the printer driver used, it might not be possible to use the entire surface of the paper for printing.

If the printing position is beyond the printable area, the page is automatically changed. When using a printer driver that allows you to set the margins, specify the minimum margins, thus maximizing the printable area.

PDT Mode

Printing using a PDT file is restricted as follows:

- Only the fonts specific to the printer being used are supported.
- Graphics are printed using the Windows® printer driver selected in **Printer Setting**, regardless of the PDT mode.
- Postscript printers are not supported. There are no PDF files for Postscript printers.

Setting the Code Page

The host code page, which is set in the Configuration panel, is used as the default. Use the Set Initial Condition (SIC) command to set the host code page.

You can change the code page by using Set GCGID Through GCID (SCG) command or Set CGCS Through Local ID (SCGL) command. The same code pages for the display session are available.

Chapter 9. Data Transfer for PC400

This chapter explains file-description files and data conversions for the data transfer function. .

Data Transfer Function Overview

PC400 can transfer data between the host and workstation. The data transfer function can be invoked manually by clicking the **Data Transfer** icon. The Data Transfer application is automatically invoked from a 5250 session when you click **Send File to Host** from the **Actions** menu and **Receive File from Host** from the **Actions** menu. You can change this default to invoke normal file transfer functions; to do so, click **Preferences → Transfer** from the **Edit** menu, then click the **Data Transfer** radio button on the property page with the **General** tab.

Transferring data, described in this chapter, is quite different from transferring files, which is described in File Transfer for PC400 on page 191. The main differences are listed in the following table.

Table 17. Data Transfer Summary

Type of Transfer	**Products required on an iSeries™, eServer™ i5, or System i5™**	**Access Method**	**Sending and receiving unit**	**Type of connection to an iSeries™, eServer™ i5, or System i5™**
File Transfer	Z and I Emulator for Windows Tools (PCT/400 see Transferring Files on page 183)	• Transfer menu in the session window • EHLLAPI application that invokes File Transfer • DDE application that invokes File Transfer • Playing a macro that invokes File Transfer • Clicking the Send or Receive button on the tool bar	Entire file	Display session
Data Transfer	PC Support/400 V2R2 or V2R3, OS/400® V3R1 or later, or i5/OS™[1]	Data Transfer icon or File Transfer selections from Actions menu	Field, record, or file in a database	• TCP/IP

[1] OS/400® and i5/OS™ provide the host transaction program for Data Transfer.

Long Password Support

The Z and I Emulator for Windows Data Transfer utility supports 128-character case-sensitive passwords, when connecting to an iSeries™, eServer™ i5, or System i5™ host running i5/OS™ or OS/400®, V5R1 or later. This functionality is determined by the OS/400® or i5/OS™ system value QPWDLVL. Refer to the *iSeries Security Reference* (SC41-5302) for details.

Transferring Files from an iSeries, eServer i5, or System i5 System to a Workstation

When using a workstation, you can retrieve and use data from the following file types on an iSeries™, eServer™ i5, or System i5™:

- Physical database
- Logical database
- Distributed data management (DDM)

When retrieving files, you can do the following:

- Control which records (and which fields within a record) are retrieved
- Control the ordering of records and the ordering of fields within the record
- Select a subset of the records
- Group records into summary records
- Join two or more files
- Specify formats and separators of date and time fields
- Specify the decimal separator character

You can specify the following output destinations:

- Display
- Disk
- Printer

Transferring Files from a Workstation to an iSeries, eServer i5, or System i5

The PC→iSeries™ Transfer function enables the transfer of data from a workstation to an iSeries™, eServer™ i5, or System i5™ physical file. Data can be transferred to any of the following destinations:

- Existing members in an existing iSeries™, eServer™ i5, or System i5™ physical file
- New members in an existing iSeries™, eServer™ i5, or System i5™ physical file
- New members in a new iSeries™, eServer™ i5, or System i5™ physical file

Note: Data cannot be transferred from a workstation file to an iSeries™, eServer™ i5, or System i5™ logical file.

Transferring Data to Existing Members in an Existing File

Note the following considerations when transferring data from a workstation to an existing iSeries™, eServer™ i5, or System i5™ member.

- When data is transferred to an existing member, data in that member is replaced with that transferred from a workstation.
- When iSeries™, eServer™ i5, or System i5™ members already contain data, a message appears, indicating that the data in the existing members will be replaced with the data that is about to be transferred.
- Consider the effect of returning data that was previously transferred from the iSeries™, eServer™ i5, or System i5™ (such as when an iSeries™, eServer™ i5, or System i5™ master file is updated on a workstation).

  For example, you can transfer only the field subset of an iSeries™, eServer™ i5, or System i5™ file by issuing a transfer request from the iSeries™, eServer™ i5, or System i5™ to a workstation. In this case, when returning data from the workstation to the iSeries™, eServer™ i5, or System i5™, only the subset included in that iSeries™, eServer™ i5, or System i5™ file can be transferred. Other fields that had been defined in the iSeries™, eServer™ i5, or System i5™ file but not transferred, are filled with blanks if they are character fields or, if they are numeric fields, with zeros or the values specified at file creation.

  Therefore, the data must be transferred to another iSeries™, eServer™ i5, or System i5™ file and the transferred data must be embedded in the iSeries™, eServer™ i5, or System i5™ file by running the iSeries™, eServer™ i5, or System i5™ application program. Follow this procedure to control the update processing for an iSeries™, eServer™ i5, or System i5™ master file.

  To prevent users from transferring data to a certain iSeries™, eServer™ i5, or System i5™ file, check that the authority level for that file is defined correctly.

Transferring Data to New Members in an Existing File

You can transfer the data in a workstation file to new members in an existing iSeries™, eServer™ i5, or System i5™ file. The transfer function automatically creates these members in the specified file in the specified library. New members are created according to the file description in the existing file.

Be particularly careful when only the field subset of the iSeries™, eServer™ i5, or System i5™ file can be transferred from the iSeries™, eServer™ i5, or System i5™ to a workstation by the previous transfer request. When data is returned to the iSeries™, eServer™ i5, or System i5™, new members can receive only the subset defined in that iSeries™, eServer™ i5, or System i5™ file. Other character fields that are defined, but not transferred are filled with blanks. Numeric fields are filled with zeros or the valued specified at file creation. The date, time, and time-stamp fields use iSeries™, eServer™ i5, or System i5™ default values.

Transferring Data to New Members in a New File

By using a transfer request from a workstation to the iSeries™, eServer™ i5, or System i5™ system, you can transfer data to new members in a new iSeries™, eServer™ i5, or System i5™ file. This is one of the safest transfer methods, because data already stored in theiSeries™, eServer™ i5, or System i5™ file is not replaced with that transferred from the workstation.

There are two ways of transferring data to new members in a new iSeries™, eServer™ i5, or System i5™ file. The method used depends on the data to be transferred.

- For data that is broken up into fields, correct conversion is achieved by transferring it in units of fields. Specify use of the workstation file-description file at data transfer. In addition, specify *data* as the type of the eServer™ i5 or iSeries™ file.

  When an iSeries™, eServer™ i5, or System i5™ file and its members are created, the transfer function must access the description of the format of each field to be transferred in the iSeries™, eServer™ i5, or System i5™ file. You can get this description, called a field-reference file, from the iSeries™, eServer™ i5, or System i5™ file. To create an iSeries™, eServer™ i5, or System i5™ file and its members, specify the name of this iSeries™, eServer™ i5, or System i5™ field reference file, as well as the parameters for the other files and members. Note that only the fields to be transferred are defined in a new file.
- For data consisting only of text or source statement records, it is not necessary to break up the records into fields. In addition, the workstation file-description file is not required to transfer data. In other words, an iSeries™, eServer™ i5, or System i5™ physical source file is created.

Transferring Data to an iSeries, eServer i5, or System i5 Data File and Source File

You can transfer data to the following two types of iSeries™, eServer™ i5, or System i5™ physical files.

Physical data file

The members of a physical data file can contain numeric and character data of any iSeries™, eServer™ i5, or System i5™ data type. To transfer data to a physical data file, use the workstation file-description file to define how data is stored in a workstation data file. Besides this definition, the file description of the iSeries™, eServer™ i5, or System i5™ file is required to ensure correct conversion of the data.

When data is transferred to an existing iSeries™, eServer™ i5, or System i5™ file, the file description becomes part of the iSeries™, eServer™ i5, or System i5™ file. When data is transferred to a new iSeries™, eServer™ i5, or System i5™ file, the file description is included in the iSeries™, eServer™ i5, or System i5™ field-reference file.

Physical source file

Normally, a physical source file stores no data. It contains only text or source statements, as follows:

- The first part (field) of a source file always contains numbers indicating the order.
- The second part (field) of a source file always contains the date on which the file was created.
- The third part (field) of a source file contains the text of the file. This part can contain data fields of character type or zoned type only. Physical source files provide the optimum means of transferring text or source statements with a workstation.

Note the following considerations when transferring data to and from an iSeries™, eServer™ i5, or System i5™ physical source file:

- To transfer text from the iSeries™, eServer™ i5, or System i5™ to a workstation, specify the name of the source file and members in **FROM**. Specify an asterisk (*) in **SELECT**. This informs the iSeries™, eServer™ i5, or System i5™ that only text is transferred from the source file, with the order number and date fields excluded.
- The iSeries™, eServer™ i5, or System i5™ text must be stored in the workstation code text file. Normally, a workstation text editing program can be used to manipulate this workstation code text file.
- Specify that the file-description file is not to be stored for that workstation file. Because text is assumed to be a record consisting only of character data, it is not necessary to define fields.
- To return text from a workstation file to an iSeries™, eServer™ i5, or System i5™ file, specify the type of the workstation file containing the text. This is almost always workstation code text. Specification of the file-description file is not required.
- To create a new iSeries™, eServer™ i5, or System i5™ file and its members, specify a valid record length. This record length must be equal to the maximum record length of the workstation file, plus 12 bytes. This is because the transfer function automatically creates the order number and date fields when the file is transferred to the iSeries™, eServer™ i5, or System i5™ members. The order number and date fields together occupy 12 bytes.

Preparing for Data Transfer

The following topics describe the software products required to transfer data and the points you must understand before transferring data with PC400.

Required Software Products

To use Data Transfer, IBM® PC Support/400 (5738-PC1) must be installed on the iSeries™, eServer™ i5, or System i5™. IBM® PC Support/400 is not required with OS/400® Version 3 or later, or with i5/OS™.

Before using the data transfer function, run the router of PC400 or PC Support/400.

Transfer Function

You can transfer only source programs, records, and the following information:

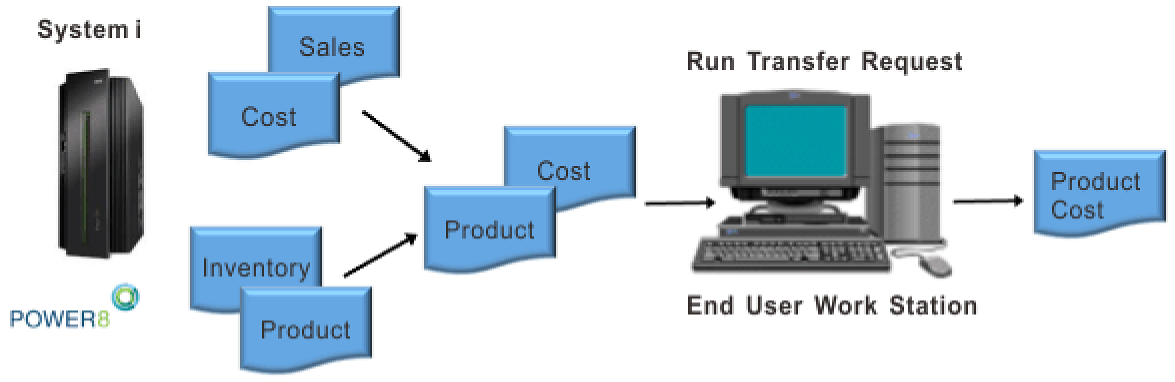
- Information organized for analysis
- Information used for decision making
- Information suited for computer processing

When using a spreadsheet, for example, you might want to use inventory data to create a cost analysis report. If there is no way to copy the data into the workstation, you must print the data from the iSeries™, eServer™ i5, or System i5™ and manually type it into a workstation file. With the transfer function, however, you can access the inventory database directly, select only the data needed for the report, process the data as required, then complete the report using that data.

You can also send data from the workstation to the host system for processing by iSeries™, eServer™ i5, or System i5™application. When a remote user is authorized to access the iSeries™, eServer™ i5, or System i5™ directly, he or she can access the created cost analysis report to compare with their results.

Figure 9: Data Transfer Example on page 102 outlines the joining of two files, transferring the information to the workstation, and creating a report.

Figure 9. Data Transfer Example



To transfer data by using PC400, you must create a transfer request. A transfer request provides the necessary information about the data you want to transfer.

Before creating a transfer request, you must have the answers to the following questions:

Where is the data located?
How much of the data do you want to transfer?
How should the data be sorted?
Where do you want data to be transferred?

When transferring data from the iSeries™, eServer™ i5, or System i5™ to a workstation, PC400 allows you to specify which data is to be transferred and whether the data is to be displayed or written to a workstation file.

In addition, a transfer request can be saved to a workstation file, allowing you to easily perform the same transfer at a later date. After a transfer request is saved, you can call the request to make changes or to run it again.

Data Transfer Program

PC400 data transfer is classified into two types, depending on the direction of the transfer:

- Transferring data from the workstation to the iSeries™, eServer™ i5, or System i5™ is called *data sending*.
- Transferring data from the iSeries™, eServer™ i5, or System i5™ to the workstation is called *data receiving*.

Data transfer can also be classified according to how the program is started, as follows:

- Data is transferred by interactively entering information such as *what data is transferred from which file to which file* on the screen. In this case, the interactive screen for sending is called the PC→iSeries™ Transfer window, and that for receiving is called the iSeries™→PC Transfer window.
- Data is transferred according to the information that has already been registered. The interactive screen is not necessary. This is called the *automatic transfer* of data.

In both cases, data transfer is performed by PCSFT5.EXE on the workstation and by the *PC Support/400* transfer program on the iSeries™, eServer™ i5, or System i5™.

The **Data Transfer** icon is registered in the PC400 folder by installing PC400. Double-clicking on this icon displays the iSeries™→PC Transfer window (for receiving). This icon includes:

```
   \Z and I Emulator for Windows\PCSFT5.EXE
```

The PC→iSeries™ Transfer window (for sending) opens when the registered contents are changed as follows:

```
   \Z and I Emulator for Windows\PCSFT5.EXE
```

The iSeries™→PC Transfer and PC→iSeries™ Transfer windows have a **Switch to SEND** button and **Switch to RECEIVE** button, respectively. By clicking either of these buttons, the window for sending can be switched to the window for receiving, and vice versa.

To perform *automatic transfer*, you must create transfer information, using the interactive screen window, and then save the information. You can then perform data transfer automatically by specifying the file name in which the data was saved.

For example, if you save transfer information to file TENSOU.TTO, contained in directory C:\Z and I Emulator for Windows\PRIVATE, run *automatic transfer* as follows:

```
“C:\Z and I Emulator for Windows\PCSFT5.EXE” “C:\Z and I Emulator for Windows\PRIVATE\TENSOU.TTO”
```

When you save the transfer information, register it as an icon in the PC400 folder. You can then transfer data automatically simply by double-clicking on this icon.

Data Concepts of the iSeries, eServer i5, or System i5 and Your Workstation

The basic components of data management are files, records, and fields. A *file* is an aggregate of records, referenced by a single name. Each record in a file contains one or more items of correlated information. Each item of information is called a *field*.

The iSeries™, eServer™ i5, or System i5™ and your workstation use different functions to store and group data, and to set the format.

Workstation Files

To transfer data from a workstation to the iSeries™, eServer™ i5, or System i5™, the transfer function uses a special-format workstation file, called a *file-description file*. Using this file, data is stored in a valid format and converted into a valid type.

A file-description file identifies the format of a workstation data file and contains a description of the fields in the data file. The file-description file also contains a name list of all the fields in the data file. This list reflects the order, as well as the names, in which each field appears within the data file. In addition, this list includes a description of the data type, length, and decimal position of each field. Using this information, the transfer function can recognize not only how data has been modified but also where a certain field exists in a file record.

When data is transferred from the iSeries™, eServer™ i5, or System i5™ to a workstation, you can use the transfer function to automatically create the file-description file. In this case, the information in the file-description file depends on the file description in the iSeries™, eServer™ i5, or System i5™ file.

You must create a file-description file with the same name as the workstation data file to transfer a workstation data file to the iSeries™, eServer™ i5, or System i5™.

Distributed Data Management (DDM) Files

Distributed data management (DDM) is one of the functions supported by iSeries™, eServer™ i5, and System i5™. This function is used to access database files that are stored on remote iSeries™, eServer™ i5, and System i5™ systems. To use the transfer function to access these database files, specify a DDM file name as the name of the iSeries™, eServer™ i5, or System i5™ file to be transferred. Refer to *DDM Guide* for details of how to use DDM files.

iSeries, eServer i5, or System i5 Files

The following list provides a simple explanation of the requirements for transferring data between the iSeries™, eServer™ i5, or System i5™ and a workstation.

Library

The iSeries™, eServer™ i5, or System i5™ library contains related objects that are used to generate significant groups. For example, the objects might be all the programs and files related to credit sales management. Using the library, you can group objects and find a desired file by name. The transfer function uses the library to locate an iSeries™, eServer™ i5, or System i5™ file.

File

iSeries™, eServer™ i5, or System i5™ files that you can manipulate consist of a file description and data stored in the file. PC400 processes an iSeries™, eServer™ i5, or System i5™ file, called a database file. The database file can be either a physical file or a logical file.

A *physical file* is a database file that contains data stored in records. It includes a description of the record format in addition to the data itself.

A *logical file* is a database file, that you can use to access data stored in one or more physical files. Logical files, like physical files, contain a file description. However, logical files do not contain any actual data. Instead, you can access fields in one or more physical files by using the record format included in the logical file description. When a logical file is transferred from the iSeries™, eServer™ i5, or System i5™ to a workstation, data is obtained from one or more physical files. You need only specify a logical file as the file to be transferred. The iSeries™, eServer™ i5, or System i5™ recognizes which physical file contains the actual data to be transferred.

Note: Data cannot be transferred from a workstation to logical files.

Member

Data records in a database file are grouped into several members. At least one member must be included in one file.

When data is transferred to and from the iSeries™, eServer™ i5, or System i5™, actual data transfer is done between file members. For example, a certain workstation file can be transferred to the iSeries™, eServer™ i5, or System i5™. In this case, the file members become new members of a new or existing iSeries™, eServer™ i5, or System i5™ file, or substitute for existing members in an existing iSeries™, eServer™ i5, or System i5™ file.

Record format

A record format describes the fields contained in a file record and the order in which these fields appear in the record. Record formats are stored in the file description. Both physical and logical database files can have one or more record formats.

Creating a Workstation-to-iSeries Transfer Request

To create a request for data transfer from a workstation to the iSeries™, eServer™ i5, or System i5™, do as follows.

1. Using the router session, establish attachment to the iSeries™, eServer™ i5, or System i5™ to which data is to be transferred.
2. Click the **Data Transfer** icon.
3. When the iSeries™→PC Transfer window displays, select **Switch to SEND**. The display is switched to the PC→iSeries™ Transfer window.

   To choose additional settings, select **Advanced**.
4. Specify each item. See Items to Be Specified on page 106 for details.

Items to Be Specified

The following section explains the items that you specify in the PC→iSeries™ Transfer window.

FROM

PC file name

This item is always required. It specifies the name of the workstation file containing the data to be transferred to the iSeries™, eServer™ i5, or System i5™. Specify this item using the following format. (Items inside brackets [ ] can be omitted.)

```
[d:][path-name]file-name[.ext]
```

A list of workstation files can be displayed by selecting **Browse**. You can limit the number of names listed. To limit the listing, specify a combination consisting of part of a file name and a global file name character (* or ?) in the input area of the workstation file list. For example:

- When you click **OK** with **/A:** specified, the displayed listing contains the names of all files in the current directory of the diskette inserted into drive A.
- When you click **OK** with **A:\SUPPLY\** specified, the displayed listing contains the names of all files under the **SUPPLY** path of the diskette inserted into drive A.
- When you click **OK** after specifying **B:*.XLS**, the displayed listing contains the names of all files having extension **XLS** in the current directory of the diskette inserted into drive B.

TO

System name

This item is always required. When the router program is active, this item specifies the default system name.

Library/File (Member)

This item is always required. It specifies the name of the iSeries™, eServer™ i5, or System i5™ physical file that will receive the data to be transferred from the workstation. You can specify either an existing file name or new file name.
Specify this item using the following format. (Items inside brackets [ ] can be omitted.)

```
[library-name/]file-name[(member-name[,record-format-name])]
```

library-name

This is the name of theiSeries™, eServer™ i5, or System i5™ library containing the iSeries™, eServer™ i5, or System i5™ file to which data is to be transferred. If no library is specified, *LIBL is used. To create a new file to receive transferred data, specify the library name.

When the input field is null and **Browse** is selected, the iSeries™, eServer™ i5, or System i5™ displays a list of all libraries defined in *USRLIBL of the iSeries™, eServer™ i5, or System i5™ job library list. You can modify this list by changing the job description. Run a change job description (CHGJOBD) command on the iSeries™, eServer™ i5, or System i5™.

file-name

This is the name of an iSeries™, eServer™ i5, or System i5™ physical database file. When data is transferred to an existing file, the data in that file is replaced with the transferred data. To create a new file to receive transferred data, specify a new file name of 1 to 10 characters.
To list the available files, do one of the following things:

- To list all files within all libraries defined in *USRLIBL of the iSeries™, eServer™ i5, or System i5™ job library list, specify ***USRLIBL** followed by a slash (/), then select **Browse**. If a slash (/) is not specified after the library name, the iSeries™, eServer™ i5, or System i5™ displays a list of library names rather than the file names.
- To list the names of the files in a certain library, specify the library name followed by a slash (/), then select **Browse**. You can also specify a part of a file name followed by an asterisk (*), then select **Browse**. The iSeries™, eServer™ i5, or System i5™ lists all the files whose names begin with the specified character string.

member-name

This is the name of a member in the specified iSeries™, eServer™ i5, or System i5™ file to which data is to be transferred. If this member name is not specified, data is transferred to the first member, *FIRST, in the iSeries™, eServer™ i5, or System i5™ file.

To transfer data to an existing file, specify the member name. The data within that file member is replaced with the transferred data.

To create a new member in an existing file or in a new file, specify a new member name of 1 to 10 characters.

By selecting **Browse** with a file name specified, the names of the members in that file are listed. When a left parenthesis, part of a member name, an asterisk (*), and a right parenthesis are specified, in this order, and then **Browse** is selected, the iSeries™, eServer™ i5, or System i5™ can list all member names beginning with the specified character string.

record-format-name

This is the name of the record format in the specified iSeries™, eServer™ i5, or System i5™ file. The record format name need not be specified except when a physical file contains more than one record format. Most physical files have only one record format. Before specifying a record format name, a member name or *FIRST must be specified as the member name.

When you transfer data to an existing file without specifying a record format name, it is assumed that the file has only one record format (*ONLY). Therefore, that record format is used.

When a new file is created with no record format name, QDFTFMT is used as the record format name.

Note: A library name, file name, member name, and record format name can be specified using up to 10 characters each. Each name must begin with one of the following characters: A to Z, ¥, #, or @. For characters subsequent to the first, the numbers 0 to 9, underscores, and periods can also be used.

Advanced Options

The following advanced options are available for PC→iSeries™ Transfer.

Use of File Description File

This item specifies whether a file-description file is used to transfer data to the iSeries™, eServer™ i5, or System i5™. The file-description file is required to transfer a workstation file, containing the data to be transferred (and converted), in fields. Such a workstation file can have either several fields or numeric data fields. To transfer a workstation file containing text (character data) only, the file-description file is not required. For details on creating a file-description file, see File-Description Files on page 147.

- Do not specify this item in the following case: a workstation file having only one field (for example, PC code character) is specified in **FROM**, while the iSeries™, eServer™ i5, or System i5™ file is a physical source file having the following record format.

```
  Field        Type          Length      "  " Decimal Places

Order number   Zoned            6                 2
Date           Zoned            6                 0
Data           Character     1 to 4096
               or Open
```

  Note: When fields contain character data or zoned data only, the data portion can be broken down into several fields. The destination iSeries™, eServer™ i5, or System i5™ file contains the fields for order number and date. The workstation file, however, does not. This method is recommended when transferring text only between the iSeries™, eServer™ i5, or System i5™ and the workstation.

- Specify this item in all other cases. Two examples are:
  - Data is transferred from a workstation file having more than one field.
  - The iSeries™, eServer™ i5, or System i5™ file that receives the data is other than a physical source file having the record format described above.

File Description File Name

This item appears only when item **Use of File Description File** is specified.

This item is always required. It specifies the name of the workstation file-description file that describes the data to be transferred.

Upon transferring data from the iSeries™, eServer™ i5, or System i5™ to a workstation, a file-description file might have been created.

A file-description file must be created when the data has not yet been transferred from the iSeries™, eServer™ i5, or System i5™ to a workstation or when no file-description file exists.

PC File Type

This item appears only when **Use of File Description File** is not specified.

This item is always required. You must specify the type of the workstation file specified in the **FROM** field. The values provided by the iSeries™, eServer™ i5, or System i5™ are recognized as workstation code text. If the file type of a data file is *not converted*, the file can include nothing other than data that does not require conversion.

iSeries Object

This item is always required. It specifies whether the iSeries™, eServer™ i5, or System i5™ member to which data is transferred is a new member or an existing member. When data is transferred to a new member, this item also specifies whether the file to contain the new member is an existing file.

Create New Member

This item specifies that a new member, to which data is transferred, is created in an existing iSeries™, eServer™ i5, or System i5™ file.

Note:

1. To create a new member, you must have the following authorities:
   - *OBJOPR, *OBJMGT, and *ADD for the file that will include the new member
   - *READ and *ADD for a library that will contain the file

   See *Security Descriptions* (SC41-8083) for details of object authorities.
2. To create a member to add to a file, the transfer function uses the iSeries™, eServer™ i5, or System i5™ default value for the add physical file member (ADDPFM) command.

When you specify this item, the following item must also be specified:

Member Text

This item is used to add an explanation of a new iSeries™, eServer™ i5, or System i5™ member. This explanation helps remind you of the contents of the member. This explanation appears, for example, when a list of all members in a file is requested (**Browse** is selected). If this item is left blank, no explanation is added to the new iSeries™, eServer™ i5, or System i5™ member.

To specify an apostrophe (') in the explanation, enter two apostrophes (' ').

Create New Member in New File

This item specifies that a new member, to which data is to be transferred, is created in a new iSeries™, eServer™ i5, or System i5™ file.

Note:

1. To create a new member in a new file, *READ and *ADD authorities are required for the library that will contain that file. Authority to use the create physical file (CRTPF) command of the iSeries™, eServer™ i5, or System i5™ is also required.
2. To create a new member in a new file, the transfer function uses the default value for the create physical file (CRTPF) command of the iSeries™, eServer™ i5, or System i5™. It does not, however, use the following values:

   (MAXMBRS[*NOMAX]). This indicates that the file can contain up to 32,767 members.

   (SIZE[*NOMAX]). This indicates that each member of the file can contain an unlimited number of records.

When this item is specified, also specify the following item:

Member Text

This item is optional. It is used to add an explanation of a new iSeries™, eServer™ i5, or System i5™ member. This explanation helps remind you of the contents of the member. This explanation appears, for example, when a list of all the members in a file is requested (**Browse** is selected). If this item is left blank, no explanation is added to the new iSeries™, eServer™ i5, or System i5™ member.

To specify an apostrophe (') in the explanation, enter two apostrophes (' ').

iSeries File Type

This item is always required. It specifies the type of iSeries™, eServer™ i5, or System i5™ file and the members to be created (same type for both).
Specify one of the following things:

- To create an iSeries™, eServer™ i5, or System i5™ physical source file and its members, specify **Source**. These members are created with two fields (order number and date) added to the beginning of the data transferred from the workstation file. A new iSeries™, eServer™ i5, or System i5™ source file and its members have the following record format:

```
Field         Type       Length       Decimal Places

Order number  Zoned         6            2
Date          Zoned         6            0
Data          Character   1 to 32755
              or Open
```

Note that in an iSeries™, eServer™ i5, or System i5™ physical source file, each record can be up to 32 755 bytes in length. But, the maximum size of a source file created using the workstation-to-iSeries transfer function is 4,107 bytes. Also, this file must include the order and date fields. Therefore, the maximum amount of data that can be transferred is 4,096 bytes per record.

The data portions of members inherit the workstation file characteristics. In other words, when a workstation file is a workstation code text file consisting of many records containing text, the created data fields will be the same.

- To create aniSeries™, eServer™ i5, or System i5™ physical data file and its members, specify **Data**. The file and members will contain only the data fields described in the file-description file.

The value of the **iSeries™ File Type** is assumed to be **Data** when a file-description file is used to transfer data. If a file-description file is not used for data transfer, the value of this item is assumed to be **Source**.

Field Reference File Name

This item appears only when **Use of File Description File** is specified for the creation of a new file.

When **Use of File Description File** is not specified, an iSeries™, eServer™ i5, or System i5™ physical source file is created. **iSeries™ File Type** and **Field Reference File Name** are not displayed. Instead, **Record Length** opens.

This item is always required. A new iSeries™, eServer™ i5, or System i5™ file is created using the field name in a file-description file and the field definitions in an iSeries™, eServer™ i5, or System i5™ field-reference file.

The format of a field-reference file name is as follows. (Items inside brackets [ ] can be omitted.)

```
[library-name / ] file-name
```

library-name

This is the name of an iSeries™, eServer™ i5, or System i5™ library containing a field-reference file. If this library name is not specified, *LIBL is assumed. If you cannot find the desired library, selecting **Browse** displays a list of all libraries in *USRLIBL of the iSeries™, eServer™ i5, or System i5™ job library list. *USRLIBL of the library list can be changed by modifying the job description by executing a CHGJOBD command on the eServer™ i5 or iSeries™ processor.

file-name

This is the name of the iSeries™, eServer™ i5, or System i5™ physical database file containing the field definitions. Always specify this file name. When a library name is specified concurrently, use a slash (/) to delimit the library name and file name. If the desired file cannot be found, enter the library name and a slash, then select **Browse**. The system displays a list of files in that library. To list all the files in the libraries defined in *USRLIBL of theiSeries™, eServer™ i5, or System i5™ job library list, enter *USRLIBL/ then select **Browse**.

If you enter part of a file name followed by an asterisk (*) and then select **Browse**, the system displays a list of available file names, each beginning with the specified part of the name.

For example, enter ARLIB/AR* in the **Field Reference File Name** item, then select **Browse**. The system displays a list of all physical file names beginning with AR in library ARLIB.

Note: You must have *OBJOPR authority for the field-reference file to be specified. To list certain files, you must also have *OBJOPR authority for those files.

Record Length

This item is always required. It specifies the record length of an iSeries™, eServer™ i5, or System i5™ physical source file. When the data receiver is an iSeries™, eServer™ i5, or System i5™ physical source file, the specified value must include the length of the order number and date fields that are added to a workstation file at transfer (the total length of these two fields is 12 bytes).

Authority

This item is always required. It specifies the authority level of a new iSeries™, eServer™ i5, or System i5™ file.
Specify one of the following things:

- **Read/Write**. This enables other users to read from and write to the iSeries™, eServer™ i5, or System i5™ file and allows the file name to be displayed in lists. However, users cannot delete the file (*OBJOPR, *READ, *ADD, *OBJMGT, *UPD, and *DLT authorities). If other users might be transferring data from a workstation file to the iSeries™, eServer™ i5, or System i5™ file, specify **Read/Write** or **All**.
- **Read**. This enables other users to read from the iSeries™, eServer™ i5, or System i5™ file, and allows the file name to be displayed in lists. However, other users can neither write to the file nor delete it (*USE authority).

- **All**. This enables other users to read from and write to the iSeries™, eServer™ i5, or System i5™ file as well as delete it. The file name is displayed in lists (*ALL authority).
- **None**. This prevents other users (except for the system administrator) from writing to or deleting the iSeries™, eServer™ i5, or System i5™ file. The file name does not appear in lists (*EXCLUDE authority).

File Text

This item is optional. It is used to add an explanation of a new iSeries™, eServer™ i5, or System i5™ file. This explanation helps remind the user of the contents of the file. This explanation appears, for example, when a list of all files in a library is requested (**Browse** is selected). If this item is left blank, no explanation is added to the new iSeries™, eServer™ i5, or System i5™ file.

To specify an apostrophe (') in the explanation, enter two apostrophes (' ').

Replace Existing Member

This item transfers data to an existing iSeries™, eServer™ i5, or System i5™ member, specified in the **Library/File (Member)** item. The existing data in that iSeries™, eServer™ i5, or System i5™ member is replaced with the transferred data.

Saving, Opening, Changing, and Executing a Transfer Request

The following section explains how to save, open, change, and execute, as a file, information (transfer request) on data to be transferred.

Saving a Transfer Request

Save a transfer request when the request is likely to be executed repeatedly. This eliminates the need to create a transfer request every time data is to be transferred. To save a transfer request, do as follows:

1. Specify the information needed for transfer, using the PC→iSeries™ Transfer window.
2. After specifying the necessary information, click **Save** or **Save As** from the **File** menu.

   The Save Transfer Request File As window opens.
3. Specify each item, referring to the following explanation, then click **OK**.

   File Name

   Disk to which data is to be saved. Specify a file name or diskette file name. The default extension is TFR. Extension TFR identifies a file as a transfer request file.

   Description

   This item can be used to add an additional explanation of a transfer request, as required. The explanation can be up to 40 characters in length. This explanation is saved with the transfer request, and displayed in the list of transfer request names. It is, therefore, useful for identifying a transfer request.

4. The system asks whether the saved transfer request is to be registered in the PC400 folder.

   When you click **OK**, the transfer request is registered as an icon. Subsequently selecting this icon transfers data according to the contents of the registered data transfer request.

Opening and Changing a Saved Transfer Request

To open and change a saved transfer request, do as follows:

1. Display the PC→iSeries™ Transfer window.
2. Click **Open** from the **File** menu.
3. Specify the name of the file to be opened using the Open Transfer Request File window, then click **OK**.

   The PC→iSeries™ Transfer window reopens, and the transfer request information, saved to the specified file, appears for each item. This opens the saved transfer request.
4. Change the contents of the transfer request as necessary.
5. To save the changed contents, follow the procedure explained in Saving a Transfer Request on page 113.

Performing a Transfer Request

A transfer request can be performed in any of the following ways:

- By clicking the icon with which the transfer request has been registered
- By using the PC→iSeries™ Transfer window of the Data Transfer icon

Clicking the Icon with Which the Transfer Request Has Been Registered

This method can be used only when a transfer request has been saved as an icon by using the PC→iSeries™ Transfer window.

Clicking the corresponding icon starts data transfer.

Using the PC→iSeries Transfer Window

1. Before executing a transfer request, operations such as creating, opening, and changing a transfer request must be completed.

   Note: When data is transferred from a workstation to an existing member in an iSeries™, eServer™ i5, or System i5™ file, the transferred data replaces the existing data in that member.

2. Select **Send** from the PC→iSeries™ Transfer window.

   Data transfer starts.
3. After the transfer has been completed, click **Cancel** or **Exit** from the **File** menu.

Conversion Errors That Can Occur during Transfer

Upon executing a transfer request, a file-description file (when specified) is read from the disk or diskette to be processed. The iSeries™, eServer™ i5, or System i5™ and workstation exchange information, if the data is transferable.

The workstation transfers records, one at a time, from the file specified in **FROM**. Transferred records are converted and stored in the iSeries™, eServer™ i5, or System i5™ member specified in **TO**.

During this conversion process, conversion errors might occur. For example, the values in a workstation file might have to be rounded to fit the iSeries™, eServer™ i5, or System i5™ fields. Another example is the case where the record length of a workstation file differs from that expected by the iSeries™, eServer™ i5, or System i5™.

If such an error occurs, an error message is issued with the number of the workstation file record for which the error occurred and, sometimes, information about certain fields in that record.

If a severe error occurs, data transfer might stop. In such a case, stop the transfer request, correct the error, then rerun the transfer request.

When the error is not so severe, you can request that the system continue transferring data. By doing so, even if the same error occurs in another record, an error message does not appear and the transfer function automatically continues executing the transfer request.

Creating an iSeries-to-Workstation Transfer Request

To create a transfer request to receive data from the host, do as follows:

1. Click the **Data Transfer** icon.
2. When the PC→iSeries™ Transfer window opens, select **Switch to RECEIVE** to switch the display to the iSeries™→PC Transfer window.

   For the additional settings, click the **Advanced** button.
3. Which items are to be specified by the user vary with the data type, as follows:
   - Entire iSeries™, eServer™ i5, or System i5™ file
   - Part of an iSeries™, eServer™ i5, or System i5™ file
   - Data combined from several iSeries™, eServer™ i5, or System i5™ files
   - Summary of record groups

   Before specifying each item, while referring to Items to Be Specified on page 118, note the following points regarding the data to be received.

Receiving an Entire iSeries, eServer i5, or System i5 File

This is the simplest way of transferring data from the iSeries™, eServer™ i5, or System i5™ to a workstation. All records in a file and all the data in each record are transferred.

The **FROM** items are as follows:

System name

This item specifies the name of the system.

Library/File (Member)

This item specifies the name of the iSeries™, eServer™ i5, or System i5™ file.

SELECT

Specifying an asterisk (*) for this item indicates that all fields are to be transferred, or lists all the fields in the iSeries™, eServer™ i5, or System i5™ file.

ORDER BY

This item is optional. It specifies how records are grouped. When this item is left blank, records are not grouped (data is transferred in the same order it appears in the iSeries™, eServer™ i5, or System i5™ file).

Receiving Part of an iSeries, eServer i5, or System i5 File

Only part of an iSeries™, eServer™ i5, or System i5™ file is transferred to the workstation.

The **FROM** items are as follows:

System name

This item specifies the name of the system.

Library/File (Member)

This item specifies the name of the iSeries™, eServer™ i5, or System i5™ file.

SELECT

This item specifies a field to be transferred.

WHERE

This item specifies the requirements that must be satisfied before records can be selected for transfer.

ORDER BY

This item is optional. It specifies how records are grouped. When this item is left blank, records are not grouped (data is transferred in the same order it appears in the iSeries™, eServer™ i5, or System i5™ file).

Receiving Data Combined from Several iSeries, eServer i5, or System i5 Files

The data to be transferred can be stored in two or more iSeries™, eServer™ i5, or System i5™ files. These files are assumed to be related. Based on this relationship, they can be linked or *joined*, as if all the data existed in a single file. The files can be transferred to the workstation after they have been joined. By using the iSeries™→PC Transfer function, this “join and transfer” function can be performed in a single step.

The **FROM** items are as follows:

System name

This item specifies the name of the system.

Library/File (Member)

This item specifies the names of all iSeries™, eServer™ i5, or System i5™ files from which data is to be transferred.

JOIN BY

This item specifies how to join or combine the data in each file.

SELECT

This item specifies a field to be transferred.

WHERE

This item specifies the requirements that must be satisfied before records can be selected for transfer.

ORDER BY

This item is optional. It specifies how records are grouped. When this item is left blank, records are not grouped (data is transferred in the same order as it appears in the iSeries™, eServer™ i5, or System i5™ file).

Receiving a Summary of Record Groups

A summary record is a single record that includes information on each set of records grouped from one or more iSeries™, eServer™ i5, or System i5™ files.

The **FROM** items are as follows:

System name

This item specifies the name of the system.

Library/File (Member)

This item specifies the names of all files from which data is to be transferred.

JOIN BY

This item is optional. It specifies the join conditions that must be satisfied before records can be joined.

GROUP BY

This item is optional. It must be specified only when the records of iSeries™, eServer™ i5, or System i5™ files are classified into several groups. To group all records into a single group, this item need not be specified.

SELECT

Specifying this item creates a summary record. The field names specified in **GROUP BY** can be specified.

WHERE

This item is optional. It specifies the requirements that each record to be grouped must satisfy. To group all records, this item need not be specified.

HAVING

This item is optional. It specifies the summary record to be transferred. To transfer all summary records, this item need not be specified.

ORDER BY

This item is optional. It specifies how summary records will be grouped. When this item is left blank, records are not grouped (data is transferred in the same order as it appears in the iSeries™, eServer™ i5, or System i5™ file).

Items to Be Specified

The following section explains the items to be specified using the iSeries™→PC Transfer window.

FROM

System name

This item specifies the name of the host system that contains the data to be received. When the router program is active, this item specifies the default system name.

Library/File (Member)

This item is always required. It specifies the name or names of one or more files used to store data to be transferred. Up to 32 file names can be specified. To specify several files, delimit them with commas and use JOIN BY, displayed after all **FROM** items have been specified. Only the file name must be specified. Do not specify a comma as a part of a file name. When the other optional items are not specified, they are assumed automatically. For example, the library name, member name, and format name can be assumed to be *LIBL, *FIRST, and *ONLY, respectively. When the cursor is on the input field of **FROM**, selecting **Browse** lists libraries, files, members, and formats.

Note: To transfer data from an iSeries™, eServer™ i5, or System i5™ physical file, you must have *USE authority for that file. To transfer data from an iSeries™, eServer™ i5, or System i5™ logical file, you must have *OBJOPR authority for that file and *READ authority for each subordinate file.

Specify file names as follows. (Items inside brackets [ ] can be omitted.) To specify several file names, delimit the names with commas.

```
[library-name/]file-name[(member-name[,record-format-name])],
[library-name/]file-name[(member-name[,record-format-name])],...
```

library-name

This is the name of the iSeries™, eServer™ i5, or System i5™ library that contains the iSeries™, eServer™ i5, or System i5™ file to be transferred. This iSeries™, eServer™ i5, or System i5™ file contains the data to be transferred from the iSeries™, eServer™ i5, or System i5™ to a workstation. If this library name is not specified, *LIBL is assumed. If you cannot find the desired library, selecting **Browse** displays a list of all libraries defined in *USRLIBL of the iSeries™, eServer™ i5, or System i5™ job library list. *USRLIBL of the library list can be changed by modifying the job description by executing the CHGJOBD command on the iSeries™, eServer™ i5, or System i5™.

file-name

This is the name of the iSeries™, eServer™ i5, or System i5™ physical file, logical file, or DDM file from which data is transferred. This file name must always be specified. To specify a file name and library name concurrently, delimit them with a slash (/). If you cannot find the desired file name, enter the library name followed by a slash, then select **Browse**. The system then displays a list of files contained in that library. To display a list of all the files in the libraries defined in *USRLIBL of the iSeries™, eServer™ i5, or System i5™ job library list, enter *USRLIBL/, then select **Browse**.

member-name

This is the name of the iSeries™, eServer™ i5, or System i5™ member containing the data to be transferred, or *FIRST. If this member is not specified, the system assumes *FIRST, and the first member of that file is used.

record-format-name

This is the name of the record format contained in the specified iSeries™, eServer™ i5, or System i5™ file, or *ONLY. Before specifying the record format name, specify the member name or *FIRST. If the record format name is not specified, the system assumes *ONLY, and the only record format for that file is used. To specify a record format name, delimit the record format name and member name with a comma.

When the specified iSeries™, eServer™ i5, or System i5™ file has several record formats, a record format name must be specified. If the file member name is not specified, a record format name cannot be specified.

Note:

1. A library name, file name, file member name, and record format name can be specified using up to 10 characters for each. Each name must begin with A to Z, ¥, #, or @. For characters subsequent to the first, 0 to 9, underscores, and periods can also be used.
2. When the **FROM** field remains blank or a comma is entered to specify the next file name, selecting **Browse** displays a list of libraries defined in *USRLIBL of the iSeries™, eServer™ i5, or System i5™ job library list.
3. Enter part of the file name, member name, or record format name, followed by an asterisk (*), then select **Browse**. The system displays a list of names beginning with the specified characters.

For example, you might want to transfer data from file member ITEMMBR1 (first member) of file ITEMMAST in library ITEMLIB. ITEMFMT is the only record format of this file. The specification will be as follows:

```
ITEMLIB/ITEMMAST(ITEMMBR1,ITEMFMT)
```

Alternatively, specify:

```
ITEMLIB/ITEMMAST
```

Receiving a Summary of Record Groups

The following information is necessary to receive summary records.

To transfer a summary record, do not leave this input area blank or specify an asterisk (*) (except when all the fields of the file specified at the prompt are specified in **GROUP BY**). The field names specified in **SELECT** (except for those specified in functions) must also have been specified in **GROUP BY**.

The functions and fields specified in **SELECT** return actual summary information for each group. Enter the field names and functions in **SELECT** in the order in which they are to be displayed.

Note: Null values are not included in the functions. When an entire value is null, the function output is set to null, except for **COUNT**. The **COUNT** output is 0.

The function format is as follows.

```
function (field-name)
```

This has the following meaning:

function

This is one of the following functions:

AVG

Transfers the average value of the specified fields for each record group. This function can be used only for numeric fields.

MIN

Transfers the minimum or lowest value of the specified fields for each record group.

MAX

Transfers the maximum or highest value of the specified fields for each record group.

SUM

Transfers the total value of the specified fields for each record group. This function can be used only for numeric fields.

COUNT

Transfers the total number of records that satisfy the **WHERE** condition for each record group. Specify **COUNT(*)**.

field-name

This is the field name defined with the record format specified in **FROM**.

Each function returns one value for each record group. In **SELECT**, several functions can be specified. To do so, delimit the functions by commas, as follows:

```
SUPPNO, AVG(PRICE), MIN(PRICE), MAX(PRICE)
```

This indicates that the average, minimum, and maximum values for PRICE are calculated for each supplier after SUPPNO has been selected. A summary record is transferred according to the function selection. Specify SUPPNO in **GROUP BY**, because SUPPNO has not been used for the functions.

Advanced Options

The following advanced options are available for iSeries™→PC Transfer.

JOIN BY

When several files have been specified in **FROM**, specify **JOIN BY**. When only one file has been specified in **FROM**, **JOIN BY** does not appear.

JOIN BY specifies how to link or join the records of the files specified in **FROM**. Each file specified in **FROM** must be joined with at least one other file that has been specified in **FROM**.

Use **JOIN BY** to specify one or more join conditions. The join conditions indicate the similarity of two files. Therefore, they indicate which records of one file are joined with those of another.

The join conditions are as follows:

```
field-name  =  field-name
```

Field name is the name of the field defined in the record format specified in **FROM**. The join conditions require two field names, one for each file to be joined.

Field names must be delimited by one of these:

=

Equal

<> or ><

Not equal

>

Greater than

>=

Greater than or equal to

<

Less than

<=

Less than or equal to

When specifying fields in **JOIN BY**, observe the following rules.

- Join a numeric field to another numeric field. The field lengths and types do not have to be identical.
- Join a character field to another character field. The lengths do not have to be identical.

The field name to be specified might have been defined in the files specified in **FROM**. When such a field name is used in the following items, prefix the field name with the file qualifier:

- JOIN BY
- GROUP BY
- SELECT
- WHERE
- HAVING
- ORDER BY

The file qualifier is the character T (uppercase or lowercase) followed by a one- or two-digit number. Use T1 for fields defined with the first record format, T2 for fields defined with the second record format, and so on. Delimit the file qualifier and field name with a period (`.`). See Receiving Records Using File Qualifiers on page 138 for details of the file qualifiers.

If the field name of the file specified in **FROM** cannot be found, select **Browse** when the cursor is on the **JOIN BY** input area. Then, a list of file qualifiers and field names of the files appears.

To join three or more files, or to join two files based on two or more common fields, two or more link conditions must be used. To specify several join conditions, join the conditions with AND. For example:

```
T1.EMPNO = T2.EMPNO AND T2.EMPNO = T3.EMPNO
```

In this case, records having the same value as EMPNO are joined between the first and second files specified in **FROM**. Then, such records are joined between the second and third files specified in **FROM**.

Up to 32 join conditions can be specified.

After **JOIN BY** is specified, each of **SELECT**, **WHERE**, and **ORDER BY** can be completed, by following the procedure described earlier in this chapter. To browse a field name that has been defined in several files, prefix the field name with a file qualifier.

GROUP BY

This item is required only to classify iSeries™, eServer™ i5, or System i5™ file records into several groups. When no value is specified in **GROUP BY**, all the records are treated as a single group.

If **GROUP BY** is not displayed, select **Group functions** at the bottom right of the screen. Then, **GROUP BY** appears. **GROUP BY** and **HAVING** are displayed concurrently. You can specify either, both, or neither.

When **GROUP BY** and **HAVING** are displayed but you do not want to specify either, select **Remove Group functions**. The two items disappear.

To classify several records into groups, specify one or more fields to act as the base for grouping. Records are grouped according to the field specified first, then by the field specified second, and so on. For example, suppose that the following groupings are specified:

```
SHIFT, DEPTNO
```

In this example, the records are first grouped by SHIFT. Records belonging to a single group will subsequently have the same value as SHIFT. Then, the records in each group are grouped by DEPTNO. When there is only one record having a certain SHIFT value, the group has only one record.

Delimit field names with commas. Blanks can be specified to improve readability. Up to 50 field names can be specified. These fields must have been defined in the record format defined in **FROM**.

If a field cannot be found, selecting **Browse** displays a list of all the fields contained in the record.

With **GROUP BY** specified, specify **SELECT** to transfer the summary record of each group.

SELECT

This item is always required. It specifies the field to be transferred or the function that indicates the type of summary information to be transferred.

The field to be specified must have been defined in the record format specified in **FROM**.

To transfer all the fields in the specified record, specify an asterisk (*) in this input field. (Specifying an asterisk causes all fields in the record to be transferred.)

Note: Up to 256 fields can be transferred. When more than 256 fields have been defined in a file, an asterisk cannot be used. In this case, specify the names by selecting the fields to be transferred.

To transfer fields by selecting from a record, enter the field names in the order in which the fields are arranged. One or more blanks can be placed between the field names to improve readability. However, the names must be delimited by commas, as follows:

```
ITEMNO,  QONHAND,  PRIC
```

You can also specify:

```
ITEMNO,QONHAND,PRICE
```

When records are transferred from an iSeries™, eServer™ i5, or System i5™ source file, specifying an asterisk (*) causes all fields in the file to be transferred, with the exception of the order number field and date field. (To transfer all the fields, including the order number field and date field, specify all the field names, including each data field name.)

A field can be specified repeatedly as required. However, bear in mind that no more than 256 fields can be selected. A list of field names can be displayed by selecting **Browse**.

WHERE

This item is optional. It specifies one or more conditions that records to be transferred must satisfy.

To transfer summary records, use this item to specify which records are to be grouped, then group the records. Using this item, you can specify one or more conditions that the record must satisfy to belong to a certain group. When **WHERE** is not specified, all records are grouped.

As the *conditions*, specify the test to be applied to the records in the specified file member. All the records in the specified file member are tested for the conditions specified here. Only those records that pass this test are transferred.

When **WHERE** is not specified, all records in the specified file member are transferred.

The condition format is as follows:

```
field-name  test  value
```

field-name

This must be a field substring or field name defined in the record format.

Fields or constants can be manipulated by specifying a supported function, with the results being used for comparison. The supported functions and usage are as follows:

SUBSTR

Returns the specified part of a character string. This function contains three parameters: the field name, starting position, and length of the returned substring. The following example returns the 20 characters starting from the 10th character of the FULLNAME field:

SUBSTR(FULLNAME 10 20)

VALUE

Returns the first non-null value in the parameter list. (If all parameters are null, null is returned.)

VALUE(DEPOSIT WITHDRAW BALANCE)

CURRENT

Returns DATE, TIME, TIMEZONE, or TIMESTAMP for the current system.

CURRENT(TIMEZONE)

DIGITS

Returns a character string representation of a numeric field.

DIGITS(EMPLOYEE#)

CHAR

Returns a character string representation of the date field, time field, or time-stamp field. The second parameter is used to specify the format of the Systems Application Architecture® (SAA®) of the string to be returned (supported values are USA, EUR, ISO, or JIS).

CHAR(DATEHIRE USA)

DATE

Returns the date of the time-stamp field.

DATE(TIMECRTD)

TIME

Returns the time of the time-stamp field.

TIME(TIMECRTD)

TIMESTAMP

Returns the time-stamp, combining the date field and time field.

TIMESTAMP(DATESEND TIMESEND)

YEAR

Returns the year of the date field or time-stamp field.

YEAR(DATEHIRE)

MONTH

Returns the month of the date field or time-stamp field.

MONTH(DATEHIRE)

DAY

Returns the date of the date field or time-stamp field.

DAY(DATEHIRE)

DAYS

Returns the day of the year, counted from January 1, of the date field or time-stamp field.

DAYS(DATEHIRE)

HOUR

Returns the time of the time field or time-stamp field.

HOUR(TIMESEND)

MINUTE

Returns the minute of the time field or time-stamp field.

MINUTE(TIMESEND)

SECOND

Returns the second of the time field or time-stamp field.

SECOND(TIMESEND)

MICROSECOND

Returns the microsecond of the time field or time-stamp field.

MICROSECOND(TIMECRTD)

test

This is the comparison type to be applied to fields or functions.

The following tests can be used. One or more blanks can be placed before and after these tests.

Note: Values are searched according to the exact characters specified by the user. In other words, when the user's specification consists only of uppercase characters, only uppercase character strings are returned. Similarly, when the specification consists only lowercase characters, only lowercase character strings are returned.

=

Equal

<> or ><

Not equal

>

Greater than

>=

Greater than or equal to

<

Less than

<=

Less than or equal to

LIKE

The field is similar to the specified value.

BETWEEN

The field is equal to one of two constants, or to a value between them.

IN

The field is the same as one of the values in the constant list.

IS

The field contains null values.

ISNOT

The field contains no null values.

Test usage is as follows:

Using the LIKE Test

The **LIKE** test checks the field specified with the field name for a character pattern specified as a value. The field to be specified must be a character field.

The values to be tested must be character-string constants. This string can contain any characters. A percent (%) character indicates a character string consisting of zero or more characters. A 1-byte underscore (_) character indicates any single 1-byte character. A 2-byte underscore (_) character indicates any single 2-byte character.

The following example explains how to use the **LIKE** test:

```
NAME LIKE '%ANNE%'
```

The previous example searches for names containing character string ANNE, such as ANNE, ANNETTE, and SUZANNE.

The following example searches for names beginning with character string ANNE, such as ANNE and ANNETTE.

```
NAME LIKE 'ANNE%'
```

The following example searches for names ending with character string ANNE, such as ANNE and SUZANNE.

```
NAME LIKE '%ANNE'
```

The following example searches for all names whose second character is A.

```
NAME LIKE '_A%'
```

The following example searches for all last names beginning with character J.

```
LSTNAM LIKE 'J%'
```

This has the same effect as the following example:

```
SUBSTR (LSTNAM,1,1) = 'J'
```

When the pattern does not include a percent character (%), the length of the character string must be identical to that of the field.

Using the BETWEEN Test

The **BETWEEN** test checks the fields specified in the field name for character strings or numeric values that are equal to or between the specified constants. The values to be tested must be two character-string constants or two numeric constants. The types of these constants must be identical to that of the field name specified by the user. Delimit the two constants with AND.
The following example searches for those records for which the price is between 50.35 and 75.3, inclusive:

```
PRICE BETWEEN 50.35 AND 75.3
```

The following example searches for those records for which the name begins with C:

```
NAME BETWEEN 'C' AND 'CZZZZZZZZZ'
```

The following example searches for those records for which the balance is between 0 and 5␠000.

```
BALDUE BETWEEN 0 AND 5000
```

This has the same meaning as the following expression.

```
BALDUE >= 0 AND BALDUE <= 5000
```

Note: Specify the values to be tested in the form of **BETWEEN** (minimum) AND (maximum). For instance, `BETWEEN 1 AND 10` is a valid specification. However, `BETWEEN 10 AND 1` returns no records.

Using the IN Test

The **IN** test checks the fields specified in the field name for the character strings or numeric values in the list specified as the value. The value to be tested must be a list of character-string constants or numeric constants. In addition, the types of these constants must be identical to that of the specified field. Delimit the constants with blanks and enclose them in parentheses. Up to 100 constants can be specified. The following example shows how to use the **IN** test:

```
NAME IN ('SMITH' 'JONES' 'ANDERSON')
```

This example searches for those records for which the name is SMITH, JONES, or ANDERSON.

The following example searches for the values in the **STATE** field for which the value is other than NY, MN, or TX:

```
NOT STATE IN ('NY' 'MN' 'TX')
```

Note: Values are searched according to the exact characters specified by the user. In other words, when the user's specification consists of only uppercase characters, only uppercase character strings are returned. Similarly, when the specification consists of only lowercase characters, only lowercase character strings are returned.

Using the IS Test

The **IS** test checks the fields specified in the field name for null values.

The following example searches for those records for which the commission field contains null values:

```
COMMISSIONS IS NULL
```

Using the ISNOT Test

The **ISNOT** test checks the fields specified in the field name for non-null values.

The following example searches for those records for which the commission field does not contain null values:

```
COMMISSIONS ISNOT NULL
```

In the test, logical AND and logical OR can be combined. When both AND and OR are specified, AND comparison is performed first. Up to 50 conditions can be specified. For example:

```
MONTH=2 AND LOC='MIAMI' OR LOC='CHICAGO'
```

In this example, each record to be selected must satisfy the following condition:

```
MONTH=2 AND LOC='MIAMI'
```

or must satisfy the following condition:

```
LOC='CHICAGO'
```

This command can be modified by using parentheses. For example:

```
MONTH=2 AND (LOC='MIAMI' OR LOC='CHICAGO')
```

In this example, each record to be selected must satisfy the following condition:

```
MONTH=2
```

and it must satisfy the following condition:

```
LOC='MIAMI' OR LOC='CHICAGO'
```

NOT can also be used. The following example selects items where data is transferred not only from those records in which the DEPT field is not equal to 470, but also from those records for which the DEPT field is equal to 470 and, additionally, STATE is equal to NY.

```
NOT (DEPT = 470) OR (DEPT = 470 AND
  STATE = 'NY')
```

Comparison can start from a certain line and end at the next line. However, a field name cannot start from a certain line and end at the next line. Field names must not exceed one line.

When a value to be tested is a character string enclosed in quotation marks, the value can start from a certain line and continue to the next line.

HAVING

This item is optional. It specifies which summary record is transferred.

Pay particular attention to the difference between **HAVING** and **WHERE**. **WHERE** operates on each record within a certain group. **HAVING**, on the other hand, operates only on summary records (records that contain summary information for each group).

With this item, you can specify one or more conditions that a summary record must satisfy prior to being transferred.

As the conditions, specify the tests that should be applied to the summary records. The specified test conditions are applied to all summary records, only those summary records that pass the tests are transferred. To transfer all summary records, leave the **HAVING** item blank.

The format of the conditions is as follows:

```
function (field-name)  test  value
```

This indicates:

function

This is a function supported for **SELECT**. See the description of **SELECT** in this section for details of these functions.

field-name

This is the field defined by the record format specified in **FROM**. A field name is acceptable even when it has not been specified in **SELECT**.

Test

This is the comparison type for functions. The types are listed below.

=

Equal to

<> or ><

Not equal

>

Greater than

>=

Greater than or equal to

<

Less than

<=

Less than or equal to

value

This is a function operating on certain fields or a constant. See WHERE on page 123 for details of constants, expressions, and tests.

Note: A comma is treated as a decimal point. Therefore, do not separate numbers with commas.

Test conditions can be combined by using logical AND or logical OR. When both AND and OR are specified, AND comparison is performed first. Up to 50 tests can be specified. By using parentheses, the operation order can be modified, or a description can be added to an operation. For example, you can specify:

```
COUNT(*) >=2 AND MAX(PRICE) > 100
```

In this case, the following conditions are applied concurrently: groups to be transferred must contain more than one record, and the summary records in such groups are transferred only when the maximum price is greater than 100.

If the desired field cannot be found, selecting Browse displays a list of the names of all fields in the record.

The type, length, digit, and number of decimal places of the value returned for each function are:

```
            Type     Length    Digit       Decimal Places
SUM         Packed     16       31    (Same as tested field)
AVG         Packed     16       31     31 (Total of the digit
                                           and decimal places
                                           of the field)
COUNT       Binary      4       10         0
MAX              (Same as tested field)
MIN              (Same as tested field)
```

ORDER BY

This item is optional. It specifies the order in which the requested records are grouped. When **ORDER BY** is not specified, record transfer is not done according to a certain order.

Records are grouped according to the field specified first. Those records having the same value in each field specified first are grouped by the field specified second, and so on. Records containing null values are grouped after all records without null values have been grouped.

For example, you can specify:

```
DEPT,NAME,PHONE
```

In this case, records are first grouped according to DEPT. Then, the records having the same value for DEPT are grouped by NAME. The records with the same DEPT and NAME values are finally grouped by PHONE.

When a field name is specified in **ORDER BY**, it must also have been specified in **SELECT**, or **SELECT*** must have been specified.

Fields can be grouped in ascending or descending order. To do this, specify one blank after a field name then enter ASC or DESC. The default value is ASC. For example, specify:

```
DEPT DESC, NAME ASC
```

This indicates that the DEPT fields are to be grouped in descending order, after which the NAME fields are to be grouped in ascending (alphabetic) order.

Absolute values (ABS) can be specified for numeric fields. To do this, add a blank after a field name then enter ABS. For those fields having negative values, the negative signs are ignored and the absolute values are used.

The total length of the fields to be specified must not exceed 120 digits.

Return Record at Missing Field Value

When joining records from several files, joining might fail because a record is missing. This item specifies whether records with missing fields are transferred.

When you specify that records with missing fields are to be transferred, the alternative values for the missing fields are transferred. These values are normally blanks for character fields and zeros for numeric fields.

When you do not specify transfer of records with missing fields, those records are not transferred.

Specify this item to transfer data records that have alternative values for missing fields.

Do not specify this item if data records that have alternative values for missing fields are not to be specified. In this case, only those data records created from those records that exist in all files specified in **FROM** are transferred.

TO

Output device

This item specifies where received data is to be sent.

Display

The received data is displayed on the screen.

Disk

The received data is written to a workstation diskette or hard disk file.

Printer

The received data is printed on the printer.

When **Disk** is selected as the output device, also specify the following items.

PC file

This item specifies the name of the workstation disk file or diskette file to which the data is to be written.

Replace old file

This item is always required. It specifies whether the records in the file specified by **PC File** are to be replaced with the transferred records.

The default value is **Replace old file**.

Workstation file type

This item is always required. It specifies the type of the workstation disk file or diskette file to which the transferred records are written.

The system default is PC code test.

Save transfer description

This item is always required. It specifies whether the workstation file description is written to a workstation file. This file description describes the transferred data and it is required to subsequently return data to the iSeries™, eServer™ i5, or System i5™.

The system default is **Save**.

Description file name

This item is always required. It appears only when **Save Transfer Description** is selected. The **File Description File Name** specifies the name of the workstation disk file or diskette file to which the file description is written.

This item automatically sets the desired file name. This file name is the same as that specified by the user for **TO**, but to which extension .FDF has been added. Extension .FDF indicates that this file is a file-description file.

The use of extension .FDF is recommended when using a unique file name. To specify a file name in this item, use the same format as that in **TO**. (Items inside brackets [ ] can be omitted.)

```
[d:][path-name]file-name[.ext]
```

After **Save File Description File** is specified or a name is specified for **File Description File Name**, the iSeries™→PC Transfer Request window reopens after the **Return** key is pressed. Using this screen, a transfer request can be changed, saved, or executed.

Saving, Opening, Changing, and Executing a Transfer Request

The following section explains how to save, open, change, and execute, as a file, the information (transfer request) on the data to be transferred.

Saving a Transfer Request

You should save a transfer request, especially when the request will be executed repeatedly. This eliminates the need to create a transfer request every time a request is executed. To save a transfer request, do as follows:

1. Specify the information needed for transfer, using the iSeries™→PC Transfer window. See Creating an iSeries-to-Workstation Transfer Request on page 115 for an explanation of how to specify the required data.
2. After specifying the necessary data, click **Save** or **Save As** from the File menu of the menu bar.

   The Save Transfer Request File As window opens.
3. Specify each item, referring to the following explanation, then click **OK**.

   File Name

   Disk to which data is to be saved. Specify a file name or diskette file name. The default extension is TTO. Extension TTO identifies a file as a transfer request file.

   Description

   This item can be used to add a short explanation of a transfer request, as required. The explanation can be up to 40 characters in length. This explanation is saved with the transfer request, and displayed in the list of transfer request names. It is useful, therefore, for identifying a transfer request.
4. The system asks whether the saved transfer request is to be registered in the PC400 folder.

   When you click **OK**, the transfer request is registered as an icon. Subsequently selecting this icon transfers data according to the registered data transfer request.

Opening and Changing a Saved Transfer Request

To open and change a saved transfer request:

1. Display the iSeries™→PC Transfer window.
2. Select **Open** from the **File** menu.

   The Open Transfer Request File window opens.
3. Specify the name of the file to be opened using the Open Transfer Request File window. Then click **OK**.

   The iSeries™→PC Transfer window reopens, with the information specified for each item for the transfer request displayed. This completes opening of the saved transfer request.
4. Change the contents, as necessary.
5. To save the changed contents, follow the procedure given in Saving a Transfer Request on page 133.

Executing a Transfer Request

You can execute a file transfer request in one of the following two ways:

- By selecting the icon with which the transfer request has been registered
- By using the iSeries™→PC Transfer window of the Data Transfer icon

Selecting the Icon with Which the Transfer Request Has Been Registered

This method can be used only when a transfer request has been saved as an icon by using the iSeries™→PC Transfer window.

Data transfer starts as soon as you select the icon with which a transfer request has been registered.

Using the iSeries→PC Transfer Window

1. Before attempting to execute a transfer request, all operations such as creating, opening, and changing a transfer request must have been completed.

   Note: When data is transferred from a workstation to an existing member in an iSeries™, eServer™ i5, or System i5™ file, the transferred data replaces the existing data in the member.
2. Select **Receive** from the iSeries™→PC Transfer window.

   Data transfer starts.
3. After the transfer has been completed, click **Cancel**, or click **Exit** from the **File** menu.

Status during Transfer

Display can be specified as the **output device**, when the current transfer request is created or changed. This sends the transferred record to the screen. On the screen, each record is displayed on one line.

Each field in a transferred record is converted from the iSeries™, eServer™ i5, or System i5™ data type to workstation code.

Note: The workstation receives the iSeries™, eServer™ i5, or System i5™ records in order and then writes them to a temporary file of the default directory in the default drive (usually, the directory in which PC400 is installed). The maximum number of records that can be transferred is 4096 records, limited by the amount of records that can be stored in free space of the default drive.

When **Disk** is selected as the **output device**, the following actions are performed:

1. The workstation file description is written to a workstation disk file or diskette file according to the **Save File Description File** specification. (If **Save File Description File** has not been specified, this procedure is not performed.)
2. The transferred records are written to a workstation disk file or diskette file.

Limited Usage of File Names and Field Names

For a transfer request from a workstation to the iSeries™, eServer™ i5, or System i5™, none of the following reserved words can be specified as a file name or field name:

CRTFILE	MBRTEXT
CRTMBR	PUBAUT
FILETEXT	RCDLEN
FILETYPE	REFFILE
INTO	

For a transfer request from the iSeries™, eServer™ i5, or System i5™ to a workstation, none of the following reserved words can be used as a file name or field name:

ABS	IS
AND	ISNOT
ASC	LIKE
AVG	MAX
BETWEEN	MIN
BY	NOT
COLUMNS	OPTIONS
COUNT	OR
DESC	ORDER
EXTRACT	PARTOUT
FROM	REPLACE
GROUP	SELECT
HAVING	SUBSTR

IN	SUM
INNER	TABLES
	WHERE

To use one of these reserved words as a file name or field name, use the reserved word in uppercase, enclosed in quotation marks:

```
TO MYLIB/"INTO"
```

Examples of Transfer Requests for Receiving

This section provides examples of transfer requests for receiving. The contents of this section provide supplementary information to help you better understand transfer requests for receiving.

This section describes how to transfer data from the iSeries™, eServer™ i5, or System i5™, based on the inventory control file INVENTORY and supplier file SUPPLIERS.

The INVENTORY file contains information about the various parts in stock. Each part has a three-digit identification number, PARTNUM. The INVENTORY file contains the names of parts (DESCRIPTION) and the quantity on hand (QONHAND) for each part.

```
        File: INVENTORY
 Field name: PARTNUM  DESCRIPTION  QONHAND
              -------  -----------  -------
    Record 1:     209          CAM       50
           2:     221         BOLT      650
           3:     222         BOLT     1250
           4:     231          NUT      700
           5:     232          NUT     1100
           6:     207         GEAR       75
           7:     241       WASHER     6000
           8:     285        WHEEL      350
           9:     295         BELT       85
```

The SUPPLIERS file contains information about the suppliers of each part. Each supplier is identified by a two-digit number, SUPPNO. The SUPPLIERS file contains the number of parts delivered (PARTNO), their prices (PRICE), times of delivery (DELIVTIME), and ordered quantities (QONORDER). The parts listed in the SUPPLIERS file are the same as those listed in the INVENTORY file.

```
        File: SUPPLIERS
  Field name: SUPPNO   PARTNO   PRICE   DELIVTIME   QONORDER
              ------   ------   -----   ---------   --------
    Record 1:     51      221     .30          10         50
           2:     51      231     .10          10          0
           3:     53      222     .25          15          0
           4:     53      232     .10          15        200
           5:     53      241     .08          15          0
           6:     54      209   18.00          21          0
           7:     54      221     .10          30        150
           8:     54      231     .04          30        200
           9:     54      241     .02          30        200
          10:     57      285   21.00          14          0
          11:     57      295    8.50          21         24
          12:     61      221     .20          21          0
          13:     61      222     .20          21        200
          14:     61      241     .05          21          0
          15:     64      207   29.00          14         20
          16:     64      209   19.50           7          7
```

Receiving Part of an iSeries, eServer i5, or System i5 File

Specify the following items:

Library/File (Member)	INVENTORY
SELECT	PARTNUM, QONHAND
WHERE	QONHAND < 100
ORDER BY	PARTNUM

In this case, only part of the INVENTORY file is to be transferred. Specifically, only the part number (PARTNUM) and quantity on hand (QONHAND) fields of the records for which the number of parts in stock is less than 100 (QONHAND < 100) are transferred. Records are transferred in ascending order of parts numbers (PARTNUM).
The following data is transferred:

```
     Field:   PARTNUM  QONHAND
              -------  -------
  Record 1:     207       75
         2:     209       50
         3:     295       85
```

Receiving Records Joined from Several iSeries, eServer i5, or System i5 Files

Two iSeries™, eServer™ i5, or System i5™ files, INVENTORY and SUPPLIERS, are assumed. Note that both files contain records including part number fields. The INVENTORY file contains inventory information about individual parts. The SUPPLIERS file contains information about purchasing and ordering.

You might want to transfer information on part numbers, part names, and the prices of the parts to be ordered from supplier 51. The desired fields are PARTNO (SUPPLIERS file), DESCRIPTION (INVENTORY file), and PRICE (SUPPLIERS file).

By comparing the data in the INVENTORY file and the SUPPLIERS file, the user can determine that supplier 51 provides part numbers 221 and 231, called BOLT and NUT, respectively, and that their prices are 30 cents and 10 cents, respectively. The following table summarizes this information:

```
     Field:   PARTNO   DESCRIPTION   PRICE
              ------   -----------   -----
  Record 1:    221     BOLT            .30
         2:    231     NUT             .10
```

The same results are available by joining the data in these two files by using the iSeries™→PC Transfer function. To do this, specify both files (INVENTORY and SUPPLIERS) in the **FROM** item. For **SELECT**, specify which fields are to be transferred (PARTNO, DESCRIPTION, and PRICE). For **WHERE**, specify which records are to be transferred (records for which SUPPNO = 51).

Respecify the relationship between the two files in **JOIN BY**. From these results, the user can determine, by checking the SUPPLIERS file, that part number 221 is delivered from supplier 51 at a cost of 30 cents. In addition, to determine the part name, the user must check the INVENTORY file for part number 221 and its product name. In other words, the user observes that data is joined from the records in both the SUPPLIERS file and the INVENTORY file and that those records have the same part number. Therefore, to link the two records in these files, the records must have the same part number.

In short, to obtain this information, specify:

Library/File (Member)	SUPPLIERS, INVENTORY
JOIN BY	PARTNO = PARTNUM
SELECT	PARTNO, DESCRIPTION, PRICE
WHERE	SUPPNO = 51
ORDER BY	PARTNO

Receiving Records Using File Qualifiers

To join records from several iSeries™, eServer™ i5, or System i5™ files, fields of the same type must be joined.

For example, the part number fields in the INVENTORY and SUPPLIERS files can have the same name PARTNO. To specify the desired PARTNO fields, you must specify which file contains those fields. To do so, file qualifiers are used.

A file qualifier is the character T (uppercase or lowercase) followed by a one- or two-digit number. Use a comma to delimit the file qualifier and field name. In the previous example, prefix T1. and T2. to the PARTNO field names. **T1.** indicates the first file of **FROM**, while T2. indicates the second.

To obtain the same information as in the previous example, specify:

Library/File (Member)	SUPPLIERS, INVENTORY
JOIN BY	T1.PARTNO = T2.PARTNO
SELECT	T1.PARTNO, DESCRIPTION, PRICE
WHERE	SUPPNO = 51
ORDER BY	T1.PARTNO

T1.PARTNO indicates the PARTNO fields in the SUPPLIERS file, while T2.PARTNO indicates the PARTNO fields in the INVENTORY file.

Qualifiers are not needed for the names of the DESCRIPTION, PRICE, and SUPPNO fields, because they exist in one file only. However, the user can specify the following qualifiers for clarity:

```
T2.DESCRIPTION, T1.PRICE, T1.SUPPNO
```

The following examples of joining several iSeries™, eServer™ i5, or System i5™ files describe more sophisticated techniques. You should now be familiar with the basics of how to join two files. For a more detailed explanation, refer to the following sections.

Receiving with Field Missing Records Joined

The joining of records from several files could fail because one or more records is missing. For example, the record containing part number 221 might not be found in the INVENTORY file. This means that the records that can be joined to the 1st, 7th, and 12th records in the SUPPLIERS file do not exist in the INVENTORY file. In this case, the PARTNO field and PRICE field for part number 221 can be determined, but the DESCRIPTION field cannot be determined. So, the DESCRIPTION field is missing.

To transfer field missing records, use **Return Record at Missing Field Value**.

When **Return Record at Missing Field Value** has been specified, the default iSeries™, eServer™ i5, or System i5™ values are transferred instead of the missing field values. The default values for character fields are blanks, while those for numeric fields are zeros. For example, if the INVENTORY file does not contain the part number 221 record, the result of the previous example will be as follows:

```
    Field: PARTNO   DESCRIPTION   PRICE
           ------   -----------   -----
 Record 1:    221                   .30
        2:    231   NUT             .10
```

If **Return Record at Missing Field Value** has not been specified, the field missing records are not transferred. For example, if the INVENTORY file does not contain the part number 221 record, the result of the previous example will be as follows:

```
    Field: PARTNO   DESCRIPTION   PRICE
           ------   -----------   -----
 Record 1:    231   NUT             .10
```

Receiving with Records in a Same File Joined

Records in the same file can be joined. In other words, a file can be repeatedly specified in **FROM**. For instance, data in certain records can be compared using this function.

For example, the SUPPLIERS file shows that several suppliers provide the same part. The user might want to know which supplier sets a price that is double, or greater than double, that of another. To transfer the necessary information to a workstation, specify:

Library/File (Member)	SUPPLIERS, SUPPLIERS
JOIN BY	T1.PARTNO = T2.PARTNO
SELECT	T1.PARTNO, T1.SUPPNO, T1.PRICE, T2.SUPPNO, T2.PRICE
WHERE	T1.PRICE > 2 * T2.PRICE
ORDER BY	T1.PARTNO

The same file has been specified in **FROM** twice. **JOIN BY** specifies that records having the same part number are joined. This creates a joined record containing information about two suppliers of a single part. The user can spot those records for which the price is double, or greater than double, that of another supplier.

Records in the **SUPPLIERS** file are compared, one by one, with all the records (including itself) in the **SUPPLIERS** file. When the same part number is found, the two corresponding records are linked. This processing is performed for each record in the **SUPPLIERS** file.

For each record, the first supplier's price is compared with the second supplier's price. When the first supplier's price is double, or greater than double, that of the second, only the record containing the first supplier price is kept.

The final result is as follows:

```
   Field:  T1.PARTNO   T1.SUPPNO   T1.PRICE   T2.SUPPNO   T2.PRICE
           ---------   ---------   --------   ---------   --------
Record 1:       221          51        .30          54        .10
       2:       231          51        .10          54        .04
       3:       241          53        .08          54        .02
       4:       241          61        .05          54        .02
```

Specifying Records To Be Included in a Group

You might want to limit which records will be included in a group. To do so, use **WHERE**. The following example transfers the average and lowest prices of each part for those records for which the delivery time (DELIVTIME) is less than 30 days.

Library/File (Member)	SUPPLIERS
GROUP BY	PARTNO
SELECT	PARTNO, AVG(PRICE), MIN(PRICE)
WHERE	DELIVTIME < 30

The result is as follows:

```
   Field: PARTNO   AVG(PRICE)   MIN(PRICE)
         ------   ----------   ----------
Record 1:   221          .25          .20
       2:   231          .10          .10
       3:   222          .23          .20
       4:   232          .10          .10
       5:   241          .07          .05
       6:   209        18.75        18.00
       7:   285        21.00        21.00
       8:   295         8.50         8.50
       9:   207        29.00        29.00
```

Note that the conditions specified in **WHERE** are checked first, then the records that satisfy those conditions are included in the group.

Specifying Summary Records To Be Transferred

In some cases, you might want to transfer only summary records that satisfy certain conditions. The use of **HAVING** enables the selection of which summary records are to be transferred. **WHERE** is applied to certain records in a group, while **HAVING** is applied only to summary records.

The following example transfers the highest and lowest prices for each part. However, the summary records to be transferred are only those for which the highest price exceeds 10.00.

Library/File (Member)	SUPPLIERS
GROUP BY	PARTNO
SELECT	PARTNO, MAX(PRICE), MIN(PRICE)
HAVING	MAX(PRICE) > 10.00

The following table shows the result of removing unnecessary summary records by using **HAVING**

```
    Field: PARTNO   MAX(PRICE)   MIN(PRICE)
           ------   ----------   ----------
 Record 1:    209        19.50        18.00
        2:    285        21.00        21.00
        3:    207        29.00        29.00
```

One summary record for an entire file can be transferred. To do this, specify only the summary function in **SELECT** and nothing in **GROUP BY**. As a result, an entire file can be recognized as one group, while one summary record can be transferred for the group.

You can concurrently use the concept of summarizing groups and that of joining records from several files. To obtain the desired results, do as follows:

1. Specify a file in **FROM**, and specify the join conditions to join the records in **JOIN BY**.
2. Specify the conditions in **WHERE** to remove unnecessary records.
3. Specify the fields used for grouping the remaining records in **GROUP BY**.
4. Specify the function in **SELECT**, then create summary records.
5. Specify the conditions in **HAVING** to remove unnecessary records.
6. Specify the items for grouping the final summary records in **ORDER BY**.

Functions Available from the Pull-Down Menu

The following section provides a simple explanation of the menu bar of the iSeries™→PC Transfer window and PC→iSeries™ Transfer window.

File

Transfer request files can be processed.

Create

Creates a transfer request file

Open

Displays the contents of an existing transfer request file

Save, Save As

Save the current settings to the transfer request file being used or to a new transfer request file, respectively

Exit

Terminates the operation started by selecting the **Data Transfer** icon

Setup (Only for iSeries→PC Transfer)

User Options

Time, date, and numeric value format for receiving can be specified.

Ignore Decimal Data Error

Specifies whether decimal data errors found in packed or zoned decimal fields upon executing requests are to be ignored. Selecting **Yes** to ignore decimal data errors and using existing indices can considerably reduce the time needed to execute a request. If this item is not specified, the transfer request creates indices again and modifies any detected decimal data errors. This requires extra processing time.

Time Format

Specifies a desired time format for fields of iSeries™, eServer™ i5, or System i5™ field type having a selected time. If no time format is specified, the default value in the workstation's national information file is used when the transfer request starts, and that in an existing transfer request is assumed when the request is called again.

Supported time formats are as follows:

HMS

Hours, minutes, seconds (hh:mm:ss)

ISO

International Standard Organization (hh.mm.ss)

USA

USA Standard (hh:mm AM or PM)

EUR

IBM® European Standard (hh.mm.ss)

JIS

Japanese Industrial Standard (hh:mm:ss)

DDS

iSeries™, eServer™ i5, or System i5™ DDS (Format given by iSeries™, eServer™ i5, or System i5™ file attribute)

DFT

iSeries™, eServer™ i5, or System i5™ default format (Host job default is used)

Time separator

Specifies enabled delimiters. The fields of the iSeries™, eServer™ i5, or System i5™ field type for the selected time must be in a format that supports delimiters.

When no delimiters are specified, the default value in the workstation's national information file is used when the transfer request starts, and that in an existing transfer request is assumed when the request is called again.

Supported time delimiters are as follows:

Colon

(:)

Period

(.)

Comma

(,)

Blank

( )

Null

(NULL) No Separator

Default value

(DFT) iSeries™, eServer™ i5, or System i5™ Default Separator

Date Format

Specifies the date format for fields of iSeries™, eServer™ i5, or System i5™ field type for the selected date.

If this date format is not specified, the default value in the workstation's national information file is used.

Supported values are as follows:

MDY

Month, day, year (mm/dd/yy)

DMY

Day, month ,year (dd/mm/yy)

YMD

Year, month, day (yy/mm/dd)

JUL

Julian (yy/ddd)

ISO

International Standard Organization (yyyy-mm-dd)

USA

USA Standards (mm/dd/yyyy)

EUR

IBM® European Standard (dd.mm.yyyy)

JIS

Japanese Industrial Standard (yyyy-mm-dd)

DDS

iSeries™, eServer™ i5, or System i5™ DDS (Format given by iSeries™, eServer™ i5, or System i5™ file attribute)

DFT

iSeries™, eServer™ i5, or System i5™ default format (Host job default is used)

Date separator

Specifies delimiters. The fields of the iSeries™, eServer™ i5, or System i5™ field type for the selected date must be in a format that supports delimiters.

When no date delimiters are specified, the default value in the workstation's national information file is used when the transfer request starts, and that in an existing transfer request is used when the request is called again.

Supported date delimiters are as follows:

Slash

(/)

Dash

(-)

Period

(.)

Comma

(,)

Blank

( )

Null

(Null) Delimiters are not used.

DFT

(DFT) iSeries™, eServer™ i5, or System i5™ default separator

Decimal separator

Specifies the decimal point character in an iSeries™, eServer™ i5, or System i5™ field whose type is packed decimal or zoned decimal.

When decimal points are not specified, the default value in the workstation's national information file is used when the transfer request starts, and that in an existing transfer request is used when the request is called again.

Supported decimal point delimiters are as follows:

Period

(.)

Comma

(,)

DFT

(DFT) - Default decimal separator

Sort Sequence

Specifies which sort sequence should be used for this transfer request.

iSeries job default

Sort by the table identified on the iSeries™, eServer™ i5, or System i5™ as the job sort table.

Hexadecimal

Sort by the internal hexadecimal representation.

User specified table

Sort by the table identified by the user in a subsequent prompt.

Shared Weight Table

Sort by the shared weight table associated with the language named in a subsequent prompt.

Unique Weight Table

Sort by the shared unique table associated with the language named in a subsequent prompt.

Changing the sort sequence affects the order in which records appear ***only*** if the **ORDER BY** clause is being used. The sort sequence affects all character comparisons that depend on the order of the alphabet. Such comparisons can occur in the **WHERE** clause, the **GROUP BY** clause, the **HAVING** clause, the **JOIN BY** clause, the **IN** predicate, the **LIKE** predicate, the **BETWEEN** predicate, the **MAX** function, and the **MIN** function. Comparison operations are =, <>, >, >=, and >=.

Sort Sequence Table Name

Type the name of the sort sequence table that you want to use for this transfer request. The format of the table name should be *library/table*. *LIBL and *CURLIB are allowed for the library name.

Translation Table

Translation tables for ASCII-to-EBCDIC translation or for EBCDIC-to-ASCII translation can be specified, created, and customized.

Current Table

Specifies whether the default translation or the user-defined translation table is to be used.

Host Code Page

Specifies the host code page to be used for translation.

Workstation Code Page

Specifies the workstation code page to be used for translation.

File Name

Specifies the file name of the user-defined table to be used for translation.

- To list all files in your workstation, click **Browse**.
- To customize the translation table, click **Customize**.

Signon Options

When configuring a Data Transfer session, the user specify the following signon parameters:

Use Kerberos principal, no prompting

This function enables Kerberos authentication, using the ticket generated by the Windows user credentials. This option is disabled by default.

Prompt as needed

The host prompt the user for signon information. For each host, the signon dialog is presented only once during the transfer session.

The user enable or disable the “Kerberos Signon Options” using the “Kerberos auto-signon” check box provided in the “Custom” installation UI. By default, “Kerberos auto-signon” is checked and “Kerberos Signon Options” is enabled.

The user disable the “Kerberos Signon Options” by unchecking the “Kerberos auto-signon” check box in the “Custom” installation UI.

For the “Typical” installation, “Kerberos Signon Options” is enabled by default and there are no UI options to enable or disable the “Kerberos Signon Options.”

Silent Installation

The user enable or disable the “Kerberos Signon Options” by setting the value of “KERBEROSSIGNON” property in the *custom.ini* file.

By default, “Kerberos Signon Options” is enabled and “KERBEROSSIGNON” property is set to “1” in the custom.ini file. The user disable the “Kerberos Signon Options” by setting the value of “KERBEROSSIGNON” property to “0” in the custom.ini. The user must pass *custom.ini* as an input during the reinstallation of Personal Communications.

Note: The user is not able to enable/disable the "Kerberos Signon Options" for the Refresh Pack Installer.

File-Description Files

A file-description file is a workstation file that contains all field descriptions of the data in the corresponding workstation data file. Each field descriptor contains the field name, data type, and field length. There is one field descriptor for each field in the workstation file.

A file-description file defines the following:

- The file type of the workstation file to be transferred. For an explanation of each file type, see Creating a File-Description File on page 148.
- The field names and order of these fields in each data record.
- The data type of each field in the workstation file.
- The size and number of decimal places of each field.

The workstation files require field definitions when the files are transferred. The field definitions describe the file as it exists on the workstation. These definitions contain data that is similar to the field definitions (DDS) required by iSeries™, eServer™ i5, or System i5™ files. The data must be defined for both the iSeries™, eServer™ i5, or System i5™ and the workstation files, because the field names from each file are needed to send the data to the iSeries™, eServer™ i5, or System i5™ and the data in each file might be in different formats.

A file-description file is created on request during the transfer process of data from an iSeries™, eServer™ i5, or System i5™ file to a workstation file. Therefore, you usually do not need to worry about the contents or the format of the file-description file. However, if you transfer data that has not been previously transferred to the system, you must create a file-description file.

Creating a File-Description File

You can create a file-description file using a workstation text editor. The file-description file must be an ASCII text file. Therefore, each record must end with a carriage return (CR) character (hex 0D) followed by a line feed (LF) character (hex 0A). All tab characters (hex 09) are treated as ASCII spaces. The last byte of the file must contain an end-of-file (EOF) character (hex 1A). Workstation editors that create ASCII text files usually use these special character designators, so normally you do not need to be concerned about them.

File-Description File Format

The format of the file-description file is as follows:

```
PCFDF [comment]
PCFT file-type-indicator [comment]
PCFO time-format,time-separator, date-format, date-separator, decimal-separator [comment]
PCFL field-name-1 data-type-1 length-1[/decimal-position-1][comment]
    .
    .
    .
PCFL field-name-n data-type-n length-n[/decimal-position-n][comment]
[* comment]
```

Items within brackets are optional. Use either uppercase or lowercase characters anywhere in the file.

PCFDF Entries

PCFDF is a keyword that identifies this file as a workstation file-description file. It must appear in the first line of the file, starting in column 1. A comment is the only other entry allowed on the first line. If you type a comment, it must be separated from the PCFDF keyword by a space.

PCFT Entries

PCFT is a keyword that identifies this record as containing the file type indicator. It is followed by an indicator identifying the type of file in which the data is stored. It must appear only once, and must start in column 1, after the PCFDF record and before any PCFL records. An optional comment can follow this file-type indicator if separated from the indicator by at least one space.

Following is an example of a PCFT entry:

```
PCFT 4 BASIC RANDOM FILE
```

Table 18: File-Type Indicators on page 148 shows the valid file-type indicators.

Table 18. File-Type Indicators

Indicator	File Type
1	ASCII text

Table 18. File-Type Indicators (continued)

Indicator	File Type
2	DOS random
3	BASIC sequential
4	BASIC random
5	Data interchange format (DIF**)
6	No-conversion file
7	Reserved
8	DOS random type 2
9	BIFF format

PCFO Entry

The PCFO entry is optional. PCFO is a keyword that identifies this record as containing information about the date and time formats, time stamp, and separator characters for applicable formats. It must appear only once and must start in column 1, after the PCFT record and before any PCFL records. If there is no PCFO entry, the information or characters assigned as defaults for the host system are used.

Table 19: Time Formats on page 149 shows the valid time formats.

Table 19. Time Formats

Indicator	Format Name	Time Format
1	HMS	hh:mm:ss
2	ISO - International Standards Organization	hh.mm.ss
3	USA - USA standard	hh:mm AM or PM
4	EUR - European	hh.mm.ss
5	JIS - Japanese Industrial Standard Christian Era	hh:mm:ss
6	DDS	Format given by iSeries™, eServer™ i5, or System i5™ file attribute
7	DFT	Host job default is used
*	Unspecified	Host job default is used

Table 20: Time Separators on page 149 shows the valid time separators.

Table 20. Time Separators

Indicator	Separator
1	Colon (:)
2	Period (.)
3	Comma (,)
4	Blank ( )

Table 20. Time Separators (continued)

Indicator	Separator
5	Null (N)
6	Default (D) (host job default)
*	Unspecified (host job default)

Table 21: Date Formats on page 150 shows the valid date formats.

Table 21. Date Formats

Indicator	Format Name	Date Format
1	MDY	mm/dd/yy
2	DMY	dd/mm/yy
3	YMD	yy/mm/dd
4	Julian	yy/ddd
5	ISO	yyyy-mm-dd
6	USA	mm/dd/yyyy
7	EUR	dd.mm.yyyy
8	JIS	yyyy-mm-dd
9	DDS	Format given by iSeries™, eServer™ i5, or System i5™ file attribute
10	DFT	Host job default is used
*	Unspecified	Host job default is used

Table 22: Date Separators on page 150 shows the valid date separators.

Table 22. Date Separators

Indicator	Separator
1	Slash (/)
2	Dash (-)
3	Period (.)
4	Comma (,)
5	Blank ( )
6	Null (N)
7	Default (D) (host job default)
*	Unspecified (host job default used)

Table 23: Decimal Separators on page 150 shows the valid decimal separators.

Table 23. Decimal Separators

Indicator	Separator
1	Period (.)

Table 23. Decimal Separators (continued)

Indicator	Separator
2	Comma (,)
*	Unspecified (workstation default used)

Following is an example of a PCFO entry:

```
PCFO 1,1,1,1,1 OPTIONS SETTINGS
```

PCFL Entries

PCFL identifies a definition for a field. Enter a PCFL entry in the file-description file for each field in the data file. The PCFL records must be in the same order as the fields they define in the data file.

Define as many as 256 PCFL records in the file-description file and start PCFL records in column 1. If you enter more than 256 PCFL records, you receive an error message. You cannot continue a record on one line, and only the first 80 characters of a record are used.

Following is an example of a PCFL entry:

```
PCFL CUSTNAME 1 20  CUSTOMER NAME
```

Each PCFL entry contains the following things:

- The keyword, PCFL, starting in column 1 and followed by a space. This identifies the record as a field description.
- The field name, followed by a space. This must match the name that exists in the field definitions on the iSeries™, eServer™ i5, or System i5™ and can be from 1 to 10 characters.
- The indicator for the data type. Table 24: Data Type Indicators on page 151 shows the indicators that represent the data type of the data in the field. Follow the specified indicator with a space.
- The size of the field (in bytes) as it is stored in the workstation file. The length specification can be from 1 to 4 characters.

Table 24. Data Type Indicators

Indicator	Data Type
1	ASCII[1]
2	ASCII numeric
3	Hexadecimal
4	Binary
5	Zoned
6	Packed
7	BASIC integer
8	BASIC single-precision floating point

Table 24. Data Type Indicators

(continued)

Indicator	Data Type
9	BASIC double-precision floating point
10	EBCDIC
11	EBCDIC zoned
12	EBCDIC packed

[1] Includes date, time, and time stamp except for files that are not converted.

The data type indicator you enter must be valid for the file type entered earlier. Any other data types are not valid and are diagnosed as errors during a data transfer to the iSeries™, eServer™ i5, or System i5™.

Table 25: Valid SBCS Data Types for File Types on page 152 shows the valid single-byte character set (SBCS) data types for each file.

Table 25. Valid SBCS Data Types for File Types

File Type	Valid Data Type
ASCII text	ASCII ASCII numeric
DOS random	ASCII Binary Hexadecimal ASCII Packed Zoned
BASIC sequential	ASCII ASCII numeric
BASIC random	ASCII BASIC double-precision floating point BASIC integer BASIC single-precision floating point Hexadecimal
DIF	ASCII ASCII numeric
No-conversion	Binary EBCDIC EBCDIC packed EBCDIC zoned Hexadecimal
DOS random type 2	ASCII Binary Hexadecimal Packed Zoned
BIFF format	ASCII ASCII numeric

Note: ASCII (SBCS) includes date, time, and time stamp types if converted. EBCDIC includes date, time, and time stamp if not converted.

For numeric fields in BASIC sequential and DIF files, a size specification must be present. However, because the data in these fields is of variable length, the data transfer function assumes a maximum length of 65 characters. This length more than covers the largest possible exponential ASCII numeric value. The size specifications for character fields must be the maximum size of any data item in that field.

Table 26: Allowable Data Length Limits for Personal Computer SBCS Data Types on page 153 shows the allowed data length limits for each workstation data type. These are the maximum lengths you can specify for size in the PCFL entry.

Table 26. Allowable Data Length Limits for Personal Computer SBCS Data Types

Personal Computer Data Type	**Data Length Limit (in Bytes)**
ASCII	4093
ASCII numeric	33 (65 for DIF and BASIC sequential)
BASIC double-precision	8 (only allowed length)
BASIC integer	2 (only allowed length)
BASIC single-precision	4 (only allowed length)
Binary	4
EBCDIC	4093
Hexadecimal	2048
Packed decimal (ASCII and EBCDIC)	16
Zoned decimal (ASCII and EBCDIC)	31
Time HMS [1] USA ISO, EUR, and JIS [1] DDS, DFT	• 8 • 8 • 8 • 8 or 10 [2]
Date MDY, DMY, YMD Julian ISO, EUR, JIS, USA (see note 1) DDS, DFT	• 8 • 6 (only allowed length) • 10 • 6, 8, or 10 [2]
Time stamp	• 26

Notes:

1

These abbreviations appear in the time and date parameter sections.

HMS

Hours Minutes Seconds

Table 26. Allowable Data Length Limits for Personal Computer SBCS Data Types

(continued)

Personal Computer Data Type	Data Length Limit (in Bytes)
EUR IBM® European Standard **JIS** Japanese Industrial Standard Christian Era **ISO** International Standards Organization **2** The length is determined by the format defined in the host file for DDS, or from the iSeries™, eServer™ i5, or System i5™ job default (DFT keyword).	

Table 27: Allowable Data Length Limits for iSeries, eServer i5, or System i5 Data Types on page 154 shows the allowed data length limits for each iSeries™, eServer™ i5, or System i5™ data type.

Table 27. Allowable Data Length Limits for iSeries™, eServer™ i5, or System i5™ Data Types

iSeries, eServer i5, or System i5 Data Type	Data Length Limit in Bytes [1]
Binary	2 or 4 (only allowed lengths)
EBCDIC	4096
Hexadecimal	2048
Packed decimal (EBCDIC)	16
Zoned decimal (EBCDIC)	31
Time	
HMS	8
USA	8
ISO, EUR, and JIS	8
DDS, DFT	8 or 10 [2]
Date	
MDY, DMY, YMD	8
Julian	6 (only allowed length)
ISO, EUR, JIS, USA	10
DDS, DFT	6, 8, or 10 [2]
Time stamp	26
Notes:	

Table 27. Allowable Data Length Limits for iSeries™, eServer™ i5, or System i5™ Data Types

(continued)

iSeries, eServer i5, or System i5 Data Type	Data Length Limit in Bytes [1]
1 The data length limits for the workstation and the system data fields are different in some cases. For these cases, the transfer function attempts to fit the workstation data into the system field. If the data does not fit into the field, a message is displayed. Refer to Data Conversions on page 156 for more details. **2** The length is determined by the format defined in the host file for DDS, or from the iSeries™, eServer™ i5, or System i5™ job default (DFT keyword).	

If there is a decimal position associated with the data in that field, place a forward slash (/) and then the number of decimal positions after the length specification. There are no spaces between the length, slash, and decimal position specifications.

The decimal position specification refers to the number of positions from the right-hand byte of the resulting decimal number. Do not specify a decimal position for floating-point numbers unless the data type is one of the following types:

- ASCII numeric
- Binary
- Packed
- Zoned

Note: The number of decimal positions in a field ranges from 0 to 9 or the maximum number of decimal digits in this number, whichever is smaller. The data transfer function might round the number to fit it into the field. Refer to Data Conversions on page 156 for more details.

Comment Entries

Enter comment lines anywhere in the file-description file, observing the following restrictions:

- The last element of the field-descriptor entry specification is a comment. This is an optional entry for your information only, and must be separated from the size entry by a space. PCFL entries created by the data transfer function (RTOPC) do not contain a comment field.
- Precede the comment with an asterisk (*) as the first nonspace character in the line.
- Do not exceed 80 characters in length.
- Do not make the comment the first record in the file-description file.

Following is an example of a comment:

```
* This is a comment
```

File-Description File Example

Following is an example of a file-description file for an inventory file:

```
PCFDF
PCFT 3 BASIC SEQUENTIAL FILE
* ITEM INVENTORY FILE
PCFO 1,1,1,2,1 OPTIONS SETTINGS
PCFL ITEMNO 2 8  ITEM NUMBER
PCFL ITEMDESC 1 20  DESCRIPTION OF ITEM
PCFL COLOR 1 8  COLOR
PCFL WEIGHT 2 7/2  ITEM WEIGHT
PCFL PRICE 2 7/2  PRICE PER ITEM
PCFL INSTOCK 2 6  ITEMS IN STOCK
```

Data Conversions

The data transfer function needs data conversions for transferring data from the system to the workstation, and vice versa. For both types of transfers, the necessary conversion depends on the record size, the type of data being transferred, the type of workstation file being used, the system data type, and, in some cases, the data length.

Record Size

Each transferred record contains data indicating whether each field contains a null value. There is a restriction on the maximum data record that can be sent or received from the iSeries™, eServer™ i5, or System i5™ because of this data.

The following formula determines the maximum record length that can be transferred:

- 4096 - (number of fields in the record + 2) = (maximum record length)

Data Types

The data transfer function supports the following system data types:

- Date
- Time
- Time stamp
- Binary data
- Character data
- Hexadecimal data

- Packed decimal data
- Zoned decimal data

The data transfer function supports the following workstation data types:

- BASIC numeric data, including:
    - Double-precision data
    - Integer data
    - Single-precision data
- Binary data
- Character data, including:
    - ASCII
    - EBCDIC
- Hexadecimal data
- Packed decimal data
- Zoned decimal data
- ASCII numeric data

Date, Time, and Time-Stamp Data Types

Date, time, and time-stamp values can be used in certain arithmetic and character operations and are compatible with certain character constants, but they are neither characters nor numbers.

A date is a three-part value (year, month, and day) designating a point in time on the calendar. The range of the year is 0001 to 9999. The range of the year for a non-SAA format is 1940 to 9999. The range of the month is 1 to 12. The range of the day is 1 to *x*, where *x* depends on the month.

A time is a three-part value (hour, minute, and second) designating a time of day under a 24-hour clock. The range of the hour is 0 to 24 and the range of the other values is 0 to 59.

A time stamp is a seven-part value (year, month, day, hour, minute, second, and microsecond) that designates a date and time including the specified microseconds. The maximum length of the time stamp is a character string of 26.

Dates, times, and time stamps can be assigned to result fields. A valid character-string representation of a date can be compared with a date field, or a valid character-string representation of a time can be compared with a time field.

BASIC Numeric Data

Double-Precision Data

Double-precision data is defined only for the workstation. The iSeries™, eServer™ i5, or System i5™ does not support this data type. BASIC applications use double-precision data. This data type is a positive or negative number from $2.938735877055719 \times 10^{-39}$ to $1.701411834604692 \times 10^{38}$. Double-precision numbers are stored in 8 bytes, with 7 bytes representing the mantissa and 1 byte representing the exponent.

Integer Data

Integer data is defined only for the workstation. BASIC applications use integer data. Integer data is stored in 2 bytes and represents a whole number from -32768 to 32767.

Single-Precision Data

Single-precision data is defined only for the workstation. The iSeries™, eServer™ i5, or System i5™ does not support this data type. BASIC applications use single-precision data. This data type is a positive or negative number from 2.938736 x 10^{-39} to 1.701412 x 10^{38}. Single-precision numbers are stored in 4 bytes, with 3 bytes representing the mantissa and 1 byte representing the exponent and sign.

Binary Data

This data represents signed or unsigned numbers in twos complement form. Binary numbers of 1, 2, 3, or 4 bytes in length are allowed on the workstation, but the iSeries™, eServer™ i5, or System i5™ allows only numbers 2 or 4 bytes in length. The bit on the left side of the high-order bit determines the sign of the number (0 for positive, 1 for negative). The system stores the data with the high-order byte on the left side of the field, whereas the workstation stores the data with the high-order byte in the right-hand position of the field.

The decimal position, if specified by the file description, represents the number of decimal digits to the right of the decimal point. The file description specifies the presence of a decimal position.

For example, the binary number 3BF5 is equivalent to the decimal number 15349, and the binary number FFB4 is equivalent to the decimal number -76.

Character Data for SBCS

You can think of this data as a string of bits that represents particular characters and symbols.

The tables used to translate characters from ASCII to EBCDIC and from EBCDIC to ASCII contain the following kinds of values:

- Values where the workstation ASCII characters and iSeries™, eServer™ i5, or System i5™ EBCDIC characters match exactly
- Values where a substitute character is chosen for a character that cannot be translated

The data transfer function uses tables to translate data from ASCII to EBCDIC and EBCDIC to ASCII. You can change these default tables using the translation table utility (TRTABLE).

Note: ASCII (SBCS) data includes date, time, and time stamp types if converted. EBCDIC data includes date, time, and time stamp if not converted.

Hexadecimal Data

You can think of this data as a string of bits representing base 16 numbers. For example, you can represent hex 3D with the following string of bits:

0011 1101

Packed Decimal Data

For both the iSeries™, eServer™ i5, or System i5™ and the workstation, each half-byte represents a value from 0 through 9. The hexadecimal value in the half-byte on the right side of the right-hand byte specifies the sign.

For the iSeries™, eServer™ i5, or System i5™, a value of hex B or hex D in this half-byte represents a negative number.

For DOS random files, only the last half-byte (the half-byte that contains the sign) is changed. For the sign half-byte, the workstation uses hex 3 to indicate a positive number or hex B to indicate a negative number.

For example, X'0865431F' appears as X'08654313'.

For DOS random type-2 files, the last half-byte (the half-byte that contains the sign) is not changed. The sign convention used on the workstation and on the host system is the same.

For example, X'0865431C' appears as X'0865431C'.

The decimal position, if specified, represents the number of decimal digits to the right of the decimal point. The presence of a decimal position is specified in the file description.

Zoned Decimal Data

This data is represented in a form in which each byte corresponds to one decimal digit. Each of these bytes is stored in character form. For example, the digit 7 is stored on the iSeries™, eServer™ i5, or System i5™ as F7, which is the EBCDIC representation, and is stored on the workstation as 37, which is the ASCII representation.

The size of each digit is determined by its half-byte on the right side. Valid values for the half-bytes are decimal 0 through 9.

The sign in both the iSeries™, eServer™ i5, or System i5™ and workstation zoned decimal fields is specified by the hexadecimal value in the left half-byte of the right byte of the field. For the iSeries™, eServer™ i5, or System i5™, a hex B or hex D in this half-byte represents a negative number (for example, X'F6D2' represents -62).

For DOS random files, zoned decimal fields from the system change from EBCDIC to ASCII, as do character fields, except that the sign half-byte in the workstation field is changed to a hex 3 to indicate a positive number or a hex B to indicate a negative number.

For DOS random type-2 files, zoned decimal fields from the system change from EBCDIC to ASCII, as do character fields, except that the sign half-byte in the workstation field is changed to a hex 3 to indicate a positive number or a hex 7 to indicate a negative number.

The decimal position, if specified, represents the number of decimal digits to the right of the decimal point and is specified by the file description.

ASCII Numeric Data

The data transfer function defines ASCII numeric data to represent any numeric value stored in ASCII format. This is not a valid iSeries™, eServer™ i5, or System i5™ system data type. The number -123.45 in ASCII format is:

2D 31 32 33 2E 34 35

The decimal point and sign are stored explicitly for ASCII numeric data. The character on the left displays the sign (space or plus (+) for positive, minus (-) for negative). Leading zeros to the left of the decimal point change to spaces. The decimal point, if any, is added in the correct position.

BASIC sequential and DIF file types also support another form of ASCII numeric data called exponential numbers.

An exponential number is a decimal number followed by the letter E or D and a signed integer of two or three digits. E represents a single-precision number and D represents a double-precision number. The exponent portion (E or D and the integer) represents *"times 10 to the power of the integer specified"*.

For example, the number -1.0E+03 (representing -1.0 x 10^{3} in ASCII numeric format) is:

2D 31 2E 30 45 2B 30 33

For example, the number 9.5D-15 (representing 9.5 x 10^{-15} in ASCII numeric format) is:

39 2E 35 44 2D 31 35

Personal Computer File Types

The following workstation file types are supported:

- ASCII text files
- BASIC random files
- BASIC sequential files
- DIF files
- BIFF files
- DOS random files
- DOS random type-2 files
- No-conversion files

ASCII Text Files

ASCII text files are normally used with programs that work with text (such as editors and print routines). The characteristics of an ASCII text file are as follows:

- Records consist of ASCII characters.
- A carriage return character (hex 0D) and a line feed character (hex 0A) delimit each record from the next.
- Workstation records in an ASCII file can be variable in length due to truncation of trailing blanks at the end of an iSeries™, eServer™ i5, or System i5™ record.

Transferring Data to ASCII Text Files

When you create an ASCII text file, the data coming from the iSeries™, eServer™ i5, or System i5™ changes as follows:

- Hexadecimal fields change to equivalent ASCII characters for each half-byte. For example, X'D3' expands to ASCII 4433 and is written to the file. When displayed by an editor or printed, the string appears as D3.
- EBCDIC character fields change byte by byte and are mapped into ASCII characters as defined by the translation tables.
- Date, time, and time-stamp data is mapped into ASCII characters as defined by the translation tables.
- Variable-length and null fields are converted to fixed lengths, and trailing blanks (for character, hexadecimal, date, time, and time-stamp data) or zeros (for binary, zoned, and packed,) are added to the maximum length of the field.

  Note: Some nondisplayable EBCDIC characters are translated into ASCII control characters on the workstation. If EBCDIC character fields contain nondisplayable data, you might get unexpected results and your ASCII text file might appear to be corrupted.

  For example, X'05' in an EBCDIC field is translated to an ASCII X'09', which is an ASCII control character for horizontal tab. Most workstation text editors process this tab character so that the data in your workstation text file appears to be shifted to the right when viewed.

  One possible solution to this problem is to define these fields on the host system as hexadecimal fields instead of character fields.

- Binary fields change to ASCII numeric. For example, X'FFD3' with no decimal position expands to ASCII 20202020202020202D3435. When displayed by an editor or printed, the string appears as -45.

  Note: The length of the ASCII field depends on the length of the binary field.

  A binary field on the iSeries™, eServer™ i5, or System i5™ is either 2 or 4 bytes long. The resulting ASCII field length is from 6 to 11 bytes, including the sign. Another byte is added for a decimal point.

  Table 28: Binary-to-ASCII Field Length Mapping on page 162 shows the mapping between binary field lengths and their ASCII lengths.

Table 28. Binary-to-ASCII Field Length Mapping

Binary Length	ASCII Length	Value Range
2	6	-32768 to 32767
4	11	-2147483648 to 2147483647

- Zoned decimal fields are changed to ASCII numeric. For example, EBCDIC F0F0F9F5F2D6 with a field length that indicates two digits to the right of the decimal point expands to ASCII 20202D39352E3236. When displayed by an editor or printed, the string appears as -95.26. The resulting workstation field length is equal to the length of the system field plus 1 for the sign and 1 for the decimal point, if specified.
- Packed decimal fields change to ASCII numeric. For example, X'871D' (no decimal point) changes to ASCII 2D383731. When displayed by an editor or printed, the string appears as -871.

  Since two decimal digits are packed into 1 byte, the length of the resulting workstation field is equal to two times the length of the iSeries™, eServer™ i5, or System i5™ field, plus 1 for the decimal point (if specified). This length always includes the sign. A minus sign (-) indicates negative, and a space indicates positive.

Transferring Data from ASCII Text Files

When you transfer data from ASCII text files to system files, the data changes as follows:

- ASCII character data changes to EBCDIC character, date, time, or time-stamp data (based on the iSeries™, eServer™ i5, or System i5™ field type) on a byte-to-byte basis, or to hexadecimal data by changing 2 ASCII bytes into 1 hexadecimal byte.
- ASCII numeric data changes to iSeries™, eServer™ i5, or System i5™ binary, zoned decimal, or packed decimal data, depending on the specified data type.

  The field lengths on the iSeries™, eServer™ i5, or System i5™ and the workstation are different because of the explicit way minus signs and decimal points are stored in ASCII numeric fields. Each field changes individually, to ensure that the resulting field length matches the specifications for that field. The data transfer function tries to fit the workstation data into the system field.
- For null-capable iSeries™, eServer™ i5, or System i5™ fields, null values (except date, time, and time stamp) cannot be reliably detected and are not uploaded. For variable-length iSeries™, eServer™ i5, or System i5™ fields, trailing blanks are removed and the field is converted to the variable-length format.

Errors When Transferring Data from ASCII Text Files

When you transfer data from a workstation ASCII text file to an iSeries™, eServer™ i5, or System i5™ file, the following errors can occur:

- A data field in the ASCII text file is too long for a field in the iSeries™-, eServer™ i5-, or System i5™-defined file. In this case, the data is truncated. This occurs when the description file defines the character data as longer than the field length specified for the system file.

  If the data transfers to an EBCDIC field, this error occurs only if the extra bytes are not spaces.

If the data transfers to a hexadecimal field, this error occurs only if the extra bytes are not zeros. These extra bytes are truncated so the data fits into the specified field.

- The value of numeric data is too large for the system field. The maximum value is used. This error occurs when:
    - Numeric data in the field does not fit into the specified number of bytes for the field.
    - The decimal value of a numeric field contains more digits than were specified for the field.

  The value of the field is set to the maximum value possible for the number of bytes and digits specified by the iSeries™, eServer™ i5, or System i5™.
- Data in this field has too many decimal positions. The number is rounded. This error occurs when the number of decimal positions in the field is greater than the number of decimal positions specified on the iSeries™, eServer™ i5, or System i5™. These extra bytes are significant because the data rounds up if the first extraneous digit is 5 or greater, and rounds down if it is less than 5.
- Data in this field is incorrect or does not match the data type. This error occurs when:
    - Nonnumeric data is found in a field that the file descriptions defined as numeric. The transfer request ends to prevent transferring incorrect data to the file.
    - ASCII numeric data is found that does not match the format the file description specified. An incorrectly positioned decimal point within the field could cause this error.
    - A value other than X'30' through X'39', minus, plus, or decimal point is found. A duplicated decimal point or minus is found. The transfer request ends to prevent transferring incorrect data to the file.
- Data for this field is missing. The default values are used. This error occurs when a data field is defined, but the data is not in the file. This means that the end of the record is reached before all of the defined data is found.

  The field or fields for which data has been defined but not found then fill with default values and transfer to the file. The default values are EBCDIC spaces for character fields, or zeros for numeric and hexadecimal fields.

  To supply your own default values, use the default (DFT) keyword in the data description specifications (DDS) for the file.
- Extra data is found at the end of this record. The extra data is not transferred. Data found at the end of this record and not defined by the system data definitions or workstation file-description file is not transferred to the system file, because no definitions exist to define the data and how it should change.

When you transfer data from an ASCII text file to an iSeries™, eServer™ i5, or System i5™ file without using a file-description file, any extra data found past the record length specified for the file is not transferred.

BASIC Random Files

BASIC random files are the most general-purpose BASIC file type. They contain fixed-length records with:

- No delimiters between fields or records
- No end-of-file marks

Transferring Data to BASIC Random Files

When you create a BASIC random file, system data changes as follows:

- Hexadecimal fields do not change.
- Change from a system binary field depends on the field length:
    - Fields of 2 bytes, with no decimal positions to the right of the decimal point, change to 2-byte BASIC integer values. The only change is that the order of the bytes reverses.
    - Fields of 2 bytes, with decimal positions to the right of the decimal point, change to BASIC single-precision numbers.
    - Fields of 4 bytes change to BASIC double-precision numbers.
- EBCDIC character, date, time, and time-stamp fields change byte by byte and are mapped into ASCII characters as defined by the translation tables.
- Variable-length and null fields are converted to fixed lengths, and trailing blanks (for character, hexadecimal, date, time, and time-stamp data) or zeros (for binary, zoned, and packed data) are added to the maximum length of the field.
- Zoned decimal fields change into one of the following BASIC variables depending on the field length and the number of decimal positions:
    - Zoned decimal fields of 4 bytes or less with no positions to the right of the decimal point change to a BASIC integer of an equivalent value.

      A zoned decimal field of 4 bytes or less, but with a decimal point, falls into the following category.
    - Zoned decimal fields up to 7 bytes (including those that did not fall into the previous category) change to a BASIC single-precision number of an equivalent value.
    - Zoned decimal fields greater than 7 bytes change to a BASIC double-precision number of an equivalent value.
- Packed decimal fields change into one of the following BASIC variables depending on the length of the field:
    - Packed decimal fields of 2 bytes or less with no positions to the right of the decimal point change to a BASIC integer of an equivalent value.

      A packed decimal field of 2 bytes or less, but with a decimal point, falls into the following category (up to 4 bytes).

    - Packed decimal fields of up to 4 bytes (including those that did not fall into the previous category) change to a BASIC single-precision number of an equivalent value.
    - Packed decimal fields greater than 4 bytes change to a BASIC double-precision number of an equivalent value.

Note: Changes between binary, packed decimal, and zoned decimal numbers with decimal points are not equivalent to their BASIC number counterparts, because BASIC uses a binary number format that does not always change into exact decimal fractions.

Transferring Data from BASIC Random Files

When you transfer data from BASIC random files to system files, the data changes as follows:

- Hexadecimal fields transfer to the system file as unchanged hexadecimal data. The field lengths as stored on the workstation should be the same as the field lengths as stored on the system.
- ASCII character, date, time, and time-stamp data changes to EBCDIC character data byte by byte.
- For null-capable iSeries™, eServer™ i5, or System i5™ fields, null values (except date, time, and time stamp) cannot be reliably detected and are not uploaded. For variable-length iSeries™, eServer™ i5, or System i5™ fields, trailing blanks are removed and the field is converted to the variable-length format.
- Numeric fields from BASIC random files (BASIC integers, single-precision floating-point numbers, and double-precision floating-point numbers) change to system binary data, zoned decimal data in EBCDIC format, or packed decimal data in EBCDIC format.

Note: Because the change of floating-point numbers into decimal fractions is not always exact, each number automatically changes into the most precise number possible with respect to the system field length. If you want more precision, specify a larger system field size.

Errors When Transferring Data from BASIC Random Files

When you transfer data from a workstation BASIC random file to a system file, the following errors can occur:

- Data in this field is too short for the system field. The data is padded. This error occurs when the file contains character or hexadecimal data shorter than the field length specified on the system. This error can occur if the workstation field is defined as shorter than the system, or if the data in the last record of the file is too short. Character fields are padded on the right with EBCDIC spaces, and hexadecimal fields are padded with zeros.
- Data in this field is too long for the system field. The data is truncated. This error occurs when the workstation file-description file defines character or hexadecimal data as longer than the field length specified on the system.

  For character data, this error occurs only if the extra bytes are not spaces. For hexadecimal data, this error occurs only if the extra bytes are not zeros. These extra bytes are then truncated so that the data fits into the specified iSeries™, eServer™ i5, or System i5™ field.

- The value of numeric data is too large for the system field. The maximum number is used. This error occurs when:
  - Numeric data in the workstation field does not fit into the specified number of bytes for the system field.
  - The decimal value of a numeric field contains more digits than are specified for the system field.
- Data in this field has too many decimal positions. The number is rounded down to zero. In BASIC random processing, this error occurs if the value of the number is too small to fit into the specified field.

  For example, the number 0.00001 does not fit into a system zoned field specified as being 2 bytes in length and 2 decimal positions to the right of the decimal point. In this example, the resulting value is zero.
- Data for this field is missing. The default values are used. This error occurs when a data field is defined, but the data is not in the file. This means that the end of the file is reached before all of the defined data is found. For BASIC random files, this error occurs only on the last record in the file, since there are no explicit record delimiters.

  When this error occurs, the field or fields for which data is defined, but not found, are filled with default values and are transferred to the iSeries™, eServer™ i5, or System i5™ file. These default values are EBCDIC spaces for character fields and zeros for numeric fields.

  To supply your own default values, use the Default (DFT) keyword in the DDS for the file.

When you transfer data from a BASIC random file to an iSeries™, eServer™ i5, or System i5™ file, any data shorter than the record length defined for the system file is padded with EBCDIC spaces.

Because there are no record delimiters in BASIC random files, this error can occur only on the last record of the file. This probably indicates that the record length of the system file does not match the record length of the workstation file.

BASIC Sequential Files

BASIC uses BASIC sequential files for sequential processing (for example, INPUT and WRITE statements). The fields written are considered either character or numeric. Characteristics of BASIC sequential files are as follows:

- Both numeric and character fields are written as displayable characters. However, character strings are distinguished from numeric strings by the ASCII double quotation marks (X'22') that surround them.

  Therefore, character data in BASIC sequential files cannot contain ASCII double quotation marks, because they are interpreted as the end of the character string.
- Fields are delimited by ASCII commas (X'2C'). Therefore, commas are not allowed as date, time, or decimal separators.
- Each record is delimited from the next by a carriage return character (X'0D') and a line feed character (X'0A'). The end-of-file character is X'1A'.
- Records and fields are variable in length.

Transferring Data to BASIC Sequential Files

The following list describes how iSeries™, eServer™ i5, or System i5™ data created by a BASIC-sequential-file-defined data definition changes:

- Hexadecimal fields change to equivalent ASCII characters for each half-byte. Double quotation marks surround them.

  For example, X'F3' expands to ASCII 22443322 and is written to the file.
- EBCDIC character, date, time, and time-stamp fields change byte by byte and are mapped into ASCII characters as defined by the translation tables. ASCII double quotation marks are added before and after the character string.
- Null fields are represented by the absence of the field (comma comma, or by a single comma if the null field is the last field of the record).
- For null fields, successive commas in the file will result in a null value being sent to the iSeries™, eServer™ i5, or System i5™ if the field is null-capable.
- In variable-length fields, if the iSeries™, eServer™ i5, or System i5™ field is variable length, the field is converted to the iSeries™, eServer™ i5, or System i5™ variable-length format.
- Binary fields change to ASCII numeric. Leading zeros to the left of the decimal point and trailing zeros to the right of the decimal point are removed.

  For example, X'FFD3' appears as ASCII 2D3435. When displayed on an ASCII device, the string appears as -45.
- Zoned decimal fields change to ASCII numeric. Leading zeros to the left of the decimal point and trailing zeros to the right of the decimal point are removed.

  For example, EBCDIC F0F0F9F5F2D6 with a field length that indicates two digits to the right of the decimal point expands to ASCII 2D39352E3236. The string appears as -95.26 when an editor displays it or it prints.
- Packed decimal fields change to ASCII numeric. Leading zeros to the left of the decimal point and trailing zeros to the right of the decimal point are removed.

  For example, X'871F' (no decimal point) changes to ASCII 383731. The string appears as 871 when an editor displays it or it prints.

Transferring Data from BASIC Sequential Files

When you transfer data from BASIC sequential files to iSeries™, eServer™ i5, or System i5™ files, the data changes as follows:

- ASCII character, date, time, and time-stamp data changes to EBCDIC character data on a byte by byte basis and to hexadecimal by changing 2 ASCII bytes into 1 hexadecimal byte.
- ASCII numeric data translates to system binary, zoned decimal, or packed decimal data, depending on the specified data type. The lengths of the system data and the workstation data might be different because the minus signs and decimal points are stored in ASCII numeric fields, and leading and trailing spaces are stripped away.

BASIC might create exponential numbers in these files. The data transfer function also changes these numbers.

Each translated field is individually verified to ensure that the resulting field length matches the specifications for that field. The data transfer function tries to fit the workstation data into the system field.

Errors When Transferring Data from BASIC Sequential Files

When you transfer data from a BASIC sequential file to a iSeries™-, eServer™ i5-, or System i5™-defined file, the following errors can occur:

- Data in this field is too long for the iSeries™, eServer™ i5, or System i5™ field. The data is truncated. The file-description file defines character data as longer than the field length specified for the file.

  If the data transfers to an EBCDIC field, this error occurs only if the extra bytes are not spaces. If the data transfers to a hexadecimal field, this error occurs only if the extra bytes are not zeros. These extra bytes are truncated so that the data fits into the specified iSeries™, eServer™ i5, or System i5™ field.
- The value of numeric data is too large for the system field. The maximum value is used. This error occurs when:
  - Numeric data in the workstation field does not fit into the specified number of bytes for the system field.
  - The decimal value of a numeric field contains more digits than were specified for the system field.

  The value of the field is set to the maximum value possible for the number of bytes and digits specified by the iSeries™, eServer™ i5, or System i5™.
- Data in this field has too many decimal positions. The number is rounded. This error occurs when the number of decimal positions in the workstation field is greater than the number of decimal positions specified on the system. The extra bytes are significant, because the data is rounded up if the first extraneous digit is 5 or greater, and is rounded down if it is less than 5.
- Data in this field is incorrect or does not match the workstation data type. This error occurs when a field defined as numeric by the file description contains nonnumeric data. This could also result if a character or hexadecimal field contains a numeric field, or if a numeric (zoned, packed, or binary) field contains a character field.

  When this error occurs, the transfer request ends to prevent transferring incorrect data to the system file.
- Data for this field is missing. The default values are used. This error occurs when a data field is defined, but the data is not in the file. This means that the end of the record is reached before all of the defined data is found.

  When this error occurs, the field or fields for which data has been defined, but not found, are filled with default values and transferred to the iSeries™, eServer™ i5, or System i5™ file. These default values are EBCDIC spaces for character fields, or zeros for numeric fields.

  To supply your own default values, use the default (DFT) keyword in the DDS for the file.

- Data in this field exceeds the workstation field size. The data is lost. This error occurs when extra data, not defined by the file-description file, is found at the end of a character field. The extra bytes are truncated and are not transferred to the system file.
- Extra data found at the end of the record. The extra data is not transferred. This error occurs when extra data is found at the end of the record, and has not been defined by the system data definitions or workstation file-description file. This extra data is not transferred to the system, because no definitions exist to define the data and describe how it should change.

Data Interchange Format Files

Data Interchange Format (DIF) files represent data in rows and columns. DIF files contain character and numeric data (positive and negative decimal numbers).

DIF is used for data interchange between spreadsheet programs and other application programs.

The data transfer function supports only the following two data types within DIF files:

- **Character data:** The data in a character cell (think of a *cell* as one field in one record) must be enclosed in double quotation marks if there is an embedded space in the string. However, if the string begins with a quotation mark, it must also end with a quotation mark.
- **Numeric data:** The numeric data supported by the data transfer function consists of a decimal number that can contain a minus sign or a decimal point or both. The data transfer function also supports exponential numeric data.

Transferring Data to DIF Files

When creating a DIF file, system data changes as follows:

- Hexadecimal fields change to equivalent ASCII characters for each half-byte. Double quotation marks surround them.
- EBCDIC character, date, time, and time-stamp data changes byte by byte and is mapped into ASCII characters as defined by the translation tables. ASCII double quotation marks are added before and after the character string.
- Binary fields change to ASCII numeric. Leading zeros to the left of the decimal point, and trailing zeros to the right of the decimal point, are removed.
- Zoned decimal fields change to ASCII numeric. Leading zeros to the left of the decimal point, and trailing zeros to the right of the decimal point, are removed.

  For example, EBCDIC F0F0F9F5F2D6 with a field length that indicates two digits to the right of the decimal point expands to ASCII 2D39352E3236. When displayed or printed, the string appears as -95.26.
- Packed decimal fields change to ASCII numeric. Leading zeros to the left of the decimal point, and trailing zeros to the right of the decimal point, are removed.

For example, X'871D' (no decimal point) changes to ASCII 2D383731. When displayed or printed, the string appears as -871.

- If untranslatable data is found, the entire field becomes an error cell. An error cell results when untranslatable data is found when a DIF file is created or when a not valid calculation is done using the DIF file with a spreadsheet program.

Transferring Data from DIF Files

If an error cell is found when data is transferred from a DIF file to the iSeries™, eServer™ i5, or System i5™, one of the following things can occur, depending on the type of data in the file:

- If the system field is a character (EBCDIC) field, it is filled with untranslatable characters (hexadecimal zeros) and is transferred to the system. A message appears, telling you how many bytes of untranslatable data have transferred.
- If the system field is a hexadecimal, zoned, packed, or binary field, you receive an error message telling you that the data in this cell is incorrect, and that the data was not transferred to the system.

When you transfer data from a system file to a DIF file, the field names are placed in the first record and you can consider them column headings. When you transfer DIF files back to the system, the first row must either be these field names (exactly as they are defined on the system) or data. If the first row does not consist of field names, the file is processed as if it contains only data.

No DIF header information is used when sending the file to the iSeries™, eServer™ i5, or System i5™. To correctly transfer a DIF file to the system, ensure that the file is in the correct format (row and column). It is essential that the field names, if present, make up the first row of data. The subsequent records make up the remaining rows of data. Therefore, when you transfer the data to the iSeries™, eServer™ i5, or System i5™, the file must be saved in the same format as originally created by the data transfer function.

When you transfer data from DIF files to iSeries™, eServer™ i5, or System i5™ files, the data changes as follows:

- ASCII character, date, time, and time-stamp data is changed to EBCDIC character data or to hexadecimal data. ASCII-to-EBCDIC conversion is done byte by byte. ASCII-to-hexadecimal conversion is done by changing two ASCII bytes to one hexadecimal byte.
- ASCII numeric data changes to system binary, zoned decimal, or packed decimal data, depending on the data type the system specifies.

  The lengths of the fields on the system and the workstation can be different, because of the explicit way minus signs and decimal points are stored in ASCII numeric fields. This means that each field changes individually, to ensure that the resulting field length matches the system specifications for that field. The data transfer function tries to fit the workstation data into the system field.
- In null fields, a NULL DIF character field results in a null value being sent to the iSeries™, eServer™ i5, or System i5™ field if the field is null-capable.
- If the iSeries™, eServer™ i5, or System i5™ field is variable-length, the field is converted to the iSeries™, eServer™ i5, or System i5™ variable-length format.

Errors When Transferring Data from DIF Files

When you transfer data from a workstation DIF file to a system file with data definitions, the following errors can occur:

- Data in this workstation file is not valid, or the version of this workstation file is not supported. The DIF file does not follow the standard DIF format. Processing ends, and no more records are transferred.
- Data in this field is too long for the iSeries™, eServer™ i5, or System i5™ field. The data is truncated. The workstation file-description file defines character or numeric data as longer than the field length specified for the system file.

  For character data, this error occurs only if the extra bytes are not spaces. For hexadecimal data, this error occurs only if the extra bytes are not zeros. The extra bytes are truncated so that the data fits into the specified iSeries™, eServer™ i5, or System i5™ field.
- The value of numeric data is too large for the system field. The maximum value is used. This error occurs when:
  - Numeric data in the workstation field does not fit into the specified number of bytes for the iSeries™, eServer™ i5, or System i5™ field.
  - The decimal value of a numeric field contains more digits than are specified for the system field.

  The value of the field is set to the maximum value possible for the number of bytes and digits the system specifies.
- Data in this field has too many decimal positions. The number is rounded. The number of decimal positions in the workstation field is greater than the number of decimal positions specified on the system. The data is rounded up if the first extraneous digit is 5 or greater, and is rounded down if it is less than 5.
- Data in this field is incorrect or does not match the workstation data type. One of the following things has occurred:
  - A numeric field contains nonnumeric data.
  - A character or hexadecimal field contains a numeric field or a numeric (zoned, packed, or binary) field contains a character field.
  - An iSeries™, eServer™ i5, or System i5™ hexadecimal or numeric (zoned, packed, or binary) field contains a DIF error cell.

  When this error occurs, the transfer request ends to prevent the transfer of incorrect data to the system file.
- Data for this field is missing. This occurs when a data field is defined, but the data is not in the file. This means that the end of the record is reached before all of the defined data is found. If the host field is null-capable then a null is inserted; otherwise, the default values are used.

  When this error occurs, the field or fields for which data is defined, but not found, are filled with default values and are transferred to the system file. These default values are EBCDIC spaces for character fields, or zeros for numeric fields.

  To supply your own default values, use the Default (DFT) keyword in the DDS for the file.

- Data in this field exceeds the field size. The data is lost. This error occurs when extra data, not defined by the file-description file, is found at the end of a character field. The extra bytes are truncated and are not transferred to the system file.
- Extra data is found at the end of this record. The extra data is not transferred. This error occurs when there is extra data at the end of the record, and the iSeries™, eServer™ i5, or System i5™ data definitions or file-description file have not defined it. This extra data is not transferred to the system, because no definitions exist to define the data and how it should change.

BIFF Files

The BIFF file format is used by Microsoft® Excel. In a BIFF file, data is expressed in lines and columns. A BIFF file contains character and numeric data (both positive and negative decimal values).

BIFF format versions 4 and 8 are supported for 5250 Data Transfer. Both BIFF4 and BIFF8 support 256 columns, which is the maximum for a Microsoft Excel worksheet. Documentation on both formats is freely available from the Microsoft Web site.

BIFF4 handles data for Microsoft Excel V2, V3, and V4. The format supports a maximum of 16 384 rows.

BIFF8 is a superset of BIFF4 and stores data as an OLE compound document. BIFF8 handles data for Microsoft Excel V5, V7 (Excel 95), V8 (Excel 97), and V9 (Excel 2000). The format supports a maximum of 65 536 rows.

The transfer facility supports only the following two data types for a BIFF file:

- Character data
- Numeric data

Transferring Data to BIFF Files

When a BIFF file is created, the system data is converted to equivalent Excel cell data.

If untranslatable data is found, the entire field is treated as an error cell.

Transferring Data from BIFF Files

If an error cell is found during data transfer from a BIFF file to the iSeries™, eServer™ i5, or System i5™, either of the following things can occur depending on the data type of the file:

- If the system field is a character (EBCDIC) field, the error cell containing untranslatable characters (hexadecimal zeros) is transferred to the system. A message indicating how many bytes of untranslatable data were transferred is displayed.
- If the system field is a hexadecimal, zoned decimal, packed decimal, or binary field, an error message indicating that the data in this cell is not valid and thus has not been transferred to the system is displayed.

When you transfer data from a system file to a BIFF file, the first record contains field names, which can be treated as column headers.

To return a BIFF file to the system, the first line must contain these field names (as defined in the system) or data. If the first line does not contain field names, the file is regarded as containing data only.

When a file is sent to the iSeries™, eServer™ i5, or System i5™, cell information (such as the character size and font information) is ignored. This means that cell information is lost, even if the contents of a BIFF file that have been sent to the iSeries™, eServer™ i5, or System i5™ are retransmitted to a workstation.

When you transfer data from a BIFF file to an iSeries™, eServer™ i5, or System i5™ file, the data is converted as follows:

- ASCII character cell data is converted to EBCDIC character data or hexadecimal data; 1-byte ASCII data is converted to 1-byte EBCDIC data.
- ASCII numeric cell data is converted to a binary number, or a zoned or packed decimal number, depending on the data type specified in the system.

When you transfer data from a BIFF file to the iSeries™, eServer™ i5, or System i5™, the following specific processing is performed:

- When you transfer data to a BIFF file, the first record contains the names of the fields to be transferred, which can be treated as column headers. To return a BIFF file to the iSeries™, eServer™ i5, or System i5™, the first line must contain the same field names (as defined in the iSeries™, eServer™ i5, or System i5™) or data. If the first line or the first set does not contain a character field that exactly matches the iSeries™, eServer™ i5, or System i5™ field, the file is treated as being a file with no column headers, and only data is processed.
- When you transfer a BIFF file to the iSeries™, eServer™ i5, or System i5™, header information is not used.
- To ensure correct transfer of a BIFF file to the iSeries™, eServer™ i5, or System i5™, the file format must be valid (lines and columns). Data for each set or line must correspond to one record in the iSeries™, eServer™ i5, or System i5™ file.

Errors When Transferring Data from BIFF Files

When you transfer data from a BIFF file on a workstation to the system file with the data definition, the following errors can occur:

- Data in this workstation file is not valid, or the version of this workstation file is not supported. The BIFF file does not conform to the standard BIFF format. Processing terminates, and no more records are transferred.
- Data in this field is too long for the corresponding iSeries™, eServer™ i5, or System i5™ field. The data is truncated. A file-description file defines character or numeric data that is longer than the field specified in the system file.

    - For conversion from ASCII to EBCDIC, this error occurs if a file-description file defines ASCII data that is longer than the field specified on the iSeries™, eServer™ i5, or System i5™.

      During conversion from ASCII to hexadecimal, this error will occur if a file-description file defines ASCII data that is twice as long as the field specified on the iSeries™, eServer™ i5, or System i5™. This is because 2-byte ASCII data is converted to one hexadecimal character.
    - A truncation error only occurs if excess bytes are other than blanks (X'20') during conversion from ASCII to EBCDIC, or other than zeros (X'30') during conversion from ASCII to hexadecimal. Truncating these excess bytes enables data to fit into the specified iSeries™, eServer™ i5, or System i5™ fields.
- Numeric data is too long to fit into the corresponding iSeries™, eServer™ i5, or System i5™ field. The maximum value is assumed. This error occurs under either of the following conditions:
    - Numeric data in a workstation field is too long to fit into the number of bytes specified for the iSeries™, eServer™ i5, or System i5™ field.
    - The number of decimal digits in a numeric field exceeds the number of digits specified for the iSeries™, eServer™ i5, or System i5™ field.

  The field value is set to the maximum value that can be specified for the number of bytes, and that for the number of digits, specified for the iSeries™, eServer™ i5, or System i5™.
- Data in this field contains too many decimal places. The data is rounded off. The number of decimal places in a workstation field is greater than the number of decimal places specified for the system. If the first excess digit is 5 or more, the data is rounded up. Otherwise, it is rounded down.
- Data in this field is not correct, or its type does not match the type of workstation data. One of the following things has occurred:
    - A numeric field contains other than numeric data.
    - A character field or a hexadecimal field contains a number, or a numeric (zoned or packed decimal, or binary) field contains characters.
    - A hexadecimal field or a numeric (zoned or packed decimal, or binary) field for the iSeries™, eServer™ i5, or System i5™ contains a BIFF error cell.

  If this error occurs, the transfer request terminates to avoid transferring incorrect data to the system file.
- Data for this field is missing. This error occurs if the data field is defined, but the file does not contain any data. This means that the end of the record is reached before all defined data has been found.

  If this error occurs (that is, if data is defined for one or more fields, but it is not found there) the fields containing the default value are transferred to the system file. The default value is EBCDIC spaces for a character field and zeros for a numeric field.

  To specify a user-specific default value, use the default value (DFT) keyword in DDS for the file.
- Data in this field exceeds the size of a workstation field. Data is lost. This error occurs if excess data, not defined in the workstation file-description file, is found at the end of the field. For character data, excess bytes are truncated, and not transferred to the system file. For numeric data, the entire field is converted to zeros and transferred to the system file.

- Excess data is found at the end of this record. The excess data is not transferred. This error occurs if such excess data is not defined in the iSeries™, eServer™ i5, or System i5™ data definition or in the workstation file-description file. This excess data is not transferred to the system, because the data and the conversion method are not defined.

DOS Random Files

DOS random files are fixed-length files used by the DOS random read and write routines. The characteristics of DOS random files are as follows:

- There are no end-of-record or end-of-file markers.
- Records are delimited by their constant length, relative positions in the file, and the total length of the file.

Note: DOS random and DOS random type-2 files are identical, except for the way in which the signs are represented for packed decimal and zoned decimal numbers.

Transferring Data to DOS Random Files

When creating DOS random file data definitions, system data changes as follows:

- Binary fields on the iSeries™, eServer™ i5, or System i5™ and the workstation are represented as two-complement numbers, so it is unnecessary to change individual bytes. The workstation uses the convention of storing numeric values with the least significant byte in the left-hand byte position. The data transfer function then reverses the order of the bytes in the binary fields.

  For example, X'CEF3', coming from the system as a 2-byte binary number (representing the value -12557), appears as X'F3CE'.
- EBCDIC character, date, time, and time-stamp data changes byte by byte and is mapped into ASCII characters as defined by the translation tables.
- Variable-length and null fields are converted to fixed length, and trailing blanks (for character, hex, date, time, and time stamp) or zeros (for binary, zoned, and packed) are added to the maximum length of the field.
- Hexadecimal fields do not change.
- Packed decimal fields do not change except for the last half-byte, which contains the sign. The workstation uses X'3' to indicate a positive number and X'B' to indicate a negative number in the sign half-byte.

  For example, X'0865431F' appears as X'08654313'.
- Zoned decimal fields from the system change from EBCDIC to ASCII, as do character fields, except that the sign half-byte in the workstation changed field is X'3' to indicate a positive number and X'B' to indicate a negative number.

  For example, EBCDIC X'F0F1F2F5F2D6' appears as ASCII X'3031323532B6'.

Transferring Data from DOS Random Files

When you transfer data from DOS random files to iSeries™, eServer™ i5, or System i5™ files, the data changes as follows:

- ASCII character, date, time, and time-stamp data changes to EBCDIC character data on a byte by byte basis.
- Binary fields in the workstation file are stored in an order reversed from what the system file expects. These bytes reverse and transfer to the system file.
- Hexadecimal fields do not change. The field length on the system should be the same as the field length on the workstation.
- For packed decimal fields, only the last half-byte (the byte that contains the sign) is changed. The host system uses X'F' to indicate a positive number and X'D' to indicate a negative number for the sign half-byte.

  For example, X'08654313' appears as X'0865431F'.
- Zoned decimal fields on the workstation change from ASCII to EBCDIC , as do character fields. The last half-byte (the half-byte that contains the sign) in the workstation field is changed to X'F' to indicate a positive number and X'D' to indicate a negative number.

  For example, ASCII X'3031323532B6' appears as EBCDIC X'F0F1F2F5F2D6'.
- For null-capable iSeries™, eServer™ i5, or System i5™ fields, null values (except date, time, and time stamp) cannot be reliably detected and are not uploaded. For variable-length iSeries™, eServer™ i5, or System i5™ fields, trailing blanks are removed, and the field is converted to the variable-length format.

Errors When Transferring Data from DOS Random Files

When you transfer data from a DOS random file to an iSeries™, eServer™ i5, or System i5™ file, the following errors can occur:

- Data in this field is too short for the system field. The data is padded. This error occurs when the workstation file contains character or hexadecimal data shorter than the specified field length. It also occurs if the length of the workstation field is defined as less than the system field, or if the data in the last record of the file is too short. Character fields are padded on the right with EBCDIC spaces. Hexadecimal fields are padded on the right with zeros.
- Data in this field is too long for the system field. The data is truncated. This error occurs when the workstation file-description file defines character or hexadecimal data as longer than the field length specified for the system file.

  For character data, this error occurs only if the extra bytes are not spaces. For hexadecimal data, this error occurs only if the extra bytes are not zeros. These extra bytes are truncated so that the data fits into the specified field.
- The value of numeric data is too large for the system field. The maximum value is used. This error occurs when:

    - Numeric data in the workstation field does not fit into the specified number of bytes for the iSeries™, eServer™ i5, or System i5™ field.
    - The decimal value of a numeric field contains more digits than were specified for the iSeries™, eServer™ i5, or System i5™ field.

  The value of the field is set to the maximum value possible for the number of bytes and digits specified by the system.

- Data in this field has too many decimal positions. The number is rounded. This occurs when the number of decimal positions in the workstation field is greater than the number of decimal positions specified on the system. The extra bytes are significant, because the data rounds up if the first extraneous digit is 5 or greater, and rounds down if it is less than 5.
- Data in this field is incorrect or does not match the workstation data type. This error occurs when nonnumeric data appears in a field defined as numeric by the file descriptions. When this occurs, the transfer request ends to prevent transferring incorrect data to the system file.
- Data for this field is missing. The default values are used. This error occurs when a data field is defined, but the data is not in the file. This means that the end of the file is reached before all the defined data is found.

  When this error occurs, the field or fields for which data has been defined, but not found, fill with default values and transfer to the system file. Default values are EBCDIC spaces for character fields, or zeros for numeric fields.

  To supply your own default values, use the default (DFT) keyword in the DDS for the file.

When you transfer data from a DOS random file to a system file without data definitions, any data shorter than the record length defined for the system file is padded with EBCDIC spaces.

Because DOS random files have no record delimiters, this error occurs only on the last record and probably indicates that the record length of the system file does not match that of the workstation file.

DOS Random Type-2 Files

DOS random type-2 files are fixed-length files used by the DOS random read and write routines. The characteristics of DOS random type-2 files are as follows:

- There are no end-of-record or end-of-file markers.
- Records are delimited by their constant length, relative positions in the file, and the total length of the file.

Note: This workstation file type is identical to the DOS random file type, except that the internal sign representation for packed decimal and zoned decimal data types follow Systems Application Architecture® (SAA®) standards. Some workstation applications, such as applications written in IBM® COBOL/2™

programming language, need to have the signs for packed decimal and zoned decimal data types represented this way. Use the DOS random type-2 file type for those workstation applications.

Transferring Data to DOS Random Type-2 Files

When you create DOS random type-2 file data definitions, system data changes as follows:

- Binary fields on the iSeries™, eServer™ i5, or System i5™ and the workstation are represented as two complement numbers, so it is unnecessary to change individual bytes. The workstation uses the convention of storing numeric values with the least significant byte in the left-hand byte position. The data transfer function then reverses the order of the bytes in binary fields.

  For example, X'CEF3', coming from the system as a 2-byte binary number (representing the value -12557), appears as X'F3CE'.
- EBCDIC character, date, time, and time-stamp fields change byte by byte and are mapped into ASCII characters as defined by the translation tables.
- Variable-length and null fields are converted to fixed length, and trailing blanks (for character, hex, date, time, and time stamp) or zeros (for binary, zoned, and packed) are added to the maximum length of the field.
- Hexadecimal fields do not change.
- Packed decimal fields do not change. The sign convention used on the workstation and on the host system is the same.

  For example, X'0865431C' appears as X'0865431C'.
- Zoned decimal fields from the system change from EBCDIC to ASCII, as do character fields. However, the sign half-byte is changed to a 3 to indicate a positive number or a 7 to indicate a negative number when the data is sent to the workstation.

  For example, EBCDIC X'F0F1F2F5F2D6' appears as ASCII X'303132353276'.

Transferring Data from DOS Random Type-2 Files

When you transfer data from DOS random type-2 files to iSeries™, eServer™ i5, or System i5™ files, the data changes as follows:

- ASCII character data, date, time, and time stamp data change to EBCDIC character data on a byte by byte basis.
- Binary fields in the workstation file are stored in an order reversed from what the system file expects. These bytes reverse and transfer to the system file.
- Hexadecimal fields do not change. The field length on the system should be the same as the field length on the workstation.
- For packed decimal fields, the last half-byte (the half-byte that contains the sign) is not changed unless the sign half-byte is less than X'A' (represented by values 0 through 9). If the sign half-byte is less than X'A', it is changed to X'F' on the host system.

  For example, X'865431D' appears as X'0865431D', but X'08654318' appears as X'0865431F'.

- Zoned decimal fields on the workstation change from ASCII to EBCDIC, as do character fields. However, the sign half-byte is changed to an F to indicate a positive number or a D to indicate a negative number when the data is sent to the host system.

  For example, ASCII X'303132353276' appears as EBCDIC X'F0F1F2F5F2D6'.
- For null-capable iSeries™, eServer™ i5, or System i5™ fields, null values (except date, time, and time stamp) cannot be reliably detected and are not uploaded. For variable-length iSeries™, eServer™ i5, or System i5™ fields, trailing blanks are removed and the field is converted to the variable-length format.

Errors When Transferring Data from DOS Random Type-2 Files

When you transfer data from a DOS random type-2 file to an iSeries™, eServer™ i5, or System i5™ file, the following errors can occur:

- Data in this field is too short for the system field. The data is padded. This error occurs when the workstation file contains character or hexadecimal data shorter than the specified field length. It also occurs if the length of the workstation field is defined as less than the system field, or if the data in the last record of the file is too short. Character fields are padded on the right with EBCDIC spaces. Hexadecimal fields are padded on the right with zeros.
- Data in this field is too long for the system field. The data is truncated. This error occurs when the workstation file-description file defines character or hexadecimal data as longer than the field length specified for the system file.

  For character data, this error occurs only if the extra bytes are not spaces. For hexadecimal data, this error occurs only if the extra bytes are not zeros. These extra bytes are truncated so that the data fits into the specified field.
- The value of numeric data is too large for the system field. The maximum value is used. This error occurs when:
  - Numeric data in the workstation field does not fit into the specified number of bytes for the iSeries™, eServer™ i5, or System i5™ field.
  - The decimal value of a numeric field contains more digits than were specified for the iSeries™, eServer™ i5, or System i5™ field.

  The value of the field is set to the maximum value possible for the number of bytes and digits specified by the system.
- Data in this field has too many decimal positions. The number is rounded. This occurs when the number of decimal positions in the workstation field is greater than the number of decimal positions specified on the system. The extra bytes are significant, since the data rounds up if the first extraneous digit is 5 or greater, and rounds down if it is less than 5.
- Data in this field is incorrect or does not match the workstation data type. This error occurs when nonnumeric data appears in a field defined as numeric by the file descriptions. When this occurs, the transfer request ends to prevent transferring incorrect data to the system file.
- Data for this field is missing. The default values are used. This error occurs when a data field is defined, but the data is not in the file. This means that the end of the file is reached before all the defined data is found.

> When this error occurs, the field or fields for which data has been defined, but not found, fill with default values and transfer to the system file. Default values are EBCDIC spaces for character fields, or zeros for numeric fields.
>
> To supply your own default values, use the default (DFT) keyword in the DDS for the file.

When you transfer data from a DOS random type-2 file to a system file without data definitions, any data shorter than the record length defined for the system file is padded with EBCDIC spaces.

Because DOS random type-2 files have no record delimiters, this error occurs only on the last record and probably indicates that the record length of the system file does not match that of the workstation file.

No-Conversion Files

No-conversion files, defined by the data transfer function, consist of data that has not changed. For example, when data transfers from the system to a workstation no-conversion file, the data transfers exactly as it is stored on the iSeries™, eServer™ i5, or System i5™. Date, time, and time-stamp data transfers to EBCDIC character data on the workstation.

Transferring Data to No-Conversion Files

When you transfer data from the iSeries™, eServer™ i5, or System i5™ to a no-conversion file, the data transfers exactly as it is stored on the system.

Variable-length iSeries™, eServer™ i5, or System i5™ fields are converted to fixed-length fields, and trailing EBCDIC blanks are added to the maximum length of the field.

Date, time, and time-stamp data is converted to EBCDIC character data.

Variable-length and null fields are converted to fixed length, and trailing EBCDIC blanks (for character, hex, date, time, and time stamp) or EBCDIC zeros (for binary, zoned, and packed) are added to the maximum length of the field.

Transferring Data from No-Conversion Files

The data types that exist in a no-conversion file are EBCDIC system data types only. When a no-conversion file transfers to the system, the data transfer function performs no data change or translation. Date, time, and time-stamp data transfers to EBCDIC character data on the workstation.

However, the data transfer function verifies that all numeric data is in the correct EBCDIC format. If any numeric data is found that is not in the correct EBCDIC format, that data and any remaining data does not transfer.

Errors When Transferring Data from No-Conversion Files

When you transfer data from a workstation no-conversion file to a system file, the following errors can occur:

- Data sizes are not equal. When you transfer no-conversion files, the length and decimal position specifications for the system and the workstation must match exactly. If not, no records transfer.
- Data in this field is too short for system field. The data is padded. This error occurs when the workstation file contains character or hexadecimal data shorter than the field length specified for the system file. This could occur if the data in the last record of the file is too short. Character fields are padded on the right with EBCDIC spaces. Hexadecimal fields are padded with zeros.
- Data in this field is incorrect or does not match the workstation data type. The transfer request ends to prevent transferring incorrect data to the system file. This error occurs when a field defined by the file descriptions as numeric contains nonnumeric data.

  Note: The data is verified assuming that the data is in EBCDIC format. If you want to transfer data in another format, do not use data definitions or file descriptions, and specify the record lengths defined on the system and the workstation in the same way.

- Data for this field is missing. The default values are used. This error occurs when a data field has been defined, but the data is not in the file. This error can occur only in the last record of the file, since no-conversion files have no explicit record delimiters.

  When this error occurs, the field or fields for which data has been defined but not found fill with default values and transfer to the system file. These default values are EBCDIC spaces for character fields, or zeros for numeric fields.

  To supply your own default values, use the default (DFT) keyword in the DDS for the file.

iSeries, eServer i5, or System i5 System-to-PC Performance Considerations

Transferring data from the iSeries™, eServer™ i5, or System i5™ to the workstation depends on the following performance considerations:

- The system workload.
- How many records have to be looked at to complete the transfer.
- If more than two files are joined. You need extra iSeries™, eServer™ i5, or System i5™ resources to join records from more than one file.
- If **GROUP BY** fields are specified.
- If complicated **WHERE** or **HAVING** comparisons are specified.

These factors and others influence the time needed to determine which data should be transferred. For example, the time needed to receive the first record of a transfer in which all the records are chosen is less than the time needed to start transferring a smaller group of records based on complicated **WHERE** or **HAVING** values. However, transferring all the records in a large file is sometimes impractical or unnecessary.

The iSeries-to-workstation data transfer function uses many functions within the iSeries™, eServer™ i5, or System i5™ to determine the fastest method of selectively retrieving records. When it selects a smaller group of records to transfer, the iSeries-to-workstation data transfer function uses the existing access paths whenever possible to improve performance.

For the iSeries-to-workstation data transfer function to consider using an existing access path (logical file), the access path must meet the following conditions:

- It must be defined to the data that transfers.
- It must have either *DELAY or *IMMED maintenance.

When you meet these conditions, you must then match the transfer request to the access path. The following considerations might be helpful when you define your transfer request:

- The time it takes to select records based on **WHERE** clause values is less when the following things are true of the **WHERE** field:
    - It is compared with a constant.
    - It is the first key field in an existing access path defined to the data to be transferred.
- A transfer request containing a **GROUP BY** or **ORDER BY** clause or both can work better if the key fields in the access path are in the same order as specified on the **GROUP BY** or **ORDER BY** clauses.
- A transfer request containing a **JOIN BY** clause can work better when:
    - An access path exists over the file that you are joining to.
    - The field you are joining to is a primary key field in the access path.
    - You are not returning records with missing fields.

Chapter 10. Transferring Files

Z and I Emulator for Windows File Transfer enables you to transfer one or more files between a host system and workstation at the same time. Transfer types and translation tables can be defined in advance.

You can perform the following file transfer functions:

- Send files to the host system
- Receive files from the host system
- Use lists of files
- Create templates to define file names and transfer types
- Define transfer types
- Set transfer options
- Modify translation tables
- Transfer files via the XMODEM or YMODEM protocols

Note:

PCT400 was withdrawn from marketing 3/98.

Host Requirements

For PC400 File Transfer in SBCS mode, you need one of the following host file-transfer programs (referred to as APVAFILE):

- Z and I Emulator for Windows Tools/400 8mm Tape — 46H8350
- Z and I Emulator for Windows Tools/400 1/2 inch Tape — 85G9973
- Z and I Emulator for Windows Tools/400 1/4 inch Tape — 85G9969

Sending Files to the Host System

To send a file from your workstation to the host system:

1. Sign on to the host system.
2. Click **Send File to Host** from the **Actions** menu of the session window. (You can also select the **Send** button on the tool bar.)

   The Send File to Host window opens.
3. Specify the name of the workstation file to be sent to the host system by entering the name in the **PC File** text box, or click the **Browse** button to open a dialog box for selecting the file.

4. Enter the name under which the file will be stored on the host; then enter or select the **Transfer Type**. If a template is provided for the file type you are transferring, the host file name and the transfer type appear automatically.

   Using List Files: Select **Open List**; then select the list to be used for transfer. See Creating List Files on page 185 for details of how to create list files.

5. Click **Send**.

   The file is sent to the host system. The send status appears in the Send a File Status window.

Receiving Files from the Host System

To transfer a file from the host system to your workstation:

1. Sign on to the host system.
2. Click **Receive File from Host** from the **Actions** menu. (You can also select the **Receive** button from the tool bar.)

   The Receive File from Host window opens.
3. Specify the name of the host file to be received. Enter the name in the **Host File** text box, or specify it as follows:

   Using the Clipboard button: If you have copied one or more host file names to the clipboard, you can paste the names into the transfer list; click the **Clipboard** button to open a dialog box for this. Select one or more of the pasted file names to be transferred. Then click **OK**.

4. Enter or modify the suggested name under which the file will be stored on the workstation, and enter or select the **Transfer Type**; or click the **Browse** button to open a dialog box for selecting a location for the file.

   Using List Files: Select **Open List**, and select the list to be used for transfer. (See Creating List Files on page 185 for an explanation of how to create list files.)

   If a template is provided for the file type you are transferring, the workstation file name and the generated transfer type appear automatically.

5. Click **Receive**.

   The receive status appears in the **Receive a File Status** window.

Using List Files

If the same files are transmitted frequently, you can create a list of the files and save it.

A list file can be used for both Send and Receive. The default list file extension is .SRL.

Creating List Files

To create a list file:

1. Click **Receive File from Host** from the **Actions** menu or **Send File to Host** from the **Actions** menu of the session window; or click the **Send** or **Receive** buttons on the tool bar.

   The corresponding window opens.
2. Select a file to be transferred from the **Host-File Name** or **PC-File Name** list box by pointing to the name of a file to be selected. While holding down the Ctrl key, click the left mouse button.

   The file name, its corresponding workstation or host file name (according to the available templates), and the transfer type appear in the **Transfer List** part of the window.

   Note: You can also click the **Browse** button (for sending files) or the **Clipboard** button (for receiving files) to open the corresponding dialog box, which allows you to select files for transferring; when you click **OK**, the selected files are shown in the **Transfer List**.
3. Click the **Add to List** button to include a selected file in the **Transfer List**.
4. After all desired files have been selected, click **Save List**.

   The Save File-Transfer List File As window opens.
5. Enter or select a list name, and click **OK**.

Editing Lists

To edit the contents of a previously created list:

1. As explained in Sending Files to the Host System on page 183 and Receiving Files from the Host System on page 184, display the Send File to Host or Receive File from Host window.
2. Select **Open List**.

   The Open File-Transfer List File window opens.
3. Select the name corresponding to the list file to be edited, then click **OK**.
4. The contents of the selected list appear in the Send File to Host or Receive File from Host window.
5. Edit the contents of the list file.

   Changing the contents of a list: Choose the file to be changed from the list, and overwrite the items to be changed in the text box; then click the **Update in List** button.

   Removing a file from the list: Choose the file to be removed, and click **Remove from List**.

   Adding a file to the list: Double-click the file to be added from the list of host or workstation files.
6. Select **Save List**.

The Save File-Transfer List File As window opens.

7. Enter a name and then click **OK**.

Managing Templates

A *template* is a set of rules to be used by the workstation to automatically generate a workstation or host file name and transfer type when you specify a file to be sent or received.

You can have up to 32 templates. They are automatically numbered from 1 to 32.

When you specify a file to be transferred, the workstation scans the templates, starting from template 1. It uses the first matching template to generate a name for the transferred file and the transfer type.

To manage a template:

1. Click **Receive File from Host** from the **Actions** menu or **Send File to Host** from the **Actions** menu of the session window; or click the **Send** or **Receive** buttons on the tool bar.

   The Send File to Host or Receive File from Host window opens.
2. Select **Template**.

   The Template window opens. The contents of the window depend on the connected host system.

Adding Templates

The list box for the Template window lists the currently stored templates.

To add a template:

1. Select any template from the list box.

   The contents of the selected template appear under the list box.
2. Change the workstation or host file names or extensions by overwriting them; then select the transfer type. (For details of the transfer types, see Defining Transfer Types on page 187.)
3. Click **Add**.

   The window for determining where in the list to display the new template opens.
4. Select a template number and specify whether to display the new template before or after the template that has that number. Click **OK**.

   The new template is added to the list in the appropriate position.

Replacing and Deleting Templates

To change the contents of a currently stored template, or to delete a template:

1. Select the template to be changed or deleted.

   The contents of the selected template appear under the list box.
2. To change the contents, overwrite the appropriate part and then click **Replace**.

   To delete a template, click **Delete**.

   The selected template is changed or deleted, and the contents of the template list box are changed.

Testing Templates

To test the contents of an added or changed template:

1. Select the template to be tested from the list box.

   The number of the selected template appears in the Test Templates box in the lower part of the window.
2. Select or enter data for the following items:

   Test Mode

   Determine which mode is to be used for the test: the mode in which a file is transmitted from the workstation to the host system (send), or the mode in which a file is transmitted from the host system to the workstation (receive).

   Templates

   Determine which templates to test: only the template selected in step 1, or all registered templates.

   Source File

   Enter the name of the file to be used for the test.
3. Click **Test**.

   Target File indicates the name that has been generated by the template.

Note: Testing a template does not transfer a file.

Defining Transfer Types

Transfer types define the option information used for controlling file transfer. Up to 32 transfer types can be defined for each host system. Text, binary, and append (excluding CICS®) are the defaults.

To add or change transfer types:

1. Click **Preferences → Transfer** from the **Edit** menu of the session window.
2. Click the tab for your host type or modem protocol.

   The property page for the selected host or modem protocol opens. The items that appear depend on the selected host system.

3. Enter transfer-type names in the **Transfer Type** box, or select them from the drop-down list.
4. Select or enter the required items (see Items to Be Specified on page 188).

   To add or replace a transfer type, click **Save**. To delete a transfer type, click **Delete**.
5. A dialog box displays, asking for confirmation. Click **OK**.

Items to Be Specified

Choosing the appropriate property page enables you to set the items described in the following sections.

File Options

The file options that can be used depend on the type of the connected host system and the host code page selected when the session was configured. Table 29: Mode Values for File Transfer Options on page 188 lists the mode values for the file transfer options. File Transfer for PC400 on page 191 lists transfer options.

Table 29. Mode Values for File Transfer Options

Mode	Host Code Page
SBCS	Others

Logical Record Length (LRECL)

Enter the **logical record length** to be used (host record byte count) in the **LRECL** text box. If **Variable** and **Undefined Mode** are specified as the record format, the logical record length is the maximum record length within a file. The maximum value is 32767.

The record length of a file sent from a workstation to the host system might exceed the logical record length specified here. If so, the host file transfer program divides the file by the logical record length.

To send a file containing long records to the host system, specify a sufficiently long logical record length.

Because the record length of a workstation file exceeds the logical record length, a message does not appear normally if each record is divided. To display a message, add the following specification to the `[Transfer]` item of the workstation profile:

```
DisplayTruncateMessage = Y
```

Additional Options

The required host command options can be entered in the **Additional Options** text box.

Setting General Transfer Options

To set advanced options:

1. Select **Preferences → Transfer** from the **Edit** menu of the session window.

   The setup dialog is displayed.
2. Change the required settings on the property page labeled **General**.
3. Click **OK**.

The following sections contain information about the items which can be defined for file transfer options.

Data Transfer

You can choose whether the Data Transfer function (see Data Transfer for PC400 on page 97) is to be used instead of the normal file transfer function.

Host Command

You can specify host command to be called when file transfer starts. If nothing is entered in this text box, APVAFILE is used for 5250 sessions.

Default PC Directory

You can specify the default directory that appears in the Send File to Host or Receive File From Host window. To select the directory, click the **Browse** button.

Default Library

You can specify the iSeries™, eServer™ i5, or System i5™ library to be used as the default.

PC Code Page

When a file is transferred, EBCDIC codes are converted to 1-byte workstation codes, and vice versa. A valid value is automatically selected from among the following values for SBCS sessions: 437, 737, 806, 813, 819, 833, 850, 852, 854, 857, 858, 860, 861, 862, 863, 864, 865, 866, 869, 874, 912, 915, 916, 920, 921, 922, 1008, 1089, 1124, 1125, 1127, 1129, 1131, 1133, 1153, 1155, 1156, 1157, 1158, 1160, 1164, 1250, 1251, 1252, 1253, 1254, 1255, 1256, 1257, and 1258; —according to the host code page specified when the workstation is configured. For an explanation of how to select host code pages, see the online help for the host code page.

File-Transfer Timeout

You can define the time the workstation waits for a response from the host system (in seconds). If the host system does not respond, the transfer is canceled, and an error message appears. A number in the range 20–65535 (or 0) can be specified. The default timeout is 30 seconds. Specify an appropriate value such that the error message does not appear too early. If you specify 0, a timeout is not set.

If a packet or block size is relatively large for low-speed lines, such as or COM port lines, it is recommended that 150 seconds or greater be specified.

Extension for List-Files

You can change the default extension (.SRL) of file-transfer list files.

Show Status Window

You can choose the method of displaying the file-transfer-progress status.

In Session

When file transfer starts, the status window opens. The name of the file being transferred and the transfer progress appear.

In icon

When file transfer starts, the status icon appears on the screen. If the icon is restored, the status window opens.

Setting Up the Translation Table

You can create or edit the translation table to be used for sending or receiving files.

Changing the Translation Table

To change the translation table:

1. Select **Preferences → Transfer** from the **Edit** menu of the session window.
2. Click the **Translation Table** tab on the resulting window.

   The Translation-Table Setup property page opens.
3. The table currently being used (default or the name of a user-defined table) is shown. Choose either **Default** or **User-Defined**.
4. If you choose **User-Defined**, enter a translation-table name in the **File Name** text box, or select a name by clicking **Browse**.
5. Click **OK**.

Customizing the Translation Table

You can create a user-specific translation table for transmission or reception, or you can edit an existing translation table.

To create or edit a translation table:

1. On the **Translation Tables** property page, click **Customize** in the Upload or Download window.

   The Customize Translation window opens.

   If you chose **Default** or if you chose **New** from the File menu, the default values appear in the table.

   Translation source codes

   PC code-points when an upload translation table is edited. Host code-points when a download translation table is edited.

   Translation target codes

   Host code-points when an upload translation table is edited. PC code-points when a download translation table is edited.
2. Double-click the code to be changed in the table, and change the value in the entry field that subsequently appears.
3. Click **Save** or **Save As** from the File menu.
4. If asked, enter a name in the Save Translation File As window and click **OK**.
5. Click **Exit** from the File menu of the Customize Translation window.

File Transfer for PC400

File transfer is designed so that you can use it in the following cases:

- To store a workstation file on the iSeries™, eServer™ i5, or System i5™ for a backup
- To edit a source file of an iSeries™, eServer™ i5, or System i5™ program with a workstation editor, and send the file edited on the workstation to the iSeries™, eServer™ i5, or System i5™.
- To distribute workstation documents and programs to the iSeries™, eServer™ i5, or System i5™ users

PC File Transfer with the CRLF Option

If the CRLF option is specified, the transfer program checks for new-line characters. If the record length is reached before a new-line character is found, the record is divided at this point; one sentence of a workstation file will become two or more records. Particularly, specify a sufficiently long record length when retransmitting a workstation file containing 2-byte characters.

By default, the message `records segmented.` is not displayed. To display the message, do the following:

1. Look for the profile for the session you will use. Normally, this will be in the application data directory under the name *filename*.WS (*filename* is a user-specified file name).
2. Use an editor to insert the following sentence into the [Transfer] section. If there is no [Transfer] section, first enter [Transfer]. Be careful to enter it correctly.

   ```
   [Transfer]
   DisplayTruncateMessage=Y
   ```

The next time the session is started, this specification becomes active.

Transfer to a Physical Source File

An iSeries™, eServer™ i5, or System i5™ physical source file contains 12 bytes of information for each record as internal information: 6 bytes are for a record number, the other 6 bytes are for a date. When you transfer a file from a workstation using file transfer, the date field contains 000000. If the APPEND option is not specified, the record number is incremented by 1, up to a maximum of 9999. Otherwise, it is incremented from the nearest integer, greater than the number of the last record in the original file (for example, 24 for 23.1). If the number of records exceeds 9999, the next and all subsequent record numbers are 9999.

Use the source specifications input utility (SIU) to renumber records when saving the file after editing.

Transfer to a Physical File

A file, such as a PC program, that does not require the processing of the contents of an iSeries™, eServer™ i5, or System i5™ file or the reading of data, should be transferred to a physical file with the BINARY transfer type. Because data is not converted, if the data is subsequently retransmitted from the iSeries™, eServer™ i5, or System i5™ to a workstation, the original workstation file can be re-created exactly. If the data is converted, however, data might not be restored to its original form, depending on the contents of the conversion table.

For the maximum number of members (MAXMBRS), a physical file attribute, the default value is 1. When a physical file is created during file transfer, MAXMBRS is 1.

When a file is transferred from a workstation to a physical file, the default file name **xxxBIN** is assumed (**xxx** is a workstation file extension.) If you transfer more than one file, an error occurs when the second and subsequent files are transferred: `The TRANS58 file or member cannot be created. File transfer terminates`. A file should be created with the expected file attribute before it is transferred from a workstation to the iSeries™, eServer™ i5, or System i5™.

Using the DSPMBRLST Command

For file transfer from the iSeries™, eServer™ i5, or System i5™ to a workstation, the Paste function can be used. If the name of the Library/File(Member) to be transferred is copied with the Copy function of the Edit menu, it can be displayed as the host file candidate to be transferred on the transfer request screen by clicking **Paste**. This is particularly convenient when transferring more than one file at a time.

Use the DSPMBRLST command to list iSeries™, eServer™ i5, or System i5™ files or members. The command format is as follows:

```
DSPMBRLST LIB(lib-name) FILE(file-name)
```

LIB parameter

The LIB parameter contains the target library name. The default value is *USRLIBL. Extensive specification, such as *ALL, ***** for generic name, is possible, but is time-consuming. iSeries™, eServer™ i5, or System i5™ files or members are listed more efficiently if a specific name is specified.

FILE parameter

The FILE parameter contains the target file name. There is no default value. The parameter must be specified. *ALL and ***** for generic name can be specified.

Executing this command lists Library/File(Member) on the screen. If they cannot be listed on one screen, **MORE...** is displayed in the lower right corner of the screen. Use the next page or the preceding page key to scroll the screen. Create a list for Paste with the Copy or the CopyAppend function of the Edit menu, as required.

Restrictions for Transferred File Size

A file that is more than 1 040 000 bytes cannot be transferred correctly.

Part IV. Using Z and I Emulator for Windows VT

Chapter 11. VT Emulation

For connection to ASCII hosts, Z and I Emulator for Windows provides the **VT Emulator** for VT340, VT100, and VT52 terminals. ASCII hosts commonly use these terminal control sequences as standards for session presentation, and many ASCII-host application programs assume a VT-compatible terminal. VT emulation allows your personal computer or workstation to operate as if it were a VT terminal. Software that is designed to operate a VT340, VT100, or VT52 terminal should work correctly with the Z and I Emulator for Windows VT emulator.

Although the keyboard layout on VT terminals is similar to that of the personal computer, there are some exceptions. See Default Key Functions for the VT Emulator Layout on page 36 for the default mapping of keys for VT emulation.

For file transfer to and from ASCII hosts, using the XMODEM and YMODEM protocols, see Using XMODEM and YMODEM on page 209.

VT connections to non-ASCII hosts, such as the IBM® zSeries™ are also possible if you have the appropriate communication devices.

Configuring a VT Session

Use the **Customize Communication → ASCII Host** panel to select values for the parameters that define your ASCII host session. There are two types of parameters: Session and Link.

Customizing the VT over Telnet Attachment

1. Click **Communication** from the WorkStation-window menu bar.
2. Click **Configure** from the **Communication** menu.

   The Customize Communication window opens.
3. Select the ASCII host and then select the **LAN** or **COM Port** interface.

   The available attachments appear.
4. Select **VT over Telnet** attachment.
5. Click **Session Parameters**.

   The Session Parameters — ASCII Host window opens.
6. Set the Session Parameters (see Session Parameters on page 195).
7. Click **Link Parameters**.

   The TelnetASCII window opens.
8. Enter the host name or IP address.
9. Optionally enter the port number, change the terminal ID, or select the **Auto-reconnect** check box.
10. Click **OK** until the Customize Communication window closes.

    Customization is complete.

Session Parameters

These parameters correspond to setup choices on a VT340 terminal.

Online/Local

In the **Online** state, the emulator receives data from the host computer, and can send data to it. In the **Local** state, data you enter on the keyboard appears on the screen, but is not sent to the host; data from the host is held, and not presented on the screen until you change the state to **Online**.

Operating Mode

Select **Char** if the host does not echo the characters you type on your keyboard. The VT emulator displays them as it sends them to the host.

Select **Echo** if the host echos your keyboard characters for display. The VT emulator displays them only as they return from the host.

If you see doubled characters, you should select **Echo** instead of **Char**. **Echo** is the default.

Machine Mode

There are four machine modes. These are:

VT340 mode, with 7-bit controls

This is the default. This mode is recommended for most applications.

VT340 mode, with 8-bit controls

The emulator is set for an 8-bit environment with 8-bit controls.

VT100 mode

This mode is intended for situations requiring strict compatibility with the VT100 terminal.
In general, the VT340 7-bit mode is appropriate for applications that expect a VT100.

VT52 mode

This mode is only for applications designed for the VT52 terminal.

Screen Size

You can choose the number of rows and columns that the session screen displays. The choices are

- Rows: 24, 36, 48, 72, and 144
- Columns: 80 and 132

The defaults are 24 rows and 80 columns.

Type of Host Code-Page

The choices are for the host code page are National, PC, and Multinational. Multinational which selects the 8–bit DEC Supplemental Graphic Character Set is the default. If you select National, then you must select a country from the Host Code Page pull-down list. The PC option selects the PC Code Page 437.

ISO Latin 9 (ISO 8859-15) character set support for ASCII (VT) sessions

Support of the ISO Latin 9 (ISO 8859-15) character set is available for ASCII (VT) sessions.

Optional Parameters

These parameters correspond to setup choices on a VT340 terminal.

Reverse Screen Image

Check this box to reverse the foreground and background colors.

User Feature Lock

Check this box to lock the following functions so that the host cannot change them.

- Auto Repeat
- Keyboard Lock
- Reversed Screen Image
- Tab Stops

Auto Wrap

Check this box if you want the VT emulator to start a new line whenever the current row of characters reaches the end of line.

Auto-Answer Back Message

Check this box if you want the VT emulator to send a message automatically to the host, once a connection has been established.

Move Cursor on Mouse Click

Select this option if you want the cursor to move when you click the left mouse button in the session window presentation space.

Answer Back Message

Enter the message, which is a maximum of 31 characters, to send to the host when communication is established.

Conceal

If you check this box, your answerback message is not displayed in the configuration window. After you conceal your message, the **Conceal** box has no effect, and the message remains concealed until it is changed.

User Defined Key Lock

Check this box to lock user-defined keys. For example, you can select **User Defined Key Lock** and define the values of the F6 to F20 keys. These keys are then locked with those values and cannot be redefined by the host.

Transparent Mode

Check this box to cause the VT emulator to display control characters rather than interpreting them.

VT ID

The attributes of the selected model are sent to the host computer. Choose one of the following: **VT100 ID**, **VT101 ID**, **VT102 ID**, **VT220 ID**, **VT240 ID**, **VT320 ID**, or **VT340 ID**.

History Logging

When this option is enabled, text is logged into the VT history window as it scrolls off the screen from the top margin row. The top and bottom margins are set when the host application defines the scrolling region.

History Logging Buffer Size

Use this list to select one of the available sizes for the history log buffer. The choices are 16KB, 32KB, 64KB, 128KB, and 512KB. The default is 64KB.

History Logging – Enhanced

Data erased due to the Erase in Display command is scrolled into the history window. See Enhanced History Logging on page 208 for more information.

Advanced ASCII Host

The **Advanced** button takes you to the Advanced ASCII Host dialog. The Advanced Options dialog contains all of the configuration options needed for the Local editing feature of VT340 Emulation. The following list defines these configuration options. Default settings are indicated in bold.

Graphics Cursor

Determines whether the graphics input cursor is shown when in graphics mode. Possible values are **Enabled** or Disabled.

Sixel Scrolling

When this option is selected, a sixel graphics image scrolls to the next row when the last column is reached. Possible values are **Enabled** or Disabled.

MacroGraph Reports

Controls the ability of the host to retrieve stored macro graph procedures. Possible values are Enabled or **Disabled**.

Edit Mode

Selects whether local editing is available and the current mode of operation. Possible values are **Unavailable**, Interactive, or Edit.

Erasure Mode

Determines which characters can be erased in edit mode. Possible values are **Unprotected** or All.

Edit Key

Determines how the VT340 emulation switches between interactive and edit mode. Possible values are **Immediate** or Deferred.

Transmit

Determines how the VT340 emulation sends a block of data to the host system in edit mode. Possible values are **Immediate** or Deferred.

Application Keys

Determines how the unshifted function keys F6 through F20 work in edit mode. Possible values are **Disabled**, Immediate, Prefix, or Suffix.

Guarded Area

Determines whether protected characters can be sent to the host system. Possible values are **All** or Selected.

Selected Area

Determines whether the VT340 emulation can send all characters or only selected characters to the host system. Possible values are **All** or Selected.

Multiple Area

Determines whether VT340 emulation can send all selected areas on the page, or only the area selected with the cursor. Possible values are **Multiple** or Single.

VT131 Transfer

When **Line Transmit Mode** is disabled, this feature selects an ANSI-style or VT131-style data transmission. Possible values are **ANSI** or VT131. The size of the block depends on the **Transfer Termination Mode** value.

EOL Characters

Allows you to select characters used to indicate the end of a line (EOL) in a data block. By default the VT340 emulation sends a carriage return (CR). Up to six hexadecimal characters can be specified.

EOB Characters

Allows you to select characters used to indicate the end of a data block (EOB). This feature has no default. Up to six hexadecimal characters can be specified.

Page Coupling

Determines whether to automatically display a new page when the cursor moves to a new page in page memory. Possible values are **Enabled** or Disabled.

Line Transmit Mode

Allows you to send characters one line at a time to the host system. Possible values are **Disabled** or Enabled.

Transfer Termination Mode

When **Line Transmit Mode** is disabled, this feature determines whether the VT340 emulation sends a partial page or the scrolling region. Possible values are **Enabled** or Disabled.

Space Compression Mode

Determines how the VT340 emulation sends unused character fields and spaces in a data block. Possible values are **Disabled** or Enabled.

Link Parameters

The **Configure Links** button take you to a panel for configuring the details of the connection to the ASCII host computer. The panel you see depends upon the attachment type that you chose for your ASCII host. There are two types:

- VT over Telnet

Configuring Links for VT over Telnet

The VT over Telnet attachment is an application that uses TCP/IP (Transmission Control Protocol/Internet Protocol) and that enables remote logon to an ASCII host. TCP/IP provides connectivity functions for both local area networks (LAN) and wide area networks (WAN) and includes the ability to route information between LANs and WANs. The major TCP/IP networks—the Internet—use a standardized addressing procedure to ensure that IP addresses are unique and that communication between enterprises is possible.

The VT over Telnet attachment for Z and I Emulator for Windows requires a TCP/IP stack that supports the Windows® Sockets Version 1.1 interface. WSOCK32.DLL must be in the Windows® system directory or the current path to provide the interface for the stack program and to support the Windows® Sockets V1.1 interface.

For the VT over Telnet attachment, you must define the following attachment parameters.

- Host Name or IP Address (mandatory)
- Port Number (optional)
- Terminal ID (optional)
- Auto-reconnect (optional)

Host Name or IP Address

Specify either the alphabetic name of the target host or its numeric IP address.

Host Name

The name of the target host is a string

Host IP Address

The IP address of the target host is in dotted-decimal notation—for example: 0.0.0.0

Port Number

Specify the decimal number of the target host's Telnet port. The default, 23, is the standard Telnet port.

Terminal ID

The VT emulator and the Telnet server use the terminal ID for negotiating an appropriate connection. Ask your Telnet administrator for your host's correct terminal ID. When the default box is selected, the default values are selected from the Machine Mode, as shown in the following table:

Machine Mode	Default Terminal ID
VT340	DEC-VT220
VT100	DEC-VT100
VT52	DEC-VT52
ANSI	ansi

Auto-reconnect

If the session is disconnected from the host, and if this box is selected, you will be re-connected automatically.

The default is not selected.

Using A VT Session

Your Z and I Emulator for Windows VT session works as if you were using a VT340, VT100, or VT52 terminal. For mainframe VT, iSeries™, eServer™ i5, or System i5™ connections, the protocol converters have defined VT keyboard sequences, such as F1 or PA1.

The following tables are provided:

- Characters generated by VT Compose Key
- Characters displayed in transparent mode
- OIA line display messages

Refer to *Administrator's Guide and Reference* for default mapping of the VT340 keyboard to the PC keyboard, as used by the Z and I Emulator for Windows VT emulator.

Compose Key

The VT emulator supports the VT340 compose key for generating special characters on the display. Before using the compose key, define a key combination that represents it.

Using the compose key involves three separate actions:

1. Press and release the compose key.
2. Press and release the first character (see Table 30: Character Generation (Special Characters) on page 202).
3. Press and release the second character.

The first and second characters may be typed in either order, except when the table specifies that they must be entered as shown,
Table 30: Character Generation (Special Characters) on page 202 shows the appearance and name of each special character, the character pair that generates the character, and an indication whether the order of entering the characters is significant.

Table 30. Character Generation (Special Characters)

Generated Character		Compose Key, Plus This Pair		
Appearance	**Description**	**First**	**Second**	**Order**
Á	A acute	A	'	either
á	a acute	a	'	either
Â	A circumflex	A	^	either
â	a circumflex	a	^	either
À	A grave	A	`	either
à	a grave	a	`	either
Å	A ring	A	*	either
		A	°	either
å	a ring	a	*	either
		a	°	either
Ã	A tilde	A	~	either
ã	a tilde	a	~	either
Ä	A umlaut	A	"	either
ä	a umlaut	a	"	either
Æ	AE ligature	A	E	as shown
æ	ae ligature	a	e	as shown
´	apostrophe	'	space	either
@	at sign	a	a	either
		A	A	either
\	backslash	/	/	either
\	backslash	/	<	either
Ç	C cedilla	C	,	either
ç	c cedilla	c	,	either
¢	cent sign	c	/	either
		C	/	either
		c	|	either

Table 30. Character Generation (Special Characters)

(continued)

Generated Character		Compose Key, Plus This Pair		
Appearance	**Description**	**First**	**Second**	**Order**
		C	|	either
^	circumflex accent	^	space	either
}	close brace	)	-	either
]	close bracket	)	)	either
»	close French quote	>	>	either
@	commercial at	a	a	either
		A	A	either
©	copyright mark	c	o	either
		C	O	either
		c	0	either
		C	0	either
°	degree sign	0	^	either
		°	space	either
		#	space	either
É	E acute	E	'	either
é	e acute	e	'	either
Ê	E circumflex	E	^	either
ê	e circumflex	e	^	either
È	E grave	E	`	either
è	e grave	e	`	either
Ë	E umlaut	E	"	either
ë	e umlaut	e	"	either
a	feminine ordinal indicator	a	_	either
		A	_	either
½	fraction one-half	1	2	as shown
¼	fraction one-quarter	1	4	as shown
ß	German ess-tset	s	s	either
µ µ	Greek mu	/	u	as shown
		/	U	as shown
»	guillemets, closing	>	>	either
«	guillemets, opening	<	<	either
Í	I acute	I	'	either
í	i acute	i	'	either
Î	I circumflex	I	^	either

Table 30. Character Generation (Special Characters)

(continued)

Generated Character		Compose Key, Plus This Pair		
Appearance	**Description**	**First**	**Second**	**Order**
î	i circumflex	i	^	either
Ì	I grave	I	`	either
ì	i grave	i	`	either
Ï	I umlaut	I	"	either
ï	i umlaut	i	"	either
¡	inverted exclamation	!	!	either
¿ ¿	inverted question mark	?	?	either
º	masculine ordinal indicator	o	_	either
		O	_	either
µ µ	micro sign	/	u	as shown
		/	U	as shown
• ▪	middle dot	.	^	either
Ñ	N tilde	N	~	either
ñ	n tilde	n	~	either
#	number sign	+	+	either
Ó	O acute	O	'	either
ó	o acute	o	'	either
Ô	O circumflex	O	^	either
ô	o circumflex	o	^	either
Ò	O grave	O	`	either
ò	o grave	o	`	either
Ø	O slash	O	/	either
ø	o slash	o	/	either
Õ	O tilde	O	~	either
õ	o tilde	o	~	either
Ö	O umlaut	O	"	either
ö	o umlaut	o	"	either
Œ Œ	OE ligature	O	E	as shown

Table 30. Character Generation (Special Characters)

(continued)

Generated Character		Compose Key, Plus This Pair		
Appearance	**Description**	**First**	**Second**	**Order**
œ œ	oe ligature	o	e	as shown
{	open brace	(	-	either
[	open bracket	(	(	either
«	open French quote	<	<	either
¶	paragraph sign	p	!	either
±	plus-or-minus sign	+	-	either
£	pound sterling sign	l	-	either
		L	-	either
		l	=	either
		L	=	either
"	quotation mark	"	space	either
§	section sign	s	o	either
		S	O	either
		s	!	either
		S	!	either
		s	0	either
		S	0	either
'	single quote	'	space	either
ß	ss German	s	s	either
¹	superscript 1	1	^	either
²	superscript 2	2	^	either
³	superscript 3	3	^	either
~	tilde	~	space	either
Ú	U acute	U	'	either
ú	u acute	u	'	either
Û	U circumflex	U	^	either
û	u circumflex	u	^	either
Ù	U grave	U	`	either
ù	u grave	u	`	either
Ü	U umlaut	U	"	either
ü	u umlaut	u	"	either
|	vertical line	/	^	either
Ÿ	Y umlaut	Y	"	either

Table 30. Character Generation (Special Characters)

(continued)

Generated Character		Compose Key, Plus This Pair		
Appearance	**Description**	**First**	**Second**	**Order**
Ÿ				
ÿ	y umlaut	y	"	either
¥	yen sign	y	-	either
		Y	-	either
		y	=	either
		Y	=	either

Transparent Mode

Table 31: Character Generation (Transparent Mode) on page 206 shows the symbol displayed for each character and control code when the VT emulator is in transparent mode. The characters at AA and BA are the feminine and masculine ordinals, respectively. The characters at 1E, 1F, 80, and 9E are underlined, although they may not appear underlined on the output.

Table 31. Character Generation (Transparent Mode)

	0x	1x	2x	3x	4x	5x	6x	7x	8x	9x	Ax	Bx	Cx	Dx	Ex	Fx
x0	@	P		0	@	P		p		p	Ÿ Ÿ	°	À	Đ	à	ð
x1	A	Q	!	1	A	Q	a	q	a	q	¡	±	Á	Ñ	á	ñ
x2	B	R	"	2	B	R	b	r	b	r	¢	²	Â	Ò	â	ò
x3	C	S	#	3	C	S	c	s	c	s	£	³	Ã	Ó	ã	ó
x4	D	T	$	4	D	T	d	t	d	t	¤	´	Ä	Ô	ä	ô
x5	E	U	%	5	E	U	e	u	e	u	¥	µ µ	Å	Õ	å	õ
x6	F	V	&	6	F	V	f	v	f	v	¦ ¦	¶	Æ	Ö	æ	ö
x7	G	W	'	7	G	W	g	w	g	w	§	• ▪	Ç	×	ç	÷

Table 31. Character Generation (Transparent Mode) (continued)

	0x	1x	2x	3x	4x	5x	6x	7x	8x	9x	Ax	Bx	Cx	Dx	Ex	Fx
x8	H	X	(	8	H	X	h	x	h	x	¨	¸	È	Ø	è	ø
x9	I	Y	)	9	I	Y	i	y	i	y	©	¹	É	Ù	é	ù
xA	J	Z	*	:	J	Z	j	z	j	z	a	o	Ê	Ú	ê	ú
xB	K	[	+	;	K	[	k	{	k	{	«	»	Ë	Û	ë	û
xC	L	\	,	<	L	\	l	|	l	|	¬	¼	Ì	Ü	ì	ü
xD	M	]	-	=	M	]	n	}	m	}	-	½	Í	Ý	í	ý
xE	N	^	.	>	N	^	m		n			¾	Î	Þ	î	þ
xF	O	_	/	?	O	_	n	Œ Œ	o	œ œ	¯ ¯	¿ ¿	Ï	ß	ï	ÿ

OIA Line Display Messages

During VT emulation, messages unique to VT can appear in certain columns of the OIA line. These columns display only VT messages, and do not display any of the messages that would appear there in 3270 or 5250 mode. Table 32: OIA Line Display Messages (VT only) on page 207 shows the meaning of each VT-specific message. Columns that are not mentioned in the table show messages common to all Z and I Emulator for Windows modes.

Table 32. OIA Line Display Messages (VT only)

Columns	Message	Meaning
1 through 7	VT340 7	Machine mode is VT340, seven-bit control.
	VT340 8	Machine mode is VT340, eight-bit control.
	VT100	Machine mode is VT100.
	VT52	Machine mode is VT52.
	VTANSI	Machine mode is VTANSI.
9 through 12	LOCK	Keyboard is locked.
30 through 39	OVERSTRIKE	New characters replace the character at the cursor position in Local Edit mode.
	INSERT	New characters move characters in page memory to the right in Local Edit mode.
61 through 64	HOLD	Screen is in hold mode.
66 through 69	EDIT	Local Edit mode is enabled.
71 through 72	Pn (n=1 through 6)	Current page number.

History Logging

You can use the Windows® scroll bar control to view history data from the current VT session. When configuring the session, select the **History Logging** option and specify the size of the log (see Configuring a VT Session on page 195).

When history logging is enabled, text is logged into the VT history window as it scrolls off the screen from the top margin row. The top and bottom margins are set when the host application defines the scrolling region, using the DECSTBM command sequence (`(ESC [ Pn ; Pn r)`).

Enhanced History Logging

When the host application sends the Erase in Display (ED) command sequence to erase a portion of the emulator screen, the contents can be logged into the VT history window before being erased. To enable this functionality, select the **History Logging – Enhanced** option, when configuring the VT session parameters.

The format of the host application ED command is `ESC [ Ps J`, where `Ps` is one of the following values:

0

Erases the screen contents from the cursor position to the end of the screen. This is the default setting.

1

Erases the screen contents from the beginning of the screen up to and including the cursor position.

2

Erases the entire screen contents.

When the ED command setting is 0 or 1, and **History Logging – Enhanced** is enabled, then the portion of the screen that is about to be erased will be logged into the history window before being erased. When the parameter value is 2, the entire screen contents are logged into the history window before being erased, regardless of whether enhanced history logging is enabled.

Chapter 12. ASCII Host File Transfer

Setting Preferences

You can set up your Workstation to perform file transfers; some preferences need to be set first, as described in this section. Other facilities to simplify handling of transfers are also described.

Using XMODEM and YMODEM

Z and I Emulator for Windows allows you to transfer files to and from ASCII hosts that support the XMODEM and YMODEM protocols. In order to use XMODEM or YMODEM, you must have established a connection to an ASCII host.

Choosing a Protocol

You have four choices for protocols. The one you select will depend upon the protocols supported by your ASCII host and by your particular requirements. The following table shows the capabilities of the protocols:

	Downloading		**Uploading**	
	Single File	**Multiple Files**	**Single File**	**Multiple Files**
XMODEM	Yes	No	Yes	No
XMODEM1K	Yes	No	Yes	No
YMODEM	Yes	Yes	Yes	Yes
YMODEMG	Yes	Yes	Yes	Yes

XMODEM

The XMODEM protocol is a single-file half-duplex protocol that performs error checking. Data is transmitted in 128-byte packets. Error checking, either by CRC or by checksum, occurs automatically. The Z and I Emulator for Windows implementation of XMODEM first tries CRC. If the sender fails to acknowledge the first three requests for CRC, XMODEM shifts to the checksum mode.

XMODEM1K

The XMODEM1K protocol is the same as XMODEM, except that it always uses CRC and has a larger packet size of 1024 bytes. Because some hosts are not able to handle the 1024-byte packets, there is a need for both XMODEM and XMODEM1K

YMODEM

The YMODEM protocol is similar to XMODEM, but it allows you to send multiple files in a single transfer. You may use a set of unique file names, or you may specify groups of files.

YMODEMG

The YMODEMG protocol is the same as YMODEM, supporting multiple files, but it does not supply error checking. It assumes that the data always transfers correctly, and is only for use with error-correcting modems. For large amounts of data it can achieve much greater throughput than YMODEM because it does not wait for packet acknowledgment.

XMODEM and XMODEM1K

To use XMODEM, click **Edit → Preference → Transfer** in your Z and I Emulator for Windows session. The Transfer Preferences window appears. Select the **XMODEM** or **XMODEM1K** protocol, and optionally click on the tab for the selected modem protocol to define the **Transfer Type** or to change advanced settings.

When receiving a file, in the Receive File from Host dialog box, enter the file name in the **PC File** field or select a personal computer file name from the drop-down listbox. The transfer type is automatically generated according to the templates.

YMODEM and YMODEMG

To use YMODEM, click **Edit → Preference → Transfer** in your Z and I Emulator for Windows session. The Transfer Preferences window appears. Select the **YMODEM** or **YMODEMG** protocol, and optionally click on the tab for the selected modem protocol to define the **Transfer Type** or to change advanced settings.

When receiving a file, you cannot select the personal computer file name, but you can change the default transfer type, the drive, and the directory, if necessary.

File-Transfer Timeout

You can define the time the workstation waits for a response from the host system (in seconds). If the host system does not respond, the transfer is canceled, and an error message appears. A number in the range 20–65535 (or 0) can be specified. The default is 60 seconds for ASCII sessions. Specify an appropriate value such that the error message does not appear too early. If you specify 0, a timeout is not set.

If a packet or block size is relatively large for low-speed lines, such as COM port lines, it is recommended that 150 seconds or greater be specified.

Extension for List-Files

You can change the default extension (.SRL) of file-transfer list files.

Show Status Window

You can choose the method of displaying the file-transfer-progress status.

In Session

When file transfer starts, the status window appears. The name of the file being transferred and the transfer progress appear.

In Icon

When file transfer starts, the status icon appears on the screen. If the icon is restored, the status window appears.

Defining Transfer Types

Transfer types define the option information used for controlling file transfer. Up to 32 transfer types can be defined for each host system. The original default types are: **delete** (deletes a file on abort), **over** (overwrites existing files) and **none** (does not delete on abort, and does not overwrite).

To add or change transfer types:

1. Click **Edit → Preference → Transfer**.
2. Click the tab for the modem protocol you have selected. The items that appear depend on the selected host system.
3. Enter transfer-type names in the **Transfer-Type** text box, or select them from the drop-down list.
4. To add or replace a transfer type, click **Save**. To delete a transfer type, click **Delete**.
5. Depending on the transfer type, select one of the following file receive options:

   Delete File on Abort

   With this option, if a file transfer is aborted then the incompletely received file is automatically deleted.

   Overwrite Existing File

   With this option, any existing file with the same name as the incoming file is overwritten.
   If you do not select this option, then a new name is given to the incoming file, according to the following scheme:

Existing file:	EXAMPLE.TXT
First contender becomes:	EXAMPLE.TX1
Second contender:	EXAMPLE.TX2
Tenth contender:	EXAMPLE.T10
Hundredth contender:	EXAMPLE.100
999th contender:	EXAMPLE.999

6. Click **OK**.

These options are independent of each other.

File Transfer Templates

For sending ASCII files, Z and I Emulator for Windows automatically generates Host file names and transfer types. For receiving ASCII files via XMODEM and XMODEM1K, Z and I Emulator for Windows generates a transfer type. In both situations, templates define the rules for file name and transfer type generation.

Defining Templates

The templates are common for all sessions and are used for both sending and receiving files. For ASCII host file transfer, you can define up to three templates for each protocol.

To display the templates panels, click the **Templates** button in the **Send File to Host** or **Receive File from Host** panel.

You can add, delete, or replace templates; you can also test templates to see how Z and I Emulator for Windows generates the target file name and transfer type.

When defining templates, you can use * (asterisk) for the global searching of file names; for example, *.EXE for all files that have a file name extension of EXE.

Automatic Generation of File Names

The templates are numbered from 1 to 32; when Z and I Emulator for Windows generates file names, the templates are searched, starting from 1, and the first template that matches is used.

Example of ASCII Protocol Template

The following example shows the use of templates for ASCII host file transfer. When sending files, Z and I Emulator for Windows automatically generates a host file name from a personal computer file name, and vice versa. It also generates a transfer type. When receiving files, Z and I Emulator for Windows automatically generates only transfer types, and only for the XMODEM and XMODEM1K protocols.

For more information about templates, refer to *Administrator's Guide and Reference*.

Following are the definitions of the three default templates. The template is selected from the available choices by matching the name of the file being transmitted or received against each template's file specifications.

Template Number	Wildcard specification for PC File	Wildcard specification for host File	Type
1	*.exe	*.*	delete
2	*.txt	*.*	over
3	*.*	*.*	none

Send Example: If you enter `program.exe`, Z and I Emulator for Windows selects template 1, and displays `program.exe delete` in the list box.

Receive Example: (XMODEM AND XMODEM1K only) If you enter `program.exe`, Z and I Emulator for Windows selects template 1, and displays `program.exe delete` in the list box.

Working with Lists of Files

For transferring a group of files it is convenient to use a list. A list makes it easier to transfer the same groups of files frequently, with a single command. Even if you are transferring a group of files only once, a list can help prevent errors. A list of files is itself a file.

You can transfer multiple files at once by using the send/receive list; it is accessible from the Send Files to Host or Receive Files from Host windows. For either window, the files selected are displayed in a **Transfer List**. This list can be saved, and later retrieved and modified. For instructions on selecting a file, see Receiving Files from an ASCII Host on page 214 and Sending Files to an ASCII Host on page 215.

File Name Extension for List Files

By default, send/receive list files have a file name extension of .SRL. You can change this default on the property page with the **General** tab, by clicking **Preferences → Transfer** from the **Edit** menu.

Note: Z and I Emulator for Windows does not recognize a file as a send/receive list file unless its name has the specified extension.

Remove From List

By clicking the **Remove** button, you can delete the selected file from a send/receive list.

Open List File

If you click the **Open List** button, the Open File-Transfer List File dialog box appears, allowing you to manipulate the file names in the list.

Save List File

If you click the **Save** button, the Save File-Transfer List File As dialog-box appears and you can save the list of files.

Changing a List of Files

You can make changes to a list of files to be transferred:

Change the Personal Computer or Host File Name

When you select a file to send or receive, Z and I Emulator for Windows automatically generates a host or personal computer file name by using templates. To change the generated file name, just type over it.

Note: When receiving a file from an ASCII host, you specify the host file name on the host system.

For receiving files, you can select a personal computer file from the dialog obtained by clicking the **Browse** button.

Note: The browse function is not available when *receiving* files from an ASCII host; it is available when sending files, but only when using the YMODEM or YMODEMG protocols.

Delete File Names From List

To delete a file from the list, select it from the list and click the **Remove** button.

Add More File Names To List

To add more files to the list, select a file in the **PC File** list box with Ctrl + left mouse-button, or type a file name in the **PC File** entry field and press Enter.

Receiving Files from an ASCII Host

Receive File From Host allows you to receive files from a host system to your personal computer; with one command, you can receive a single file or several. If you often receive the same list of files, you can save the list of file names and receive all the files with one command.

For ASCII host file transfer, the host system must support one of two protocols, XMODEM or YMODEM.

Selecting a Workstation Directory

To receive files to a workstation directory, you can key in the directory information or click the **Browse** button to open the Browse dialog and select the directory; this can be done as part of setting preferences (setting the **Default PC Directory** field) or at the time of the file transfer.

Selecting Files to Receive

For ASCII host file transfer, select the file to receive on the host system.

Follow these steps to receive one or more files from an ASCII host:

1. Prepare the host system. The exact method of preparation, including selection of file names, depends on the kind of host system to which you are connected. Contact your host-system administrator for details.

   Note: The host system must support one of two protocols: XMODEM or YMODEM.
2. Click **Edit → Preference → Transfer** to display the Transfer Preferences window. Select the type of protocol you want to use from the drop-down list box on the property page with the **General** tab.
3. In the **Default PC Directory** field, type the workstation directory where the file or files should be sent; or, click the **Browse** button to open a dialog and select the directory.
4. To change the transfer parameter defaults for the protocol you selected, click the tab to display the property page for the selected modem protocol.
5. When all preferences have been set, click **OK**.
6. Click **Receive File from Host** from the **Actions** menu. The **Receive File from Host** window appears.
7. For XMODEM and XMODEM1K, click the **Browse** button to open a dialog and select a personal computer file name or names, or enter the names in the **PC File** entry field. The transfer type is automatically generated and appears in the **Transfer Type** entry-field.
8. For YMODEM and YMODEMG, select the transfer type and click the **Browse** button to open a dialog and change the directory, if you desire.
9. Click the **Receive** button to display the **Receive Files Status** window and start the transfer.

Sending Files to an ASCII Host

Send file to host allows you to send files from your personal computer to the host system; with one command, you can send a single file or several files. If you often send the same list of files, you can save the list of file names, and subsequently send all the files with one command.

Note: This is supported using the YMODEM and YMODEMG protocols only.

Selecting Files to Send

There are several ways to select files to send:

Basic Methods

Type a file name in the **PC File** field and press Tab; a host file name and a transfer type are generated automatically according to the templates.

Select files from the dialog obtained by clicking the **Browse** button.

Select from a Send/Receive List

If you have saved a list of file names in a send/receive list, click the **Open List** button and select the list you want to use; the file names saved in the list appear.

Note: For ASCII host file transfer, you can use the send/receive list only with the YMODEM and YMODEMG protocols (not with XMODEM or XMODEM1K).

Advanced Method

The Browse window, obtained by clicking the **Browse** button, displays all the files in the current directory; you can display only certain types of files if you want to.

For example, if the directory has many files and you want to display only files that have the extension .DOC, you can type `*.doc` in the **PC File** field and click the **Browse** button; the resulting dialog shows only files that have the extension .DOC.

Changing the Host File Name or the Transfer Type

When you select a file to send, Z and I Emulator for Windows automatically generates a host file name and selects a transfer type from the default templates. You can change the file name by typing over the text in the **Host File** field; you can change the transfer type by selecting a different one from the **Transfer Type** drop-down list.

Saving a List of Files to Send

If you frequently send the same set of files, it is a good idea to save the names in a list, called a send/receive list.

Note: For ASCII host file transfer, you can use the send/receive list only with the YMODEM or YMODEMG protocols (not with XMODEM or XMODEM1K).

Sending a List of Files

Select the list, then click the **Send** button.

PC Code Page

When a file is transferred, EBCDIC codes are converted to 1-byte workstation codes, and vice versa. A valid value is automatically selected from among the following values for SBCS sessions: 437, 737, 806, 813, 819, 833, 850, 852, 854, 857, 858, 860, 861, 862, 863, 864, 865, 866, 869, 874, 912, 915, 916, 920, 921, 922, 1008, 1089, 1124, 1125, 1127, 1129, 1131, 1133, 1153, 1155, 1156, 1157, 1158, 1160, 1164, 1250, 1251, 1252, 1253, 1254, 1255, 1256, 1257, and 1258; —according to the host code page specified when the workstation is configured. For an explanation of how to select host code pages, see the online help for the host code page.

Part V. Troubleshooting

Chapter 13. Troubleshooting tips

1. Connecting to z/OS console via Z and I Emulator for Windows 62x160 screen size results in error IEE938I

Problem

IEE936I CONSOLE HHSP0141 INITIALIZATION ERROR - RC:01 - 3277-2 IS ASSUMED is logged by z/OS console when using HCL Z and I Emulator for Windows version 6.0.2 with a 62x160 screen size.

Cause

z/OS console connection is being made through the OSA-ICC telnet server. This server does not support the Terminal Type of IBM-DYNAMIC.

APAR IC71220 changed the default Terminal Type string passed to the telnet server for screen sizes greater than 27x132.

Resolution

To use 62x160 screen size display sessions with OSA-ICC telnet servers, take the following steps:

1. Edit the WS session profile.
2. Add the following statements under the Telnet3270 stanza to force HCL Z and I Emulator for Windows to use the original default string:

```
[Telnet3270]
TerminalTypeString=IBM-3278-2-E
```

2. Z and I Emulator for Windows TCP/IP Data Transfer fails with terminated transfer function

Problem

TCP/IP Data Transfer might fail with the error "Transfer function will be terminated". This message provides a checklist for iSeries or AS/400 host requirements.

Resolution

To resolve this problem, take the following steps to check the status of the iSeriesTM or AS/400® host requirements:

1. Confirm that the iSeries or AS/400 host has the following required licensed program(s) installed (type GO LICPGM and select Option 10 - Display Installed Licensed Programs:):

```
For V5Rx:
5722TC1 *BASE iSeries TCP/IP Connectivity Utilities/400
5722SS1 Option 12 OS/400 ® - Host Servers

For V4R5:
5769TC1 *BASE TCP/IP Connectivity Utilities for AS/400
5769SS1 Option 12 OS/400 - Host Servers
```

2. Verify that you have a Relational Database entry for YOURSYSTEM at REMOTE Location = *LOCAL (issue the command WRKRDBDIRE to "Work with Relational Database Directory Entries"):

   ```
   RELATIONAL DATABASE . . . . . . : YOURSYSTEMNAME
   REMOTE LOCATION:
   REMOTE LOCATION . . . . . . . : *LOCAL
   TEXT . . . . . . . . . . . . . . :
   ```
3. Verify that user ID QUSER is enabled (issue the following command for profile QUSER):

   ```
   WRKUSRPRF USRPRF(QUSER)
   ```
4. Verify that TCP/IP is active, using the following command (This is required before any TCP/IP processing):

   ```
   START TCP/IP (STRTCP) COMMAND
   ```
5. If you have not already done so, issue the following "Start Host Server" command:

   ```
   STRHOSTSVR SERVER(*ALL)
   RQDPCL(*TCP)
   ```

The following sets of instructions are to verify that the required server daemon and prestart server jobs are active for DDM, Database, and File Transfer functions.

1. For DDM, take the following steps:
   a. Verify that the DDM daemon is active. Use WRKACTJOB to find the QRWTLSTN job under QSYSWRK. This daemon is automatically started when the STRTCP command is run, if the AUTOSTART parameter is set to *YES on the CHGDDMTCPA command <F4>. If the daemon does not start when the STRTCP command is run, you can start the daemon by issuing the following command:

      ```
      STRTCPSVR SERVER(*DDM)
      ```

      Note: If the DDM daemon still fails to start with error message CPF3E30, refer to APAR SA81267.

   b. Verify that the DDM prestart server jobs are active. Use WRKACTJOB to find the server jobs named QRWTSRVR. For V5R2 and newer releases, the prestart jobs run in the QUSRWRK subsystem, but can be configured to run under other subsystems. They are automatically started with the subsystem. If these jobs are not active, you can issue the start prestart job command STRPJ <F4>. For host versions prior to V5R2, these jobs runs in the QSYSWRK subsystem.
2. For Database, take the following steps:
   a. Verify that the subsystem for the Database server daemon is active. The subsystem is QSERVER, and the daemon job is named QZDASRVSD.
   b. Verify that the Database prestart server jobs are active. For V5R1 and newer releases, the prestart jobs are QZDASOINIT and QZDASSINIT. Both jobs run in the QUSRWRK subsystem, but can be configured to run under other subsystems. As with the DDM

prestart jobs, these jobs are automatically started with the subsystem, but if they are not active, they can be started with the STRPJ command.

   c. If port 8478 is not active, end and restart the Database server job QZDASRVSD using the following commands:

      ```
      ENDHOSTSVR *DATABASE
      STRHOSTSVR *DATABASE
      ```

3. For File Transfer, take the following steps:
   a. Verify that the subsystem for the transfer function server is active. The subsystem is QSERVER, and the daemon job is named QZDASRVSD.
   b. Verify that the transfer function prestart server jobs are active. The prestart job is QTFPJTCP. This job runs in the QSERVER subsystem.

3. Z and I Emulator for Windows Telnet connection timeout with error 657

Problem

Keyword for Telnet timeout.

Resolution

Z and I Emulator for Windows uses non-blocking sockets when connecting to a remote system. By default, Z and I Emulator for Windows waits for 3 seconds to establish the socket connection. When accessing remote terminals over a dialup network, this default wait time might not be sufficient and could lead to a connection failure and return the error 657.

The default time can be changed by adding the following keyword to the Telnet3270 or Telnet5250 stanza of the workstation profile:

```
[Telnet3270]
InactiveTimeout=xx
```

The value `xx` is the time in seconds and can take any positive numerical values.

4. PCSXFER041 timeout during Z and I Emulator for Windows file transfer TSO session

Problem

Three situations might cause a timeout at the Z and I Emulator for Windows client and a solution is provided for each case.

Cause

Resolution

CUT mode transfer

By default, Z and I Emulator for Windows uses DFT mode for transfers to and from a TSO host. When using Dial to connect to TSO, the host might be configured to do the transfer in CUT mode, while Z and I Emulator for Windows is still in DFT mode.

You can modify the workstation profile that you use to dial the TSO session. To manually change the DFT/CUT setting, add the following case-sensitive line to the Transfer stanza of the workstation file:

```
[Transfer]
CUTprotocol=Y
```

You can also change the LU description on the zSeries host. Change the VTAM BIND image so that the Write Structured Field query support bit is set in the PSERVIC. Refer to the VTAM documentation for more information.

Using \FT trigger in SuperSession Manager

If you encounter the timeout problem when using Supersession, you can use the \FT trigger before logon. This can be applied on an individual user and session basis.

The \FT trigger invokes the file transfer script KLSXFER, which enables query passthru. It also inhibits immediate broadcasts and session locking.

Document mode and Word Wrap mode

If Document mode or Word Wrap mode is enabled, the file transfer will fail. If you have a key mapped to toggle this feature on or off, turn it off before initiating the file transfer. If you want to disable this feature, change the following case-sensitive line in the ENTRYASSIST stanza of the workstation file:

```
[ENTRYASSIST]
DocmodeWordWrap=N
```

Appendix A. Notices

This information was developed for products and services offered in the United States. HCL may not offer the products, services, or features discussed in this information in other countries. Consult your local HCL representative for information on the products and services currently available in your area. Any reference to an HCL product, program, or service is not intended to state or imply that only that HCL product, program, or service may be used. Any functionally equivalent product, program or service that does not infringe any HCL intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-HCL product, program, or service.

HCL may have patents or pending patent applications covering subject matter described in this information. The furnishing of this information does not give you any license to these patents. You can send license inquiries, in writing, to:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

HCL TECHNOLOGIES LTD. PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you..

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the information. HCL may make improvements and/or changes in the product(s) and/or program(s) described in this information at any time without notice.

Any references in this information to non-HCL documentation or non-HCL Web sites are provided for convenience only and do not in any manner serve as an endorsement of those documents or Web sites. The materials for those documents or Web sites are not part of the materials for this HCL product and use of those documents or Web sites is at your own risk.

HCL may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085

USA
Attention: Office of the General Counsel

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by HCL under terms of the HCL Customer Agreement, HCL International Programming License Agreement, or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-HCL products was obtained from the suppliers of those products, their published announcements or other publicly available sources. HCL has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-HCL products. Questions on the capabilities of non-HCL products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

Trademarks

HCL, the HCL logo, and hcl.com are trademarks or registered trademarks of HCL Technologies Ltd., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM® or other companies.