MICRO FOCUS

Rumba+ 9.5

System Administrator Guide

Contents

Using Rumba 41

Using Rumba+ 53

AS/400 Getting Connected 57

About This Guide

The Rumba *System Administrator Guide* has been written to help system administrators to:

- Plan the deployment of Rumba
- Install Rumba
- Support end users
- Troubleshoot Rumba

This guide contains a number of sections, each focussing on one of Rumba's many features.

Conventions used in this guide

The following conventions are used in this guide:

Item	Description
`File names`	File names and extensions appear in `monospace` type.
UI element	User interface elements appear in **bold**.
Note: Note text	An item of interest.
`Command line`	Text that appears in the command line (either entered by the user or generated as a response to user entry) appears in `monospace` type.
Blue text	Cross-references to additional information appear blue text.

How this guide is organized

The *System Administrator Guide* contains the following sections:

Introducing Rumba	Provides an introduction to Rumba and describes the features provided with Rumba.
Installing Rumba	Describes how to install Rumba on Windows with Microsoft Installer.
Using Rumba	Describes how to use the main features of Rumba.
Using Rumba+	Describes how to use the main features of Rumba+.
AS/400 Getting Connected	Describes how to configure and maintain Rumba AS/400 display and printer connections.
AS/400 File Transfer	Describes the AS/400 file transfer program.
AS/400 Client Application	Describes the AS/400 Client Application which allows you to start and control non-interactive programs on your AS/400 host system without opening a display session.
Mainframe File Transfer	Describes the mainframe file transfer program.
HP and UNIX File Transfer	Describes the HP and UNIX file transfer programs.

AS/400 and Mainframe Printers	Provides an overview of the AS/400 and mainframe printer features.
Rumba Telnet Edition	Describes how you can use Rumba over TCP/IP to connect to multiple hosts, transfer files, and share host file systems with PCs.
Automating Rumba	Describes how to use macros and scripts to perform automated Rumba tasks.
Connection Tools	Describes the most commonly used Rumba tools: Rumba APPC Configuration utility, Administrator Override File, Rumba AS/400 Communications utility, and Rumba Communication Monitor.
Diagnostic and Development Tools	Describes the Rumba Trace Console, Rumba development tools for .NET, and the Rumba Development Add-in for Visual Studio.

Available documentation

In addition to this guide, Rumba documentation consists of the Readme file and the Rumba Help.

Readme file	The Readme file contains late-breaking information on Rumba. There are many Rumba products and some of them might share the same Readme file. Therefore, the Readme file might contain information about other Rumba products, apart from the product you are currently installing.
Rumba Help	The Help contains information on how to customize and use Rumba. The Help does not include information on installing Rumba. For information on installing, see *Installing Rumba*.

To open the Help:

- Select **Help** > **Rumba Help Topics** (classic interface).
- Click the Help icon above the Rumba ribbon (ribbon interface).

Introducing Rumba

This section provides a summary of the features that are included in Rumba.

The section introduces the entire Rumba product line, which provides access to multiple host types: IBM mainframe, IBM AS/400, UNIX, OpenVMS, and HP. Rumba may be able to access one or all of these host types, depending on your individual needs.

About Rumba

Rumba is a software suite that connects your users to information on multiple hosts and servers across your organization. With Rumba, you have a choice of protocols, such as TCP/IP and Rumba Router (MPTN).

All Rumba products operate on Windows 7, 8.1, and 10, WTS, and Citrix Presentation Server. In addition to providing multiple access solutions, the Rumba product suite:

- Gives users tools with which they can view, print, and transfer data located anywhere in the organization. These tools help users attain a higher level of efficiency.
- Gives system administrators flexibility in deploying and maintaining Rumba.
- Offers developers the functionality needed to quickly generate custom client/server solutions using ActiveX technology.

Rumba products are designed to meet specific access needs. Micro Focus produces and packages Rumba products to support multiple or single host access needs, with multiple connection solutions. We also provide Rumba versions for popular LANs, and migration products that help you move from one Rumba product to another, as well as from competitor products to Rumba.

Rumba products

The following table lists multiple-host Rumba products:

Rumba product	Product description	Connection requirement
Rumba OFFICE	- Supports IBM mainframe, IBM AS/400, Digital OpenVMS, UNIX systems, HP 3000 and 9000 systems. - Supports multiple servers and network protocols. - Supports display, printing, and file transfer capabilities for all hosts. - Provides browser-to-host connections and an integrated set of Internet applications.	Multiple network protocols.
Rumba Telnet Edition	- Supports IBM mainframe, IBM AS/400, Digital OpenVMS, UNIX systems, HP 3000 and 9000 systems. - Designed specifically for the Telnet network protocol	Telnet, TCP/IP.

Rumba product	Product description	Connection requirement
	• Supports display, printing, and file transfer capabilities for all hosts. • Provides browser-to-host connections and an integrated set of Internet applications.	

Rumba applications

The table below describes the applications that make up the Rumba suite:

Rumba application	Feature description
Rumba Mainframe Edition	• Supports IBM mainframe. • Supports multiple servers and network protocols. • Supports display, printing, and file transfer capabilities for the mainframe host. • Provides browser-to-host connections and an integrated set of Internet applications.
Rumba AS/400 Edition	• Supports IBM AS/400. • Supports multiple servers and network protocols, including native TCP/IP. • Supports display, printing, and file transfer capabilities for the AS/400 host. • Designed to maximize the file, print, and database serving capabilities of the AS/400 with such features as Shared Folders, Submit Remote Command, Virtual Print, Data Queues, and Rumba AS/400 Optimized Server Driver. • Provides browser-to-host connections and an integrated set of Internet applications.
Rumba UNIX-HP Edition	• Supports Digital OpenVMS and UNIX systems. • Supports multiple servers and network protocols. • Supports display, printing, and file transfer capabilities for the UNIX and VMS host. • Provides browser-to-host connections and an integrated set of Internet applications. • Supports HP 3000 and 9000 systems, and UNIX systems supporting HP terminal types. • Supports multiple servers and network protocols. • Supports display, printing, and file transfer capabilities, including MPE native file transfer, for the HP host. • Provides browser -to-host connections and an integrated set of Internet applications.

New Rumba features

Rumba provides the following new features:

EML	Express Macro Language (EML) is a new, code-free macro language that can be shared across mainstream Micro Focus emulators.
VBA scripting	Create macros using Microsoft Visual Basic for applications (VBA) scripting. The VBA Add-On is required.
MSS support	• X.509 support - smartcard • Metering server
Office 64-bit	Rumba now supports 64-bit editions of Microsoft Office.
Plus modernization	**Adding controls in a Plus session** Add a subset of controls to screens during a live Plus session without using the Screen Designer.

Table control

- Modify data on multiple screens.
- Specify the location of an action to be performed relative to the location of the selected table row.

Chooser control

Load data dynamically from an external source.

Label control

Tranlsate the text of specified Label controls to another language.

New MessageBox control

A configurable alert message box.

New MultiLine control

Input field with multiple text lines which are added at a specified screen location.

Rule Manager

- New exclusion method for rules.
- Sort rules by names.
- Support of operands in rules.
- Use variables in rules.

Bold / italic / underline support for controls with caption

Added functionality for those controls that have captions.

Host-specific features

This section describes the features specific to individual Rumba host applications.

Mainframe features

The mainframe display is used to view host applications and transfer files. The mainframe printer is used to route jobs from the mainframe printer queue to a Windows printer. Both the mainframe display and printer use interfaces to connect to the host.

The table below describes the standard features and interfaces of Rumba Mainframe Edition:

Feature	Description
Display	• Presents the traditional mainframe display terminals (3278/79) in a Microsoft Windows application: the Rumba desktop. • Includes GDDM support, which makes working with IBM mainframe graphics similar to using Windows graphics packages. • Users can run multiple host sessions, displaying and working with PC and host applications simultaneously.
File transfer	• Supports host file transfer programs IND$FILE and SimXfer. • Supports ASCII or binary transfers, Write Structured Field or Buffered data streams, EBCDIC-to-ASCII translation, and data compression. • Provides DOS-based, batch, and drag-and-drop file transfer features.
Switching system keyboards	• Allows switching keyboard displays as in Windows (ALT+SHIFT).

The table below describes the standard features and interfaces of the Rumba printer:

Feature	Description
Rumba Mainframe Printer (3287 emulation)	Allows users to route jobs from the mainframe printer queue to a Windows printer.
PC printer queues	Lists the printer queues for all network and local printers set up in Windows. Users can use PC Printer Queues to control Windows printers.
Print log	Lists documents that have been printed using the Rumba printer, including documents printed from the host as well as from PC applications.

The table below describes the interfaces used to connect to the host:

Feature	Description
Microsoft SNA Server	Connects to mainframe hosts through a suite of integrated services. SNA Server acts as an SNA gateway using two types of connections, client-to-server and server-to-mainframe.
MPTN (SNA over TCP/IP)	Connects to a mainframe host using SNA APPC to communicate over TCP/IP networks. Provides transparent access to functions such as mainframe printing and file transfer.
TN3270E	Uses a TCP/IP protocol stack to connect to a mainframe host via a Telnet server.

AS/400 features

The following features are available in the AS/400 display, printer, and AS/400 File Transfer:

• Spell checking
• Automatic hyperlink detection

• Automatic e-mail address detection

UNIX and HP features

When using Rumba UNIX-HP Edition, users perform the majority of their tasks with the host using the UNIX or the HP display. This includes viewing host applications, transferring files, and printing from the host. The UNIX display and the HP display connect to a host over a TCP/IP interface.

The table below describes the standard features for UNIX and HP:

Standard feature	Description
Display	Presents the traditional VT display terminals in a Microsoft Windows application supporting the whole range of VT terminals from VT 52 to VT 420, as well as VT340 ReGIS, Sixel, SCO ANSI, BBS ANSI and Tektronix 4010/4014 graphics support.
	Users can run multiple host sessions, displaying and working with host applications simultaneously.
File transfer	Supports Kermit, Text, and X-, Y-, Z-Modem file transfer protocols over all the supported interfaces. It supports a Rumba initiated mode where the user simply selects the protocol and file to be transferred, and Rumba provides the underlying commands to successfully transfer the file.
Enhanced security	Supports SSL 3.0 (Secure Sockets Layer), TLS 1.0, 1.1, 1.2 (Transport Security Layer), and SSH (Secure Shell) to allow users to securely access applications and transfer data between their desktops and mainframe, AS/400 and UNIX systems, encrypting the session data stream to a host capable of supporting these protocols.
Switching system keyboards	Allows switching keyboard displays as in Windows (ALT +SHIFT).
Keyboard shortcut to open Split Display or History window.	Provides a keyboard shortcut (CTRL+SHIFT+Z) to open the Split display and History window on a UNIX Display.
Double-click auto select	When connecting to UNIX and double-clicking data, the data is automatically selected and copied to the clipboard. When you right-click at the new location, the data is pasted without displaying the pop-up menu.

The table below describes the standard features of the HP-UNIX printer:

Standard feature	Description
Terminal-routed printing	Prints the display screen, selected text, and full color graphics, or routes print jobs from the host printer through the Windows print driver.

The table below describes the interfaces used to connect to the host:

Standard feature	Description
Async	Connects to a remote UNIX host through a phone line and a standard asynchronous modem or directly to the host or terminal controller using an RS-232 direct connection.
Telnet	Connects to the UNIX host using the TCP/IP protocol.

The table below describes the standard features for HP:

Standard feature	Description
Display	Presents the traditional HP terminal types in a Microsoft Windows application.
	Users can run multiple host sessions, displaying and working with host applications simultaneously.
File transfer	Supports Kermit, Text, and Xmodem, Ymodem, Zmodem file transfer protocols over all the supported interfaces, and a proprietary protocol to HP3000 servers running the MPE operating system. It supports a Rumba initiated mode where the user simply selects the protocol and the file to be transferred, and Rumba provides the underlying commands to successfully transfer the file.

The table below describes the standard features of the HP printer:

Standard feature	Description
Terminal-routed printing	Prints the display screen or selected text, or routes print jobs from the host printer through your Windows print driver.

The table below describes the interfaces used to connect to the host:

Standard feature	Description
Async	Connects to a remote HP host through a phone line and a standard asynchronous modem, or directly to the host or terminal controller using an RS-232 direct connection.
HP NSVT	Connects to an HP host using the proprietary HP protocol for Network Services Virtual Terminals (NSVT).
Telnet	Connects to the HP host using the TCP/IP protocol.

Core features

This section provides an overview of the core features of Rumba. You can find more specific information in other sections of this guide.

🖉 **Note:** Which hosts you can connect to depends on which edition of Rumba you have installed. For example, Rumba AS/400 Edition enables you to connect to AS/400 hosts.

Rumba takes full advantage of the Windows graphical environment and tabbed browsing capabilities. Together, these features help improve user efficiency, and simplify the sharing and manipulation of data among multiple PC and host platforms.

Using Rumba, you can:

- Access multiple host sessions simultaneously from the same window
- Switch system keyboards as you do in Windows (ALT+SHIFT)
- Copy and paste data between tabs or windows
- Capture screen data to a file
- Copy information to the Windows Clipboard (including, in Rumba Mainframe Edition, the ability to copy only unprotected fields)
- Use the History feature to maintain a larger list of recently performed operations and scroll back to them easily
- Rumba AS/400 Edition: Use long passwords to provide an additional authentication mechanism with stricter encryption and tighter security

Multiple HLLAPI sessions

You can run multiple HLLAPI sessions originating from the same profile. This is in addition to the API options currently available in Rumba.

Enhanced security through SSL/SSH

With the inclusion of SSL 3.0 (Secure Sockets Layer), TLS 1.0, 1.1, 1.2 (Transport Security Layer), and SSH (Secure Shell) support, users can securely access applications and transfer data between their desktops and host systems, encrypting the session data stream to a host capable of supporting these protocols.

Certificate Express Logon

With the inclusion of Certificate Express Logon (CEL) support, users can securely logon to host applications without sending their user name and password to a host, where the host is capable of supporting this protocol.

Note: When a user needs to choose a client certificate, the host disconnect timeout must be long enough for the user to do so.

Host sessions

Rumba runs on a single-user PC connected to a host, or on a local area network (LAN) with a gateway connection to a host. Rumba Mainframe Edition can also connect to a host using multiple Physical Unit IDs (PUIDs) with a Token Ring or Ethernet connection. From a single window, you can run multiple host sessions using different connection types.

You can also retrieve data from one or more hosts and transfer it to PC applications or print the data.

Rumba printer

Rumba AS/400 and Mainframe Editions support a wide range of printer features, including the ability to:

- Save and print host screens
- Print to a disk
- Route screen output to a LAN print server for shared printing
- Use international character sets

Copy and paste

Rumba provides full interoperability between your host applications and other Windows applications. As well as normal copy and paste operations, you can use the Paste Link facility to automatically update the data in a Windows application whenever both it and a Rumba session to the host are open.

File transfer

The file transfer feature copies files between your PC and the host. Rumba allows you to navigate through the host file system, listing host and PC directories to simplify file selection and transfer. When you select the Rumba Initiated File Transfer feature, Rumba automatically starts the file transfer program on the host and sends or receives the file.

Mainframe file transfer

The file transfer feature copies files between your PC and the mainframe. The Send and Receive commands open a dialog box where you specify the file type, transfer protocol, and host environment for the file transfer. You can also choose to have Rumba start the file transfer protocol. The mainframe file transfer feature works transparently with SimXfer and most IBM host operating systems, including CICS, TSO, and VM/CMS.

The mainframe file transfer also provides abundant functionality such as support for ASCII or binary transfers, Write Structured Field or Buffered data streams, EBCDIC-to-ASCII translation, and data compression. Rumba also supports DOS-based, batch, and drag-and-drop file transfer features.

AS/400 file transfer

There are two ways to run file transfers from an AS/400 Display session:

- AS/400 File Transfer

  This method sends data between the host and your PC. If the AS/400 host is not in passthrough mode, the **Send** and **Receive** commands open the **AS/400 File Transfer** window.
- Mainframe Passthrough

  This method uses the AS/400 as an emulator to transfer files to and from a mainframe host. You must first connect Rumba to the AS/400, then run the emulator program on the AS/400 to connect to the mainframe. Rumba detects the passthrough state of the AS/400, so you can use the **Send** and **Receive** commands to transfer files between the mainframe host and your PC.

User interface

Rumba provides two different versions of the user interface. You can switch between these at any time according to your personal preferences.

Classic An interface similar to that used in previous versions of Rumba, for users experienced with older versions of Rumba.

Ribbon Originally introduced by Microsoft Office 2007, this interface organizes commands into a 'ribbon' across the top of the application window. The commands shown change according to the tab that has been selected. This makes application features easier to find and quicker to use.

Screen displays

Rumba takes full advantage of the GUI capabilities of Windows to offer these features:

- Support of alternate screen sizes, adjusting the host application to fit within the window.
- Simultaneous display of applications in a variety of screen sizes without reconfiguring your PC software or modifying the host application.
- Customization of screen color schemes for existing host applications.

   Note: Rumba Mainframe Edition supports alternate screen sizes for IBM models 2, 3, 4, and 5.

International keyboards

Rumba supports international PC and host keyboards, such as French, German, Norwegian, Swedish, Danish, Spanish, and Italian.

Rumba supports most types of PC keyboards. For example, the PC, AT, and Enhanced 101- and 102-Key. If you do not find your keyboard on the supported list, contact Micro Focus SupportLine.

You must install the appropriate keyboard driver (usually supplied with the keyboard or an international version of Windows) using the Windows Setup option.

Supported VT features

Rumba supports configuration of VT features and options through the keyboard, tool bar, and menus. Features available to you when using a VT terminal and keyboard are available from your PC using Rumba. See *VT Keyboard Key Codes*.

Rumba emulates the following VT terminal types:

- VT52
- VT100
- VT101
- VT125
- VT220
- VT240
- VT241
- VT320
- VT330
- VT340
- VT420 (paging available. Default if VT340 not installed)
- 401x (Tektronix graphics, includes 4010/44014. Automatic font adjustments occur when entering this mode)
- BBS ANSI
- SCO ANSI
- WYSE 50/60

You can select and configure the terminal you want to emulate.

Supported HP features

Rumba supports configuration of HP features and options through the keyboard, tool bar, and menus. Features available to you when using an HP terminal and keyboard are available from your PC using Rumba.

Rumba emulates the following HP terminal types:

- HP70092
- HP70094
- HP70096
- HP70098
- HP2392A
- HP2394A

You can select and configure the terminal you want to emulate.

Automating your work

Rumba provides several powerful tools for automating host communications tasks.

- Macros

  Rumba macros are used to automate simple tasks such as logging on, copying and pasting text between Rumba display windows and applications such as Microsoft Excel, or sending a sequence of commands to the host.

- Auto macros

  You can save a macro as a connect or disconnect macro - a macro that automatically logs on or off each time you open or close the session profile in which it was created.
- Scripts

  The separate Rumba Script Editor provides more complex and powerful automation options than the Macro Editor. You can use scripts to automatically reconfigure the user interface, perform random access file operations, automate tasks, modify functionality, and handle incoming events.

  Where provided, the Rumba Script Editor can be installed by double-clicking the `ScriptEngine.msi` file.

Rumba tools

All Rumba products contain the following tools:

- Productivity tools
- Configuration and maintenance tools
- Developer tools

More complex and powerful task automation can also be provided by the Script Editor and Script Player applications, provided separately.

Productivity tools

Rumba applications contain tools that help end users attain a higher level of efficiency by eliminating time-consuming file and application manipulation. For example, with these tools, users can copy from a host database and paste to another application, like a spreadsheet or word processor. The **Paste Link** facility automatically updates the data in a Windows application whenever both it and a Rumba session to the host are open.

Other useful features include:

| Office integration | From within the Rumba desktop, users can instantly send the current screen text or image, fully or selectively into any Office item. |

Microsoft Office .NET primary interoperability assemblies (PIA) enables .NET managed code to interact with a Microsoft Office application's COM-based object model.

Office version	PIA auto installer	Rumba support
2013	Yes	Supported
2010	Yes	Supported
2007	Yes	Supported

Session screens history tracking
Rumba introduces an embedded, fully visual way of tracking screens visited throughout a session. Users can go back to previously visited host screens and work on them. This ability is enhanced with search capabilities, database navigation, export and import support for session sharing, information sorting, and automatic or manual capturing options.

Toolbars enhancements
Support for user-defined, customized toolbars that can be undocked and 'floated', rather than just fixed and docked to the top of the application. Ability to define more types of buttons and the ability to export and import toolbars to and from files.

Enhanced desk management

The Rumba desktop is designed to support the tiling and repositioning of all session types within the Rumba desktop in an easy, dynamic, and intuitive way. The layout of the desktop can then be saved in a desktop profile to be reused when Rumba is restarted.

As a system administrator, you can help your users by:

- Automating log-on procedures by creating macros for them
- Setting up batch files that they can use to print multiple host screens to a PC printer (Print Area Setup)

Many of these features can help personalize the workplace. For example, you and your users can change the look of a Rumba display by using watermarks or customizing the display.

The following table describes the more important Rumba tools:

Tool	Description
Rumba Desktop	A tabbed interface to Rumba that makes it easy to manage several different sessions at once, even to different hosts.
Profiles	Saves Rumba customized display/printer and interface configuration information. Using profiles, a user can tailor a Rumba session for a specific host or application and save it for re-use. Session profiles save information relating to a single host session, whereas desktop profiles save a set of session profiles.
Print screen	Specifies how to print a screen within the Rumba window. Users can print to the Clipboard, current Windows printer, or to a file. Users can also designate printers and fonts, and view how the screen will look when printed.
Send screen	Activates the user's installed mail or messaging system. Users can send a screen without exiting the Rumba window.
Print Area Setup	Sets up batch files to print multiple host screens to a PC printer.
Edit Clipboard	Supports data cutting, copying, and pasting activities.
Paste Area Setup	Sets up host screens to receive data pasted from the Windows Clipboard. Using this feature, users can paste data from a third-party application into multiple screens of a host application.
Title Bar	Customizes the title bar text for the Rumba window.
Configurable toolbars	Customizes the Rumba toolbar. You can create and modify toolbars to suit user needs and preferences. **Note:** This is currently available only in the Classic interface.
Watermark	Displays an image in the background of the current Rumba window.
Status bar	Toggles the 3D status bar on and off.
Customized display	Customizes the look of the Rumba window: fonts, colors, cursor settings, and more.
Keyboard mapping	Maps special terminal keys to the PC keyboard. Users can also select different keyboards and international keyboard options.

Tool	Description
Capture Screen	Captures the current Rumba screen in a specified `.txt` file (or `capture.txt` if no file is specified).
Macros	Records a series of actions, such as keystrokes and commands. Once users record a macro, they can play it back to automate routine tasks. Users can assign macros to a Hotspot (see below), a Rumba toolbar button, or a key combination on the keyboard.
Hotspots	Makes any non-protected text on a host screen into a Hotspot and associates that text with a macro. When the user clicks on the Hotspot, the macro runs.
Customized menus	Customizes Rumba menus. For example, users can create new menu items that start non-Rumba applications. **Note:** This is currently available only in the Classic interface.
Online help	Provides detailed instructions on how to use Rumba. Users can browse the table of contents for information, or search for a specific term.

Configuration and maintenance

The most common way to connect Rumba to a host is by selecting and configuring an interface from the Rumba display or printer. However, if you are connecting to an IBM mainframe or AS/400, there are additional tools available to you. The following table provides descriptions for the other available connection tools:

Application	Description
APPC Configuration Utility	Provides advanced configuration options for most mainframe and AS/400 interfaces. Allows the configuration of the following APPC items: Remote and local logical units (LUs), Modes, transaction programs (TPs), CPI-C side information, and conversation security for TPs.
AS/400 Communications utility	Used primarily to configure connections for Shared Folders, Data Queues, Submit Remote Command, and AS/400 file transfers.

Use the following diagnostic tools (available with all products) to monitor and troubleshoot Rumba connections on your network.

Feature	Description
Trace Console	Traces and records communication sessions. Specifically: • Data streams from the mainframe display and printer, AS/400 display and printer, UNIX display and HP display. • SNA communications. • Rumba display APIs, such as DDE, EHLLAPI, WinHLLAPI, EHNAPPC.

Feature	Description
	• Host system APIs, such as Data Queues, Shared Folders, AS/400 file transfers, and Submit Remote Command.
Rumba Communication Monitor	Displays an in-depth look at the connections of a workstation. **Note:** Not available for Rumba UNIX-HP Edition.

Developer tools

The AS/400 client applications, such as File Transfer, have APIs associated with them. Additionally, the Submit Remote Command is available as ActiveX controls.

Rumba consists of a family of ActiveX controls and Web-to-Host consists of both ActiveX controls and JavaBeans. You may use these ActiveX controls and JavaBeans to create new applications using languages such as Visual Basic, C++, and Java. Information and samples for using these components can be found on the Micro Focus Web site.

Planning installation and deployment

This section describes how to plan a Rumba installation.

Windows terminology

Before discussing Rumba setup, it is a good idea to understand some basic terminology and features of the Windows policy-based management environment.

Roaming	Windows allows you to move from computer to computer, yet maintain your specific desktop profile. Your login information enables this particular feature because your profile information is located on a server.
Windows Installer file types	The Windows Installer replaces previous installation methods, and enables cleaner installs and uninstalls, as well as repairs.
	The patch install package file (`.msp`) allows you to apply hot fixes to your Rumba product.
	The transform file (`.mst`) allows you to customize your Rumba installations. Before using the transform file, ensure you have the latest version of Windows Installer installed. For more information, see *Windows Installer service* and *How to customize Setup*.
Publishing and assigning with Intellimirror	Administrators can publish Rumba to users or assign Rumba to users or computers.
	Administrators assign an application through Group Policy Objects. Assigned applications have installation persistence. When a user selects the program for the first time, it is installed. If a user uninstalls an assigned application, it will re-advertise immediately.
	Administrators publish an application by making it available to users who might want the application. With published applications, each user decides whether or not to install the application from the Control Panel's Add/Remove Programs utility.
	Note: You can only publish to users, not computers.
Active Directory	Administrators deploy applications in Group Policy Objects (GPOs) that are associated with Active Directory containers such as sites, domains, and organizational units (OUs).

Windows Installer service

Rumba requires the Windows Installer service to install any Rumba feature on a Windows operating systems. Using the Installer service requires less of your management time and allows you to design seamless deployment packages for Rumba.

The Windows Installer service manages applications at the component level. It maintains reference counts in the form of a client list of product codes (rather than integers), so it can identify clients of the installed resource to synchronize counts. This allows clean and complete installs and uninstalls of Rumba, whether simple or sophisticated.

Note: A Rumba installation detects a previous Rumba 7.x or higher installation and uninstalls it before installing the current version.

Supported operating systems

Rumba operates on PCs with the following operating systems, applications, and environments:

- Windows 7, 8.1, and 10
- Windows Terminal Server (WTS)
- Citrix XenApp (formerly Presentation Server)
- Microsoft Application Virtualization (App-V)

Software requirements

The following software is required:

- Windows Installer 3.1 or later
- Microsoft .NET Framework 4.5.1
- Microsoft Visual C++ 2013 SP1 Redistributable Package (x86)

This software is available from the Microsoft Web site.

In addition, the Microsoft Office .NET Interoperability Assemblies (PIA) is required. The PIA enables .NET managed code to interact with a Microsoft Office application's COM-based object model. In most cases, the PIA is installed during Office installation. However, sometime it is not. If you need to add the PIA:

1. In **Control Panel**, select **Programs and Features**.
2. In the opened window, select the Microsoft Office installation and click **Change.**
3. Select **Add or Remove Features**.
4. Click **Continue.**
5. On the **Installation Options** page, click the plus sign (**+**) to expand the application for which you want to install the PIA.
6. Click the drop-down arrow next to **.NET Programmability Support**, then select **Run from My Computer**.

Hardware requirements

The following hardware is required:

- 1 GB RAM
- Disk space (minimum):

    - 32-bit: 850 MB
    - 64-bit: 2 GB

Things to consider before installation

When you install Rumba, Setup prompts you for standard information. Plan your installation based on the following questions (not necessarily in this order):

- Is this a first-time installation, or is a previous version of Rumba installed?
- Will you need to uninstall a Rumba version?
- What type of installation will you select: Complete or Custom?
- To which hosts will you connect: AS/400, Mainframe, HP, or UNIX?

- Will you be using Intellimirror for your distributed installs?
- Will you be using SMS for your distributed installs?
- Will you be using SMS combined with Intellimirror?
- Will you customize the installation with your own transforms?
- Will you be installing on Windows Terminal Server (Citrix)? After installing Rumba on Windows Terminal Server, follow its vendor instructions on how to publish Rumba applications to your users.

Deployment strategies

As an administrator, how you deploy Rumba depends largely on the change and configuration management functions of your networks, whether simple or complex. *Installing Rumba* contains detailed deployment and customization information for your enterprise-wide Rumba installations.

Server path	When you are doing any kind of administrative install, you need to know the Universal Naming Convention (UNC) path of the distribution point.
Complete or Custom installation	These installation options allow you to control which Rumba features are installed. The features available depend on your Rumba edition. • **Complete**: Installs all Rumba features, without the opportunity to add or remove individual features. • **Custom**: Gives you the opportunity to add or remove individual features.

Migration issues and directory locations

User-defined files and configuration files

User-defined files and configuration files reside in different locations in Rumba versions before 7.0. To support Windows roaming, we classify each of these files according to their scope and function. For example:

- Per user: Each user has their own version of a file and this file can roam with the user.
- Machine-specific: This file only relates to the machine that the product is currently installed on and cannot roam with the user.

These locations are configurable.

For detailed information on Rumba file locations on supported platforms, see *File locations*.

Rumba installations before Rumba 7.0

Rumba detects the existence of previous 32-bit Rumba and OnWeb Web-to-Host products. Installation does not continue until these are removed. For large SMS deployments, it is strongly advised that you create separate SMS packages for removing previous versions of the product and installing the new version.

Microsoft Host Integration Server (HIS) 2000 Server and Client

To configure a workstation to communicate with a Microsoft HIS 2000 server (formerly called Microsoft SNA server), you need to have two pieces of software on the workstation: the Microsoft HIS2000/SNA client, and the Rumba interface for the Microsoft SNA/HIS 2000 server. If you do not have the software for the Microsoft SNA client, you can download the client from Microsoft.

Changes to Rumba components

The Rumba Optimized Server ODBC driver has been moved into the AS/400 File Transfer subsystem.

The TP Director was included in previous versions of Rumba to map Local LU 6.2 information for APPC and/or CPIC Transaction Programs to a specific User ID. The TP Director is no longer installed as part of Rumba, but can be installed separately using the `TPDirector.msi` file if required.

Similarly, the Script Editor is no longer installed as part of Rumba but can be installed separately.

Installing Rumba

This section is written for IT professionals and system administrators who want to customize, install and deploy Rumba software.

The section contains instructions for installing Rumba on Windows with Microsoft Installer technology. Additionally, it contains information about:

- Installing, modifying, repairing, and uninstalling Rumba software
- Customizing setup
- Microsoft installer command line arguments
- Transforms

The installation process

Installation can be as simple as following the installation wizard on a single desktop, or as flexible as "push" and "pull" delivery options with transforms.

To modify the basic installer process, you must have a competent grasp of how to create transforms, and an authoring tool to do so, and/or are familiar with command line options.

You should already know what kind of installation you need and should already have gathered the information required. If not, see *Things to consider before installation*.

Default path of user profile

Regardless of the installation type, the default path for the user profile and configuration folders is:

`C:\Users\<UserLogon Name>\AppData\Local\Micro Focus\Rumba`

If Web-to-Host is installed on the same computer, you cannot change the destination or user profile folder location as many files are shared by both products.

Using the Setup wizard

The first, and simplest, installation method is to use the Rumba Setup wizard. During this installation, you are asked to select a **Complete** or **Custom** installation. **Complete** is the default.

The Setup wizard:

- Transparently installs the Windows Installer if it is not present on your system.
- Transparently installs the RSS Client (for increased security) if it is not present on your system.
- Checks that there is sufficient disk space on your destination drive.
- Allows you to choose the product language (English, French, or German).
- Advises you of any Rumba prerequisites that are not currently installed on your computer.

Standalone installation

1. Double-click the appropriate Rumba `.exe` file.
2. Follow the instructions in the Setup wizard.
3. Choose a directory for Rumba to be installed to. Micro Focus recommend that you use the default directory. You can also choose the location for application data files.
4. Select the language version you want to install. The default is the operating system language. If the operating system language is not English, French, or German, English is used.

5. Select the installation type: **Complete** or **Custom**.

- **Complete**: Installs all Rumba components. This option requires the most disk space.
- **Custom**: This option has the same default items selected as the **Complete** option, but allows you to add or remove individual components. The **Feature Description** panel to the right of the component selection box shows the amount of disk space required for each feature.

Setup creates a Micro Focus Rumba directory in your Windows **Start** menu from which you can access all the Rumba products and related tools that you have installed.

Administrative installation

To install Rumba as an administrative installation, use the following command:

```
<absolute_path>\<Rumba_install_file>.exe /a [/v"[TARGETDIR=<path>] [/q[n|b|r|
f]]"]
```

Using the command line

An alternative installation method is to use the command line. This method provides you with various options.

Syntax

```
<absolute_path>\<Rumba_install_file>.exe <exeFlag> <INSTALLFLAG> /v"<msiFlag>
MRILANG=<Rumba_UI_language> INSTALLDIR=<Rumba_installation_folder>
PRIVATEDIR=<user_profile_folder>
INTERFACEPRIVATEDIR=<interface_configurations_folder> CUSTOMIZEDDIRS=1"
```

Parameters

Parameter	Description	Required
`<absolute_path>`	Absolute path to the Rumba installation file.	Yes.
`<Rumba_install_file>.exe`	The type of installation file in the form of: `Rumba_<package>.exe` for a licensed installation. `Rumba_<package>_eval.exe` for an evaluation installation. where `<package>` is one of: UNIX Office Mainframe Citrix AS400	Yes.
`<exeFlag>`	Specifies either a standard or silent installation or uninstall: • Specify nothing for standard. • Specify `/s` for silent.	Only for silent.
`<INSTALLFLAG>`	Specifies whether to install or uninstall Rumba: • Specify nothing to install.	Only for uninstall.

Parameter	Description	Required
	• Specify /X to uninstall.	

<msiFlag>	Specifies the installation mode. Either or both of silent and with a log file.
	
	Silent:
	
	/q {n,b,r,f}
	
	This sets the user interface level:
	
	  n - No UI.
	  b - Basic UI.
	  r - Reduced UI.
	  f - Full UI (default).
	
	Log file:
	
	/l[i|w|e|a|r|u|c|m|o|p|v|x|+|!|*] *<log_file>*
	
	  i - Status messages.
	  w - Non-fatal warnings.
	  e - All error messages.
	  a - Start up actions.
	  r - Action-specific records.
	  u - User requests.
	  c - Initial UI parameters.
	  m - Out-of-memory or fatal exit information.
	  o - Out-of-disk-space messages.
	  p - Terminal properties.
	  v - Verbose output.
	  x - Extra debugging information.
	  + - Append to existing log file.
	  ! - Flush each line to the log file.
	  * - Log all information, except for v and x options.
	  /log *<log_file>* - Equivalent of /l* *<log_file>*.
		No. For either silent mode or to create a log file.

MRILANG	Specifies the installation language:
	
	  English: MRI2924
	  French: MRI2928
	  German: MRI2929
		No. If not, specified, the operating system language is used.

INSTALLDIR	Specifies the Rumba installation folder.	No.
PRIVATEDIR	Specifies a custom user profile folder. Default is %LocalAppData%\Micro Focus\RUMBA\.	No.

Parameter	Description	Required
INTERFACEPRIVATEDIR	Specifies a custom interface folder. Default is %AppData%\Micro Focus\RUMBA\.	No.
CUSTOMIZEDDIRS	To be specified if PRIVATEDIR and INTERFACEPRIVATEDIR are specified. Value must be 1.	Yes, if PRIVATEDIR and INTERFACEPRIVATEDIR are specified.

Examples

Standard installation

```
C:\Users\Admin\Desktop\Rumba_Office.exe
```

Standard uninstall

```
C:\Users\Admin\Desktop\Rumba_Office.exe /X
```

Silent installation with non-default Rumba INSTALLDIR

```
C:\Users\Admin\Desktop\Rumba_Office.exe /s /v"/qn INSTALLDIR=c:
\MyRumbaInstallDir"
```

Silent German installation with non-default Rumba INSTALLDIR, custom user and interface folders

```
Rumba_Office.exe /s /v"/qn MRILANG=MRI2929 INSTALLDIR=c:\MyRumbaInstallDir
PRIVATEDIR=c:\MyPrivateDir INTERFACEPRIVATEDIR=c:\MyInterfacePrivateDir
CUSTOMIZEDDIRS=1"
```

Silent French installation with non-default Rumba INSTALLDIR, custom user folder and verbose log file

```
Rumba_Office.exe /s /v"/qn /l*v c:\instLog.log MRILANG=MRI2928 INSTALLDIR=c:
\MyRumbaInstallDir PRIVATEDIR=c:\MyPrivateDir INTERFACEPRIVATEDIR=c:
\MyInterfacePrivateDir CUSTOMIZEDDIRS=1"
```

Installing Rumba prerequisites silently

Silent installation without a log file

```
RumbaPrerequisites.exe /s /v"/qn"
```

Silent installation with a log file

```
RumbaPrerequisites.exe /s /v"/qn /l*v <log_file>"
```

Example

```
RumbaPrerequisites.exe /s /v"/qn /l*v c:\InstallationLogFiles
\RumbaPrerequisitesMsiLog.log"
```

Rolling back an installation

When you are installing Rumba in any configuration using the Setup wizard, the installation process can completely roll back. That is, it undoes every operation that it performs during an installation, modification, repair, or removal. This is particularly helpful if you encounter a fatal error during an installer session. The

computer can return to the precise state it was in before the installer session. This includes restoration of deleted or overwritten files, registry keys, and other resources.

✎ **Note:** You cannot roll back a completed Rumba installation. You must uninstall the program to remove it.

Roll-back disk space requirements

The disk space for installing a specific feature can be twice the size of the net disk space consumption. Files that are deleted or overwritten during the course of an installation or removal are temporarily saved to a backup location so that they can be restored if necessary. After a successful installation, or installer operation, all temporary backup files are deleted.

Modifying Rumba

To modify a Rumba installation:

1. Select **Control Panel** > **Add/Remove Programs** > **Change**.

   The Rumba Setup wizard starts.
2. Click **Next**.

   The **Program Maintenance** window appears.
3. Click **Modify**.

   The **Custom Setup** window appears.
4. Make the changes you want.

Repairing Rumba

To repair a Rumba installation:

1. Select **Control Panel** > **Add/Remove Programs** > **Change**.
2. Click **Next**.

   The **Program Maintenance** window appears.
3. Click **Repair**.
4. Follow the instructions in the wizard.

The Repair option can:

- Determine whether the requested components and feature are installed. If a component or feature are not installed, the installer checks for missing files and installs them.
- Check that all components in the requested feature are properly installed. The installer verifies the keypath of a given component to determine if that component is broken. If the keypath resource is missing, the installer performs an on-demand repair.

Removing Rumba

To remove a Rumba installation, select **Control Panel** > **Add/Remove Programs** > **Change**.

Ensure that the Rumba installation folder (normally `C:\Program Files\Micro Focus\Rumba`) was deleted during removal. If not, it is safe to delete it manually.

✎ **Notes:**

- Web-to-Host uses some of the same files as Rumba. If you also have Web-to-Host installed on your computer, the Rumba directory is not removed and should not be deleted manually.

- If you have other Micro Focus products installed on your computer, RSS Client files are not removed during the uninstall process. Removing them could cause another Micro Focus product to stop functioning.

Customizing Setup

This section contains information to help you customize your Rumba installation for individual, group, and enterprise-wide distributions:

The **Custom** option of the Rumba Setup wizard allows you to select the features you want to install. The wizard is self-explanatory and easy to understand.

You can either create transforms (`.mst` files) to modify the Rumba installation or use Windows Installer command line arguments.

For information about deploying Rumba information with SCCM, see *Distributing Rumba with SCCM*.

There are several ways to specify custom changes to the standard operations and behavior of the Setup program. Each of these methods affects a different aspect of the installation process. By combining them, you can effectively distribute Rumba to make full use of Intellimirror features as well as SCCM management capabilities.

Using transforms

A transform modifies the base installation package file at installation time. The transform remains cached on the computer and is applied to the base installation package whenever the Windows Installer service performs a configuration change (including repair) on Rumba.

Transforms increase the flexibility of Rumba installations by applying changes to the installation database without altering the original database. For example, in organizations where different departments require different installations of Rumba, Rumba's base package can be made available to everyone at one administrative installation point, with the appropriate customizing transforms separately distributed to each group of users.

Transforms cannot be applied to an already installed application, only at initial installation.

You can perform a run-from-source (RFS) installation using transforms. This means that only a small subset of files is installed on the user workstation. The bulk of the product is run directly from a central location, such as a shared network drive. This approach minimizes local computer disk requirements but increases network load.

You should be proficient at using a transform authoring tool to make full use of a Rumba distribution.

⚠ **Caution:** We recommend that you use transforms carefully because they expose a great deal of functionality and detail. Because each installation customized by a transform is unique, Micro Focus might not be able to support customers using such installations.

Before you run an installation with a transform, you must:

- Perform an administrative installation of Rumba.
- Create the transform.

This section describes how to create a transform using InstallShield and Orca.

Performing an administrative installation

1. Select the location for the installation (RFS) directory. This is usually a shared directory on a network drive.
2. To protect the installation files, either specify read-only access to the directory for all users, or individually change the file attributes of all the installed files to read-only.

3. To create the administrative installation, use the following command:

```
<absolute_path>\<Rumba_install_file>.exe /a [/v"[TARGETDIR=<path>] [/q[n|b|
r|f]]"]
```

and specify the RFS directory as destination directory.

Creating a transform using InstallShield

1. Open InstallShield, then select **File** > **New**.

The **New Project** dialog box appears.

2. Click the **Windows Installer** tab.

3. Select **Transform**, type a name in the **Project Name** field, such as **MyTransform**, then click **Browse** to find to select the location for the transform.

4. Click**OK**.

The **Open Transform Wizard** wizard starts.

5. Click **Next**.

The **Transform Information** page appears.

6. Click **Browse** next to the **Base MSI File Name** field and select the Rumba `.msi` file.

> **Note:** From Rumba 9.5, only Rumba `.exe` files are provided. To extract the `.msi` file from the `.exe` install package, run the following command:
>
> ```
> <absolute_path>\<Rumba_install_file>.exe /a [/v"[TARGETDIR=<path>] [/
> q[n|b|r|f]]"]
> ```
>
> for example:
>
> ```
> C:\Users\Administrator\Desktop\Rumba_Office.exe /a /v"TARGETDIR=C:
> \ExtractedRumbaFolder /qn"
> ```
>
> The `Rumba.msi` file is extracted to `C:\ExtractedRumbaFolder`.

7. Click **Next**.

The **Additional Transforms** page appears.

8. Click **Next**.

The **Create a Response Transform** page appears.

9. Click **Finish**. **MyTransform (Rumba.msi)** is created.

10. In InstallSheild, click the **Installation Designer** tab.

11. Navigate to the **Organization** > **Setup Design** view.

12. Create the transform. For example:

a) Right-click the features tree in the **Setup Design** view and select **New Feature** from the pop-up menu.

b) Call the new feature **Transform**.

c) Right-click the **Transform** feature in the **Setup Design** view and select **New Component** from the pop-up menu.

The created component is associated with the **Transform** feature by default.

> **Note:** If you create the component from the **Components** view, you must associate it to the **Transform** feature manually. To do this:
>
> **1.** Right-click the feature in the **Setup Design** view and select **Associate Components** from the pop-up menu.
>
> **2.** Select the created component.

       3. Click **OK**.

  d) Create a new registry key value for the transform to add to the `Rumba.msi` installation.

13. Save the transform.

Creating a transform using Orca

1. Start Orca.

2. Click **File** > **Open** and select the Rumba `.msi` file.

> **Note:** From Rumba 9.5, only Rumba `.exe` files are provided. To extract the `.msi` file from the `.exe` install package, run the following command:
>
> ```
> <absolute_path>\<Rumba_install_file>.exe /a [/v"[TARGETDIR=<path>] [/
> q[n|b|r|f]]"]
> ```
>
> for example:
>
> ```
> C:\Users\Administrator\Desktop\Rumba_Office.exe /a /v"TARGETDIR=C:
> \ExtractedRumbaFolder /qn"
> ```
>
> The `Rumba.msi` file is extracted to `C:\ExtractedRumbaFolder`.

3. Select **Transform** > **New Transform**.

4. Navigate to the table or tables you want to edit, then modify values or create new rows as appropriate.

5. Select **Transform** > **Generate Transform**.

6. In the **Save Transform As** dialog box, type a name for the transform and click **Save**.

The **Rumba.msi transformed by** window appears.

Performing a workstation installation

1. Run the following command from the RFS directory:

```
<absolute_path>\<Rumba_install_file>.exe /v"TRANSFORMS=<transform>.mst"
```

For example:

```
C:\Users\Administrator\Desktop\Rumba_Office.exe /v"TRANSFORMS=\"C:\Users
\Administrator\Desktop\rumbaTransform.Mst\""
```

Rumba installs.

2. Changes made by the transform should appear on the machine after installation. For example, if a registry value was added, it should appear in the registry.

> **Note:** Install on Demand is not supported when performing an RFS installation.

Windows Installer command line arguments

The following information is available in more detail from the Windows Installer SDK. The most common options and parameters are presented here to enable you to easily organize and customize your deployment options.

Option	Parameters	Definition
/i	{package|ProductCode}	Installs or configures a product.
/a	package	Administrative installation option.
/f	[p][o][e][d][c][a][u][m][s][v]{package|ProductCode}	Repairs a product. This option ignores any property values entered on the command line.

Option	Parameters	Definition
		p Reinstall only if file is missing.
		o Reinstall if file is missing or if an older version is installed.
		e Reinstall if file is missing or an equal or older version is installed.
		d Reinstall if file is missing or a different version is installed.
		c Reinstall if file is missing or the stored checksum does not match the calculated value.
		a Force all files to be reinstalled.
		u Rewrite all required user-specific registry entries.
		m Rewrite all required computer-specific registry entries.
		s Overwrite all existing shortcuts.
		v Run from source and recache the local package.
/x	{package|ProductCode}	**x** Uninstalls a product.
/L	[i][w][e][a][r][u][c][m][p][v][+][!]logfile	Specifies the path to the log file. The following flags are:
		i Status messages
		w Non-fatal warnings
		e All error messages
		a Startup of actions
		r Action-specific records
		u User requests
		c Initial user interface parameters
		m Out-of-memory
		p Terminal properties
		v Verbose output
/p	*PatchPackage*	Applies a patch. To apply a patch to an installed administrative image you must combine options as follows:
		/p PatchPackage
		/a package
/q	{n | b | r | f}	In conjunction with /i (install) or /x (uninstall), sets the user interface level.
		qn No user interface.
		qb Basic user interface.

Option	Parameters		Definition
		qr	Reduced user interface with modal dialog box displayed at the end of installation.
		qf	Full user interface with modal dialog box displayed at end of installation.
		qn+	No user interface except for modal dialog box displayed at end of installation.
		qb+	Basic user interface with modal dialog box displayed at end of installation.
		qb-	Basic user interface with no modal dialog boxes.
			Note: /qb+- is not a supported user interface level. The modal dialog box does not display if the user cancels the installation.

Distributing Rumba with SCCM

The mechanics of Microsoft System Center Configuration Manager (SCCM) administration are beyond the scope of this document. It is assumed that you are familiar with the standard operations of the SCCM family.

Note: SCCM was previously known as Systems Management Server (SMS).

Uninstalling previous versions of Rumba

Before you can deploy this version of Rumba, you must uninstall any previous versions of Rumba that are currently installed.

The easiest way to uninstall Rumba varies according to the version of SCCM you have installed. In some versions of SCCM the easiest way to uninstall previous versions of Rumba is to include a response file (response.ini) or definition file (.pdf) in a command line as part of an uninstallation deployment package. In this case, if you do not have the previous-version documentation and the .pdf or response file (or files) of your previous Rumba installation, you must create a new response file.

In other versions of SCCM, the easiest method is to create a task sequence to run the uninstallation. Refer to the documentation for your SCCM version.

Preparing client computers for Windows Installer

After you have uninstalled a previous Rumba version, ensure that Windows Installer is resident on client computers before installing the current version of Rumba. If Windows Installer is not installed, you must install it.

Installing Windows Installer requires administrative rights on client computers. To install Windows Installer, use the installer setup program, InstMSI.exe.

Note: Command line options are case sensitive.

You can also create a program dependency in an SCCM deployment package which installs the Microsoft Installer service immediately prior to a Rumba installation, on an on-demand basis.

You can download the Microsoft Installer SDK from: *http://www.microfocus.com/docs/links.asp?nx=platsdk*

Creating a Rumba deployment package

Before creating a Rumba deployment package, consider:

- Source and distribution points.
- Will this be an on-demand installation?
- How your collections should be configured.
- Whether clients have the correct rights, or whether to configure package access accounts.
- Will you configure the user interface to be used by the clients?

You can use the SCCM New Package wizard to create a deployment package.

Identifying a package source directory

The directory where you install an administrative installation becomes the package source directory. Any additional files, such as Rumba profiles or transforms, must also be copied to this location. The results of the administrative installation, plus any modifications you make, are propagated to the specified distribution point servers. SCCM clients can then install the package from the distribution point servers.

Advanced deployment issues

- System restarts

  Because Rumba determines if a computer must be restarted at installation time, the safest option is to configure SCCM to perform the restart.
- Program removal

  Rumba is registered in **Control Panel** > **Add/Remove Programs**. This allows SCCM to remove the application automatically.

For more informarion, refer to *http://www.microfocus.com/docs/links.asp?nx=sms*

Distributing Rumba with the MSS

You can use the Micro Focus Management and Security Server (MSS) to distribute Rumba `.MSI` installation package files.

System administrators can configure all aspects of a Rumba desktop environment, including:

- Session profiles
- Desktop profiles
- Keyboard mapping
- Macros
- Scripts

  Note: Centralized management must be enabled in the **Rumba Options** on a user's local machine.

To use the MSS to deploy a package file:

1. Create the `.MSI` package file.
2. Use the MSS Package Manager to upload the file.
3. Use the MSS Access Mapper to associate the file with specific users or user groups.

When a user next starts Rumba, the package file is automatically downloaded to their local machine.

Rumba file locations

Preinstalled files

`default.mnc`	`C:\ProgramData\Micro Focus\Rumba`
`SmpleAs4.rmc`	`C:\Program Files\Micro Focus\Rumba\As400\Macro`
`SmpleHP.rmc`	`C:\Program Files\Micro Focus\Rumba\Hp\Macro`
`SmpleTSO.rmc`	`C:\Program Files\Micro Focus\Rumba\Mframe\Macro`
`SmpleVM.rmc`	`C:\Program Files\Micro Focus\Rumba\Mframe\Macro`
`SmpleVT.rmc`	`C:\Program Files\Micro Focus\Rumba\Vt\Macro`

Run-created files

`default.mnc`	`C:\Users\<user>\AppData\Roaming\Micro Focus\Rumba`
`config.ehn`	`C:\Users\<user>\AppData\Roaming\Micro Focus\Rumba`
`RumbaTRC.CFG`	`C:\Users\<user>\AppData\Roaming\Micro Focus\Rumba`

Macros

`.rmc, .js, .vb`	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\AS400\Macro`
	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\HP\Macro`
	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\MFrame\Macro`
	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\VT\Macro`

User toolbars

The user toolbars that were used in previous releases are now saved within individual session profiles and cannot be shared.

Default session profiles

`.prf`	`C:\Users\<user>\AppData\Roaming\Micro Focus\Rumba\Defaults`

User session profiles

`.rsda`	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\as400`
`.rsdh`	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\H`
`.rsdm`	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\MFrame`
`.rspa`	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\as400`
`.rspm`	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\MFrame`
`.rsdu`	`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\VT`

Rumba+ project files

Rumba+ project directory	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\Plus` `\Projects`
`.rdar`	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\Plus` `\Rules`

Desktop profiles

Default path:

`.rdps`	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba`

Print preview

Default path:

`.rpf`	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\AS400`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\HP`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\MFrame`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\VT`

Keyboard map

`.map`	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba`

Transferred files

`?`	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\AS400`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\HP`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\MFrame`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\VT`

Hotspots library

`.hsp`	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba` `\AS400\Macro`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\HP\Macro`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\MFrame` `\Macro`
	`C:\Users\`*`<user>`*`\AppData\Local\Micro Focus\Rumba\VT\Macro`

Help files

`.chm`	**English**	`C:\Program Files\Micro Focus\Rumba\Help\MRI2924`
		and
		`C:\Program Files\Micro Focus\Rumba\HelpRibbon`
		`\MRI2924`

French	`C:\Program Files\Micro Focus\Rumba\Help\MRI2928`
	and
	`C:\Program Files\Micro Focus\Rumba\HelpRibbon\MRI2928`
German	`C:\Program Files\Micro Focus\Rumba\Help\MRI2929`
	and
	`C:\Program Files\Micro Focus\Rumba\HelpRibbon\MRI2929`

Font mapping file

`.lcs`	`C:\ProgramData\Micro Focus\Rumba`

Using Rumba

This section describes how to use some of the main features of Rumba.

Starting a session from the command line

Syntax

```
RumbaPage.exe <path>\<session_desktop_profile> /Display:MFrame /
Host:<IP_or_host_name> /Port:<port_number> /forcenewinstance
```

where:

<path>	is the absolute path to a saved session or desktop profile. Only required when specifying a profile.
<session_desktop_profile>	The name of a saved session or desktop profile. Optional.
Display	specifies the type of session. Currently supports only `MFrame`. Optional.
<IP_or_host_name>	specifies the IP address or name of the host to connect to. Can only be used with `/Display:MFrame` for which it is required.
<port_number>	the port number to connect to. Can only be used with `/Display:MFrame` for which it is required.
forcenewinstance	forces a new Rumba instance to start. Optional.

Notes:

- If there is a syntax error or if a required parameter is not supplied, Rumba starts as if no parameters were supplied.
- Session profiles and desktop profiles cannot be set to Auto-Connect. If such a profile is specified, an error occurs and all parameters are ignored.
- If a desktop profile is specified, `Display`, `Host`, and `Port` must not be used. Otherwise, an error occurs and all parameters are ignored.

Examples

```
RumbaPage.exe /Display:MFrame /Host:<IP_or_host_name> /Port:<port_number>
```

Starts a default session and attempts to connect to the specified host and port.

```
RumbaPage.exe <path>\<session_profile> /Display:MFrame /
Host:<IP_or_host_name> /Port:<port_number>
```

Opens the specified session profile (including all keyboard mapping and color settings) and attempts to connect to the specified host and port.

```
RumbaPage.exe <path>\<desktop_profile>
```

Opens the specified desktop profile.

Rumba folders

You can access Rumba programs from **Start** > **Programs** > **Micro Focus Rumba** . This includes the **Rumba Tools** folder. Exactly what programs are available depends on which components you installed.

The **Rumba Tools** folder contains tools for system administrators, such as:

- Micro Focus Rumba 9.5 APPC Configuration
- Micro Focus Rumba 9.5 AS400 Communications
- Micro Focus Rumba 9.5 Comunication Monitor
- Micro Focus Rumba 9.5 Engine
- Micro Focus Rumba 9.5 Submit Remote Command
- Micro Focus Rumba 9.5 Trace

Using Rumba menus, toolbars, and the status bar

You can use Rumba commands, from either the classic or the ribbon interface, to perform most of your host tasks. The status bar provides you with information about Rumba and the state of the host.

You can streamline your work environment by customizing Rumba menus and the toolbar. For example, you can add icons to launch external applications, or remove menu commands and toolbar buttons that you do not use. Currently, this facility is only available for the classic interface.

Enabling and disabling global commands

Global commands are all Rumba commands which are not host-specific. For example, **Create New Session** and **Open Screen Designer**.

Note: Before you enable or disable global commands on menus, ensure that you close all open host session windows.

To enable and disable global commands:

1. Right-click the toolbar and select **Customize** from the pop-up menu.

   The **Global Commands Availability** window appears.
2. To enable a command, check the command. To disable a command, uncheck the command.
3. When you have finished, click **OK.**
4. In the **Customize** window, click **Close.**

Notes:

- If you disable the **Command Availability** command, the **Customize** command is also disabled. This means that you can no longer enable and disable global commands. To re-enable the **Command Availability** command:

  1. Edit the following file with an ASCII editor, such as Notepad:

     ```
     %LOCALAPPDATA%\Micro Focus\Rumba\Rumba.config
     ```
  2. Find the following element:

     ```
     <IsCommandsConfiguratorEnabled Type="Boolean">False</
     IsCommandsConfiguratorEnabled>
     ```
  3. Replace `False` with `True`.
  4. Save the file and exit.

- If you disable the **Save** or **Save As** commands, you are still offered the option to save a session profile when you close a host session.
- If you disable a command that is associated with a toggle button, such as **Rumba+ Mode**, the button is also disabled.

Customizing menus and toolbars

To customize commands available within a profile:

1. Open the Rumba display or printer session containing the menus that you want to change.
2. Select **View** > **Toolbars**. The **Customize** dialog box opens.
3. On the **Toolbars** page, create and rearrange custom toolbars with the commands you use most often.

   Use the **Commands** page to add commands to the standard toolbar or to change the availability of commands within the interface.

   📝 **Note:** If you make the **View** > **Toolbars** command unavailable, you cannot make further changes.

4. On the **Commands** page, select **Rearrange Commands** > **Modify Selection** > **New Button** to create new toolbar buttons of different types.

Toolbars can be docked, undocked, and floated. They can also be imported from and exported to files. Toolbar buttons can be chosen from several types.

Mainframe status bar

Status bar area	Function
Status field	Displays general Rumba status information.
Operator	Indicates the operator's application session owns the screen.
SSL	Indicates whether your session uses an SSL (Secure Sockets Layer) connection. This field applies only to TCP/IP connections.
APL	Indicates whether or not the host is in APL mode.
NUMFLD	Indicates whether or not the cursor is currently in a numeric field.
LU#	Shows the Logical Unit (LU) number.
OVR	Indicates whether Overstrike or Insert mode is chosen.
CAP	Indicates whether Caps Lock is on or off.
NUM	Indicates whether the Number Lock is on or off.
W	Indicates whether Word Wrap is on or off.
Cursor coordinates	Lists the row and column position of the cursor.
Clock	Displays the time.

AS/400 status bar

Status bar area	Function
Description area	• Connected: The status of the host connection. If the host is not connected, the description area is blank. • Toolbar button function: If the mouse cursor points to a toolbar button, the description area displays a description of the function performed by the toolbar button.

Status bar area	Function
	• Menu command function: If the context-sensitive help cursor points to a menu command, the description area displays a description of the function performed by the menu command.
SA	System available. When lit, this indicates that the host is operating and available to the PC.
MW	Message waiting. When lit, the host has one or more messages waiting for you.
KS	Keyboard shift. When lit, the keyboard is in shift mode.
IM	Insert mode. When lit, Insert is on. Characters can be inserted into a field without typing over the existing data.
II	Input inhibited. When lit, keyboard input is not being accepted by the AS/400. Try pressing the Error Reset key. If it is still highlighted, the system is processing your request.
KB	Keyboard buffer. This indicator is only displayed when you are waiting for the Input Inhibited indicator to go off so that keystrokes that are buffered can be processed. To clear keystroke buffering, press the Error Rest key.
Session Name	Displays the name of the server to which the Rumba session is connected.
Cursor coordinates	Lists the row and column position of the cursor.
Clock	Displays the time.

HP status bar

Status bar area	Function
Communications interface	Displays the communications interface you have selected, the server name (if applicable), and whether or not the session is connected to the host.
Terminal type	Shows the type of terminal this session is emulating, and other kinds of status information applicable to the terminal. Information appears if the terminal is in hold or wait conditions or if it is off-line.
Cursor coordinates	Lists the row and column position of the cursor.
Clock	Displays the time.

UNIX status bar

Status bar area	Function
Communications interface	Displays the communications interface you have selected, the server name (if applicable), and indicates whether the session is connected to the host.
Terminal type	Shows the type of VT this session is emulating, and other types of status information applicable to the terminal. Information appears if the terminal is in hold or wait conditions, is off-line, or if you are composing a character sequence.
Cursor coordinates	Lists the row and column position of the cursor. If you are using VT420 emulation, it also indicates the page number.
Numeric/ Application mode	Toggles the state of the terminal's numeric keypad between numeric mode and application mode (which is not the same as the PC NUMLOCK feature).
Clock	Displays the time.

Rumba session profiles

A session profile is a special configuration file used to store your settings for a specific session. You can customize the settings for a particular host connection to meet your particular needs and preferences, then save your settings in a session profile. When you open the profile, the Rumba session is configured with settings you designate.

Session profiles also provide a method for easily deploying standardized Rumba sessions to users. You can create session profiles for customized host display and printer sessions.

All session profiles are stored in a subfolder of the `AppData` folder. For example, an AS/400 profile would be stored in:

`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\AS400`

The subfolders are named `as400`, `HP`, `Mframe`, and `VT` (for UNIX host session profiles).

> **Note:** Each session profile opens as a separate tab in the Rumba display. This gives you the flexibility of being able to open several different profiles in one window. You can save multiple sessions in one configuration file as a desktop profile.

About session profiles

Session profiles contain specific information about the Rumba session including:

- Interface selection and configuration
- Display settings, including color, font, and status bar options
- Keyboard maps and alternative character set selection
- Macros and script information (if the Script Editor is installed)

When you change your Rumba environment, you can save those changes in a new or existing profile. Rumba stores session profile data in a binary format.

Session profiles are host-specific. The file extension for the different hosts are as follows:

Host	File extension
AS/400	`.rsda`
AS/400 Printer	`.rspa`
HP	`.rsdh`
Mainframe	`.rsdm`
Mainframe Printer	`.rspm`
UNIX	`.rsdu`

> **Note:** You can use the Profile Editor to create and edit Rumba profiles. The Profile Editor is an OLE control that you can load into your Visual Basic application. To install the editor, you must run a complete installation, or run a custom installation, then select **System Options** > **Administrator Options** > **Profile Editor Control**.

Supplemental and interface configuration files

There are also supplemental files that work together with session profiles. These supplemental files contain configuration or other important data unique to a feature. For example, the `hotspot.hsp` file is a supplemental file. It is also a Hotspot file, and it contains all the data required to run a series of predefined keystrokes.

Interface configuration files are a special class of supplemental files. They are saved in separate files with the `.cfg` extension. You may have more than one of these interface files if you use multiple host connections.

For details on the locations of these files, see *Installing Rumba*.

The supplemental files and their default extensions are:

File type	Extension
Interface configuration	`.cfg`
Keyboard map	`.map`
Macro script	`.rmc, .js, .vb`
Hotspot	`.hsp`
Menu configuration	`.mnc`
Script	`.csf, .vba`
Toolbar	`.rtbr`
Screen Designer project	`.rsdp`
Screen Designer definition	`.rdar`

Migrating session profiles from Rumba 6.x or 7.x

Session profiles that were created using Rumba versions 6.x or 7.x can be migrated to Rumba 9.5. To do this:

1. Open the profile in Rumba in 6.x or 7.x.
2. Click **Tools** > **Menu** , then click the **Preview** tab.
3. Ensure that the **Gray disabled items** option is selected.
4. Save the profile.
5. Upgrade to Rumba 9.5.

Rumba desktop profiles

A set of multiple session profiles can be saved as a single Rumba desktop profile. This allows you to provide a standard set of tabbed sessions to your users, minimizing the amount of configuration needed for each individual user.

By default, Rumba desktop profiles are stored with the file extension `.rdps` in the following folder:

`C:\Users\<user>\AppData\Local\Micro Focus\Rumba`

About desktop profiles

Desktop profiles contain specific information about a Rumba instance, including:

- The set of session profiles used and their layout
- Any customized tab names
- Any vertical and horizontal tab groups

A desktop profile does not contain any information about which Rumba interface is used. All desktop profiles can be opened in either the classic or the ribbon interface.

Session profile locations

A Rumba desktop profile contains references to the locations of all the session profiles that it includes. The locations of any session profiles that are in a different folder to the desktop profile are stored as absolute paths. If a desktop profile is to be deployed on machines other than the one on which it was saved, all of the relevant session profiles should be contained within the same folder as the desktop profile.

HP keyboard key codes

The HP keyboard sends a predefined escape sequence for each of its special keys. For example, editing keys and numeric keys. Most applications written for use with HP terminals expect to use these escape sequences. If your application is looking for a different escape sequence, you can program this sequence to a keyboard string, and map that keyboard string to a key sequence on your PC.

The tables in this section list the codes sent for each of the defined keyboard keys.

Keys mapped to active default settings

Legend	Key
Line modify	F1
Modify all	F2
Block mode	F3
Remote mode	F4
Type ahead	F5
Memory lock	F6
Display function	F7
Auto line feed	F8

F1 through F8 are local function keys that do not send codes, except when configured to do so. These keys are mapped to activate default softkeys. To see the key mappings:

1. Select **Options** > **SoftKeys** in the HP Display.
2. In the **SoftKeys Settings** dialog box, select **User Defined**.
3. Click **OK**.

Keys used to send 7-bit control characters

Control character mnemonic	ASCII value	Key pressed with Ctrl (all modes)
SOH	1	A
STX	2	B
ETX	3	C
EOT	4	D
ENQ	5	E
ACK	6	F
BEL	7	G

Control character mnemonic	ASCII value	Key pressed with Ctrl (all modes)
BS	8	H
HT	9	I
LF	10	J
VT	11	K
FF	12	L
CR	13	M
SO	14	N
SI	15	O
DLE	16	P
DC1	17	Q (see note below)
DC2	18	R
DC3	19	S (see note below)
DC4	20	T
NAK	21	U
SYN	22	V
ETB	23	W
CAN	24	X
EM	25	Y
SUB	26	Z

Note: 7-bit control codes sent only when XON/OFF support is off.

Codes sent by editing keys

Key	Code sent
Delete char	ESC P
Insert line	ESC L
Delete line	ESC M
Insert char	ESC Q
Prev char	ESC V
Next	ESC U
Clear line	ESC K
Clear display	ESC J

To view these codes, select **Tools** > **Show Host Keyboard** in the HP Display.

These codes are only sent when **Transmit Escape Sequences to Host** is enabled. To do this:.

1. Select **Options** > **Terminal**
2. On the **Advanced** tab, check **Transmit Escape Sequences to Host**.

3. Click **OK**.

Codes sent by arrow keys

Key	Cursor mode	Application mode
Up	ESC A	ESCA
Down	ESC B	ESCB
Right	ESC C	ESCC
Left	ESC D	ESCD

These codes are only sent when **Transmit Escape Sequences to Host** is enabled. To do this:.

1. Select **Options** > **Terminal**
2. On the **Advanced** tab, check **Transmit Escape Sequences to Host**.
3. Click **OK**.

Codes sent by numeric keypad keys

Key	Numeric	Application	
0	0	ESC Q	Ins
1	1	ESC F	End
2	2	ESC B	Down
3	3	ESC U	PgDn
4	4	ESC D	Left
5	5		
6	6	ESC C	Right
7	7	ESC h	Home
8	8	ESC A	Up
9	9	ESC V	Pg Up
-	(minus)		
,	(comma)		
.	(period)	ESC P	Del
Enter	CR LF		

> **Note:** In Character mode, the code goes directly to host. In Format/Block mode, the code goes to the host only when the user presses ENTER.

VT keyboard key codes

The VT keyboard sends a predefined escape sequence for each of its special keys. For example, function keys, editing keys, and numeric keys. Most applications written for VT terminals expect to see these escape sequences. If your application is looking for a different escape sequence, you can program this sequence to a keyboard string and map that keyboard string to a key sequence on your PC.

The tables in this section list the codes sent for each of the defined keyboard keys.

VT codes sent by top-row function keys

Table 1: VT codes set by top-row function keys

Legend	Key number	Code sent
Hold	F1	CSI 11~
Print	F2	CSI 12~
Set-Up	F3	CSI 13~
Session	F4	CSI 14~
Break	F5	CSI 15~
F6	F6	CSI 17~
F7	F7	CSI 18~
F8	F8	CSI 19~
F9	F9	CSI 20~
F10	F10	CSI 21~
F11 (ESC)	F11	CSI 23~
F12 (BS)	F12	CSI 24~
F13 (LF)	F13	CSI 25~
F14	F14	CSI 26~
Help	F15	CSI 28~
Do	F16	CSI 29~
F17	F17	CSI 31~
F18	F18	CSI 32~
F19	F19	CSI 33~
F20	F20	CSI 34~

F1 through F5 are local function keys that do not send codes, except when configured to send codes. To see the assigned local keys in the UNIX Display:

1. **Options** > **Terminal.**
2. Click the **Key Assignments** tab.

VT keys used to send 7-bit control characters

Control character mnemonic	ASCII Value	Key pressed with Ctrl (all modes)
SOH	1	A
STX	2	B
ETX	3	C
EOT	4	D
ENQ	5	E

Control character mnemonic	ASCII Value	Key pressed with Ctrl (all modes)
ACK	6	F
BEL	7	G
BS	8	H
HT	9	I
LF	10	J
VT	11	K
FF	12	L
CR	13	M
SO	14	N
SI	15	O
DLE	16	P
DC1	17	Q (see note below)
DC2	18	R
DC3	19	S (see note below)
DC4	20	T
NAK	21	U
SUN	22	V
ETB	23	W
CAN	24	X
EM	25	Y
SUB	26	Z

Note: 7-bit control codes sent only when XON/OFF support is off.

VT codes sent by editing keys

Key	Code sent
Find	CSI 1~
Insert here	CSI 2~
Remove	CSI 3~
Select	CSI 4~
Prev	CSI 5~
Next	CSI 6~

VT codes sent by arrow keys

Key	Cursor mode	Application mode
Up	CSI A	SS3A

Key	Cursor mode	Application mode
Down	CSI B	SS3B
Right	CSI C	SS3C
Left	CSI D	SS3D

VT codes sent by numeric keypad keys

Numeric keypad mode setting (DECNKM), ANSI mode.

Key	Numeric	Application
0	0	SS3 p
1	1	SS3 q
2	2	SS3 r
3	3	SS3 s
4	4	SS3 t
5	5	SS3 u
6	6	SS3 v
7	7	SS3 w
8	8	SS3 x
9	9	SS3 y
-	(minus)	SS3 m
,	(comma)	SS3 l
.	(period)	SS3 n
ENTER	CR or CR LF (see below)	SS3 M
PF1	SS3 P	SS3 P
PF2	SS3 Q	SS3 Q
PF3	SS3 R	SS3 R
PF4	SS3 S	SS3 S

In keypad numeric mode, ENTER sends the same codes as RETURN. You can use line feed/new line mode (LNM) to change the code sent by RETURN. When LNM is reset, pressing RETURN sends one control character (CR). When LNM is set, pressing RETURN sends two control characters (CR, LF).

Using Rumba+

This section describes how to use some of the main features of Rumba+.

You can use Rumba+ to increase end user productivity by customizing mainframe and AS/400 screens with modern controls. These controls include buttons, labels, images, check boxes, edit boxes, drop-down lists, radio buttons and Web links.

Using the Rumba Screen Designer

You use the Screen Designer to create or manage a screen customization project to be used in a Rumba+ session.

You can do this in one or both of the following ways:

Using individual screens

> You can identify each screen of the specific host session uniquely and customize each screen individually.

Using the Rule Manager

> You use the Rule Manager to create conditional actions (rules). At run-time, the data of each screen of a specific host session is passed through a set of rules to produce modern controls on the screen.

How it works

The figure below provides an overview of what you do to create and use a customization project:

Prepare
- Open a session and connect
- Using History, record the screens you want to customize

Design
- Open the Screen Designer, create a new project, then import the screen history
- Add static controls to the screens in the **Screen Designer** window
- For dynamic or conditional controls, create rules in the **Rule Manager** window

Use
- Generate a customization package file
- Select **Rumba+ > Rumba+ Mode** to activate **Rumba+ mode**
- In the session, select **Rumba+ > Settings** to associate the customization file with the session

Customization projects

A customization project comprises:

- A folder with the project name, for example, `..\MyProject`.
- A configuration file, `MyProject.rsdp`, used to store screen customization.
- Other related files.

When you open file the `MyProject.rsdp` file in the Screen Designer, you see the Rumba screens that you have customized with modern controls.

Customization projects provide a method for easily distributing Rumba+ customizations among developers. You can create customization projects for mainframe and AS/400 sessions.

By default, all customization projects are stored in a sub-folder of the `AppData` folder. For example:

`C:\Users\<user>\AppData\Local\Micro Focus\Rumba\Plus\Projects`

Deploying a customization project

When you finish a customization project, you can do one of the following:

- Generate a customization package file (`.rdar`) from the Screen Designer and associate it with a host session.
- Use the Rumba+ Server Build Tool to create a customization package file to be published to Rumba+ Server.

Using the Screen Designer command line

You can use the Screen Designer command line to:

- Generate one or more customization files from Screen Designer project files.
- Import multiple projects to a current Screen Designer project.

Syntax

```
<Install_Dir>\Plus\ScreenDesigner\RumbaScreenDesignerCLI.exe [/P
<project_file_path> | /L <project_file_list_path>] [/O <output_dir>][/S] | /I
<config_file> | /?
```

where:

`<Install_Dir>`	is the folder where the Screen Designer is installed.
`/P <project_file_path>`	is the fully qualified path to a single project file.
`/L <project_file_list_path>`	is the fully qualified path to a file that contains a list of project files.
	Each path and file name must be on a separate line. You can use the pound sign (#) to comment out a line.
`/O <output_dir>`	is the fully qualified path of where to export the customization files. Optional. If not specified, the `<Install_Dir>\Plus\ScreenDesigner` folder is used.
`/S`	saves the project after the customization file is generated.

/I *<config_file>*	is the fully qualified path of the configured list of projects to import.
/?	shows this information on the screen.

Generating customization files

Syntax

<Install_Dir>\Plus\ScreenDesigner\RumbaScreenDesignerCLI.exe [/P
<project_file_path> | /L *<project_file_list_path>*] [/O *<output_dir>*][/S]

Example: Generating a single customization file

RumbaScreenDesignerCLI.exe /P C:\Users*<user_name>*\AppData\Local\Micro Focus
\Rumba\Plus\Projects\TableTest\TableTest.rsdp /O C:\Users*<user_name>*\AppData
\Local\Micro Focus\Rumba\Plus\Rules /S

Example: Generating multiple customization files

RumbaScreenDesignerCLI.exe /L C:\Screen Designer\Projects\ProjectList.txt /O
C:\Users*<user_name>*\AppData\Local\Micro Focus\Rumba\Plus\Rules /S

Example project file list contents:

```
C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects\u1\u1.rsdp
#C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects\unique
\unique.rsdp
C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects\TableTest
\TableTest.rsdp
C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects
\longtable3\longtable3.rsdp
C:\work\rumba_mobile\TestApplications\TestPlanAppCSIMVS\TestAppCSIMVS
\TestAppCSIMVS.rsdp
```

Note: You can use the pound sign (#) to comment out a line.

Importing project files

Syntax

<Install_Dir>\Plus\ScreenDesigner\RumbaScreenDesignerCLI.exe /I *<config_file>*

Example

<Install_Dir>\Plus\ScreenDesigner\RumbaScreenDesignerCLI.exe /I C:\Screen
Designer\Projects\ImportFileList.txt

Example project file list contents:

```
duplicateresource=ckb
baseproject=C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects
\MergeProjects1\MergeProjects1.rsdp
project=C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects
\MergeProjects2\MergeProjects2.rsdp
project=C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects
\MergeProjects3\MergeProjects3.rsdp
project=C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects
\MergeProjects2\MergeProjects2.rsdp
project=C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects
\a1\a1.rsdp
project=C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects
```

```
\WYSIWYG\WYSIWYG\WYSIWYG.rsdp
saveto=C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus\Projects
\MergeProjects20
```

where:

duplicateresource	specifies copy options for duplicate resources:

cr	Copy and replace.
dc	Do not copy.
ckb	Copy, but keep both files.

baseproject	is the base project that the projects will be imported to.
project	is the absolute or relative path to the project file.
saveto	is the path of the resulting merged project. If not specified, the base project is used. If the project does not exist, a new project is created at that location with a name matching the folder name. For example, if the path is:

```
C:\Users\<user_name>\AppData\Local\Micro Focus\Rumba\Plus
\Projects\MergeProjects20
```

the created project is called `MergeProjects20`.

Migrating customization projects from earlier versions of Rumba

1. On the Rumba menu bar, select **Plus** > **Open Screen Designer**
2. On the Screen Designer menu bar, select **File** > **Open Project**.
3. Select the customization project file and click **Open**.
4. Select **File** > **Save**.

AS/400 Getting Connected

This section is intended to help you configure and maintain Rumba AS/400 Display and Printer connections. To do this, you use the Rumba **Connection Configuration** dialog box.

✎ **Notes:**

- To configure connections for features such as AS/400 file transfer and shared folders, use the AS/400 Communications tool. See *Using the AS/400 Communications tool*.
- To build a configuration file for the Rumba SNA engine, use the APPC Configuration tool. See *Rumba APPC Configuration tool*.

There is more to connecting to a host than just having the physical elements in place. When you use Rumba AS/400 software, you create a specific relationship between the host link and the interface options you configure.

The first time you make the host connection, you define these relationships following the steps outlined below. Afterwards, Rumba uses your customized parameters to configure the new session environment for you.

Connecting to the host

The connection procedures described here provide an overview of the tasks you must complete to configure an AS/400 interface for the first time.

1. Start the AS/400 Communications tool. To do this, select **Start** > **Programs** > **Micro Focus Rumba** > **Rumba Tools** > **Rumba AS400 Communications**.
2. In the **Rumba AS/400 Communications** dialog box, click the **Interface** tab.
3. In the **Installed Interface** box, select an interface.
4. Click **Configure**, then configure the interface.

Use the four pages of the **Rumba AS/400 Communications** dialog box to:

- Configure your connection for different types of interfaces.
- Set security options.
- Support an array of character sets.
- Display error messages for diagnostic purposes.

Starting the connection process

To start configuring an AS/400 host connection, open the **AS/4000 Connection Configuration dialog box**. To do this, in an AS/400 Display, select **Connection** > **Configure**.

The **General** page is the main page. The name and number of additional pages that are displayed depend on which interface you choose.

The name of the interface currently selected on the **General** page appears on one tab (for example, **Rumba Router**) and, for all but the **TN5250** and **Demo Host** interfaces, a **Security** tab is displayed as well.

Procedures for configuring each interface are described in the online Help.

The Demo host

The Demo host lets you communicate with a simulated host through a series of PC-to-host conversations previously recorded with the Data Stream Analyzer (DSA) Trace system.

When you use this host, Rumba sends commands to a program emulating a host. The host program responds with the corresponding 5250 data stream segments, based on your keyboard or mouse input.

The Demo host is especially useful for training purposes. You do not have to configure the host or the session parameters to use the Demo host.
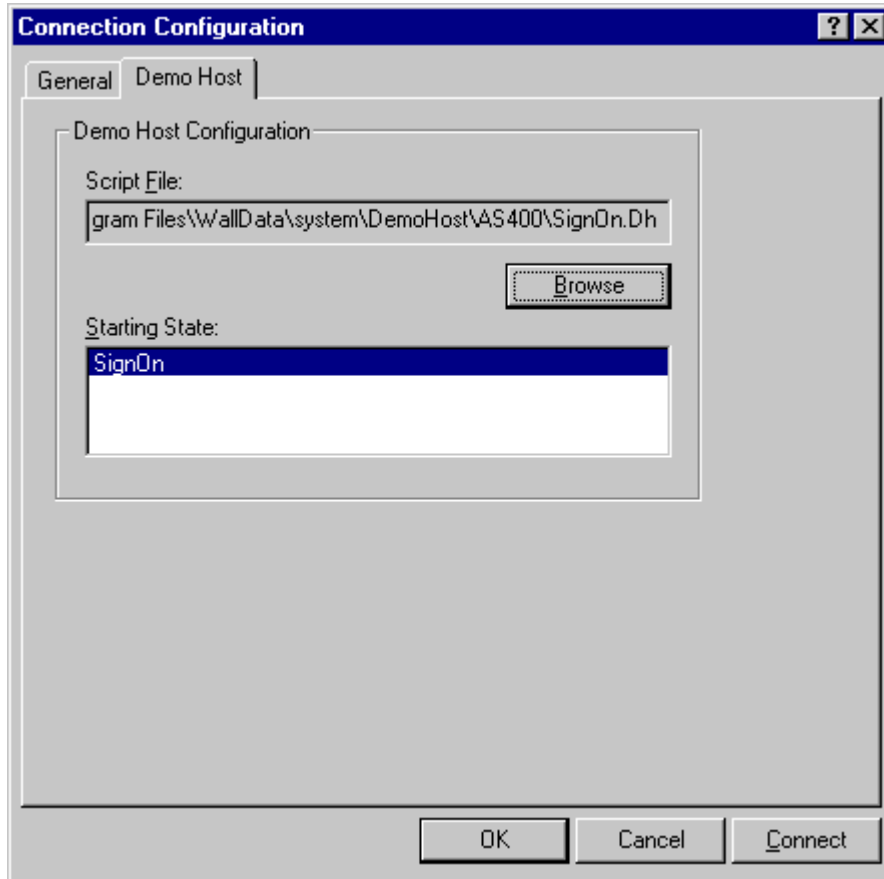
⚠️ **Important:** These fields are initialized with defaults based on data from your current profile, if available, or from system defaults. They are the recommended settings and should not be changed except for special circumstances. Selecting an inappropriate starting state could result in undesirable behavior.

Selecting an interface

When establishing a host connection, you need to determine which interface to use. By matching your network topology with one of the examples below, you can start choosing the most appropriate interface.
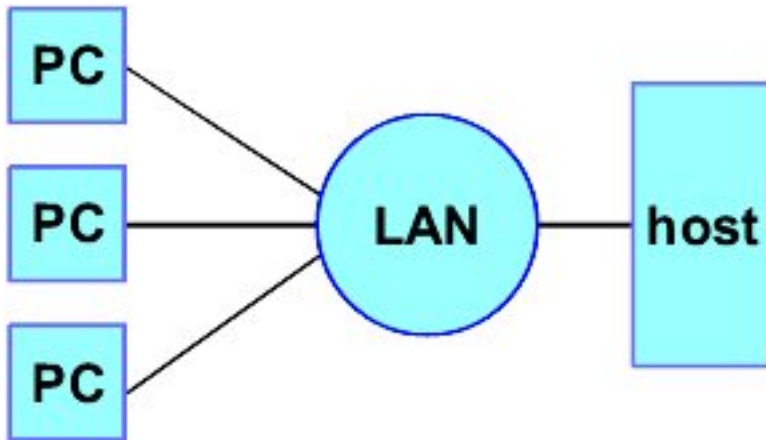
LAN Direct is used when the LAN is directly connected to a host. Examples of this connection type are listed below:

- MPTN (SNA over TCP/IP) - a Rumba Router link type
- TN5250
- LAN gateway
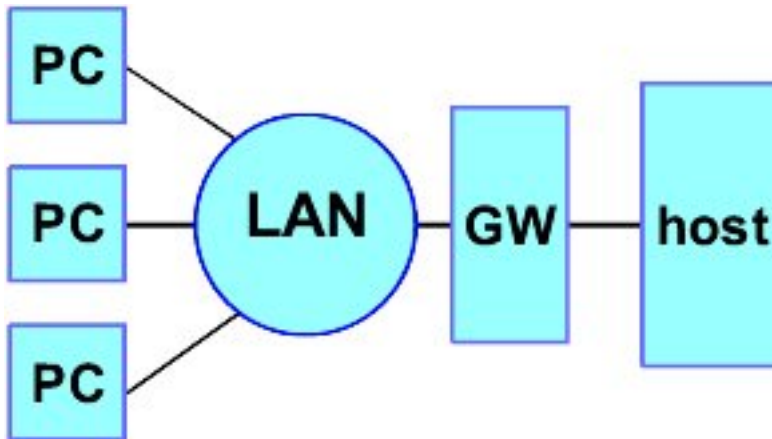
Using a LAN gateway connection, the LAN is connected first to a gateway, usually for protocol conversion. IPX and NetBIOS are two examples of protocols that require gateways, because neither are native to normal host environments. An example of this connection type is Microsoft SNA Server.

Configuring the interface

After you have determined which interface best suits the needs or constraints of your connection requirements and network topology, you need to tell Rumba how to find your host and how your host can find you (the client).

To do this, you use the Connection Configuration dialog box. The best initial approach is to leave the settings of the majority of the fields used to specify client and host data at their default values.

The connection configuration parameters required depend on the interface you select. For Rumba Router link types, the **Rumba Router** page displays additional tabs for you to specify configuration values.

Successful connection indicator

Wjhen the host connection is established, the status bar description area reads `Connected`.

When the cursor is moved around the screen, the description area shows the last command selected.

The status bar displays `SA` (system available) in the first status box to indicate that the AS/400 is ready to respond.

Other helpful features

In addition to the connection methods already discussed, Rumba has additional features designed to save you time, make the connection task easier, and help you understand the entire process better.

The administrator override file

A high priority for any system administrator is standardizing system configurations. When you create an administrator override file, you provide a set of well-defined parameters that each user can access. This tool can significantly reduce the time spent setting up and troubleshooting workstations.

An administrator override file is a configuration file you create, then place on the network. Users can access this file for connection information to SNA-type machines. If the connection information changes, you can use the administrator override feature to make one central change.

Using the Rumba Router to configure an APPC link

The Rumba Router supports Advanced Peer to Peer Networking (APPN), and the Common Programming Interface for Communications (CPI-C). Workstations using the Rumba Router can connect to AS/400s over a LAN or direct physical connection.

This section provides an overview of the procedures used to specify the settings for a host connection using the Rumba Router.

Overview of Rumba Router

Using the Rumba Router provides you with support for a wider range of client-based applications and connectivity types. By processing API commands on the client workstation, the Rumba Router takes advantage of the benefits of distributed computing to enhance both the performance and flexibility of the applications you use to interact with your AS/400 host.

There are common elements within both the **Connection Configuration** and **Link Configuration** dialog boxes, regardless of which Rumba Router link type you use to make a connection. Because of this, specifying settings does not vary on the **Rumba Router** and **Security** pages in the **Connection Configuration** dialog box, nor on the AS/400 **AS/400 Link** and **Advanced** pages of the **Link Configuration** dialog box.

When you select the Rumba Router, the **Connection Configuration** dialog box displays the **Rumba Router** and **Security** tabs in the second and third tab positions.

When you click the **Rumba Router** tab and add a new connection, you see the **Link Configuration** dialog box. This has three pages:

- The first is always **AS/400 Link**.
- The second one changes to the name of the link type that you select.
- The third is always **Advanced**.

Connection Configuration dialog box

Workstation users normally use the **Connection Configuration** dialog box to configure the connection from an AS/400 display or printer session. This is instead of using:

- The **Rumba APPC Configuration** tool, which is more suited for creating configuration files for SNA interfaces.

- The **Rumba AS/400 Communications** tool, which is used to configure host connections for Rumba client applications.

The installed interfaces that appear in the **Connection Configuration** dialog box include:

- The Demo host
- Rumba Router
- TN5250
- Microsoft SNA Server (if the SNA client is installed)

Setting up the connection

Setting up connections with the Rumba Router generally involves three steps:

1. Configure the Rumba Router interface.
2. Choose a link or connection type.
3. Configure the link type.

Because the settings you can specify vary so widely depending on which link type you choose, each link type is discussed. The **Rumba Router** and **Security** pages do not change for each link type. The remainder of this section outlines what you can specify on these pages.

Rumba Router page

The table below describes the parameters of the **Rumba Router** page in the **Connection Configuration** dialog box.

Parameter	Description
AS/400 Information	Use this group box to select or edit an existing configuration, or add a new configuration to the **System Name** list. Configurations are not displayed in the **System Name** list until you have added them.
System Name	Lists all the currently configured systems. Click the down arrow to select a system.
Set Default	Designates the selected system as your default connection, which will be used automatically unless another system is selected before connecting. The Default System is displayed in the **System Name** list with an asterisk (*).
Add	Configures a new connection. The new connection will be displayed in the **System Name** list after you have added it.
Remove	Deletes a configured connection. If you delete the default system, you must select or configure a new system before connecting.
Properties	Opens the **Link Configuration** dialog box for the selected connection. You can edit the available parameters and save the changes to the configuration.
Session Information	Use the controls in this group to specify how user information for this session is specified.
Auto sign-on	When you check this box, your connection information is used for the host sign-on.
Device Name	You can add an optional device name that will override the default device name used for this session.
	In most cases, the device name defaults to your PC system name followed by S1 or P1. Instead of using the default, you can enter a device name which the AS/400 uses to recognize this session.
PC Information	Lists the information that is used when connecting your PC with the AS/400.
PC Name	Add the name you use to log on to your network. This corresponds to the controller ID configured on the AS/400, and can be up to eight characters, including A-Z, 0-9, @, #, and $.
Network Name	Add the name of the network you are using to connect. For example, you may be using APPN.
	On the AS/400, this corresponds to the local network ID, and can be up to eight characters, including A-Z, 0-9, @, #, and $. This identifies the AS/400 on the network.

You can find step-by-step procedures for configuring individual interfaces and link/connection types in the online Help.

The interface Help files are a self-contained unit with information on requisite stacks, configuration procedures, and troubleshooting information (including most error messages associated with an interface). When necessary, the Help also assists you in locating values for interface parameters defining the local PC and the remote host.

Security page

The table below describes the parameters on the **Security** page of the **Connection Configuration** dialog box.

Parameter	Description
Sign-on Features	
Use Common Sign-on	Uses the specified user ID and password as sign-on information for all systems.
AS/400 Sign-ons	Lists the currently known AS/400 systems for which you have configured security settings. You must add systems on the **Interfaces** page before you can configure security settings for them.
Add	Sets up security for a new system.
Change	Changes the security settings for the selected system.
Delete	Deletes the security settings for the selected system.
Enable Caching of User IDs and Passwords	Enables the storing of the user ID and password for the currently configured system until you reboot your PC. If you check this option, you do not have to enter your user ID and password for subsequent connections unless you reboot.

Rumba Router link types

To configure an AS/400 Rumba Router link, you use the **Link Configuration** dialog box. To open the dialog box:

1. Select **Connection** > **Configure**. The **Connection Configuration** dialog box appears.
2. In the **Installed Interfaces** list, select **Rumber Router**.
3. On the **Rumber Router** page, select the system name from the drop-down list.
4. Click **Properties**. The **Link Configuration** dialog box appears.

AS/400 Link page
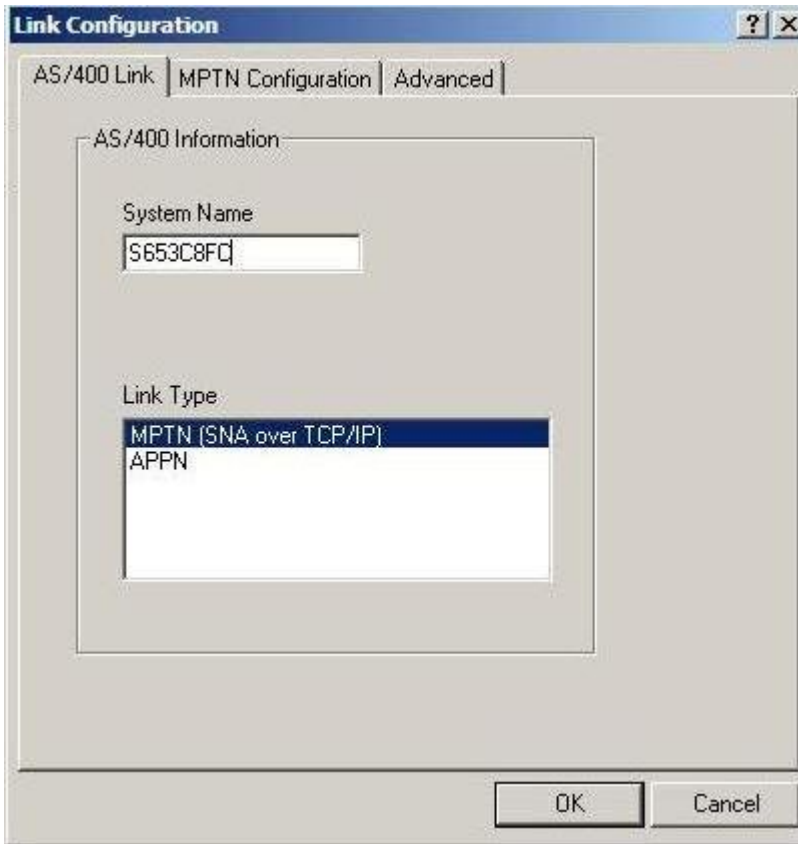
The table below describes the parameters on the **AS/400 Link** page of the **Link Configuration** dialog box.

Parameter	Description
AS/400 Information	
System Name	Displays the alias for the AS/400. This name is added to the list of configured connections on the **Rumba Router** page.
	For example, if you connect to a computer called **S1001234**, you could call the computer **Bill** instead of **S1001234**. By default, this name is used as the **LU Name** on the **Advanced** page, although you can change this at any time.
	To change the **System Name** or other connection parameters later, select it from the **System** page, then click **Properties**. If you are going through an adjacent network node (system) via an APPN connection to another system, enter the name of the target system.
Link Type	Displays the link types you can use to connect. You must select a link type to complete the connection configuration.
	Your choice of link type depends on your hardware and software configuration, as well as the type of connection

Parameter	Description
	you need. If you do not know which link type to use, check with your system administrator.

Advanced tab
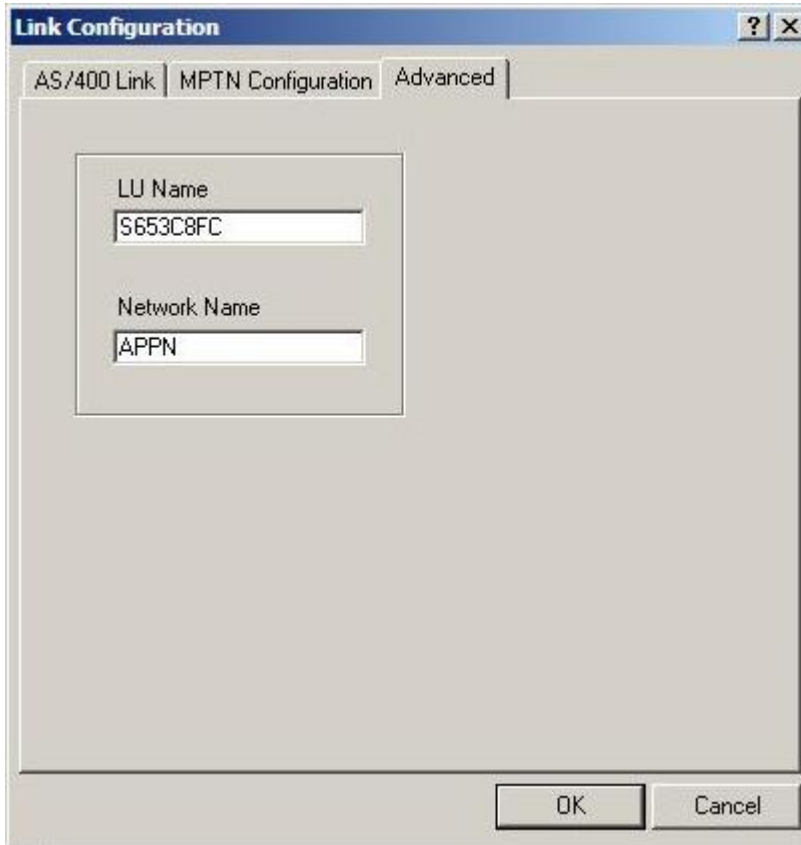
The table beow describes the parameters on the **Advanced** page of the **Link Configuration** dialog box.

Parameter	Description
LU Name	Displays the name of the AS/400's logical unit. This is not an alias. This is the actual name of the AS/400 and is used to make the connection. The **LU Name** corresponds to the **System Name** on the **Rumba Router** page. The **System Name**, however, can be an alias, while the **LU Name** is always the actual name of the AS/400.

This name can be up to eight characters, including A-Z, 0-9, @, #, and $. |
| **Network Name** | Identifies the network you are connecting over. The **Network Name** corresponds to the local network ID on the AS/400, and can be up to eight characters, including A-Z, 0-9, @, #, and $.

This name is used to identify the AS/400 on the network. |

Link types

The table below describes types of link that the Rumba Router supports.

Link Type	Function
APPN	Connects to an AS/400 host by using an existing network connection as a node. For example, an APPN connection between your PC and an AS/400 host can be used to access another AS/400 host that does not have a direct physical connection to your PC.
MPTN (SNA over TCP/IP)	Connects to an AS/400 host using SNA or APPC to communicate over TCP/IP networks. The full capability of the Rumba APPC engine is used over an existing TCP/IP backbone, linking client workstations to the host and providing transparent access to functions such as AS/400 printing and file transfer.

Native TCP support

Rumba AS/400 software supports access via native TCP for a number of client applications. With the implementation of this feature, Rumba provides support for the complete range of client applications. Companies moving to TCP/IP networking can still give their users access to key AS/400 applications which are important for user productivity.

Rumba TCP architecture

The figure below shows the data flow from your workstation to your AS/400:
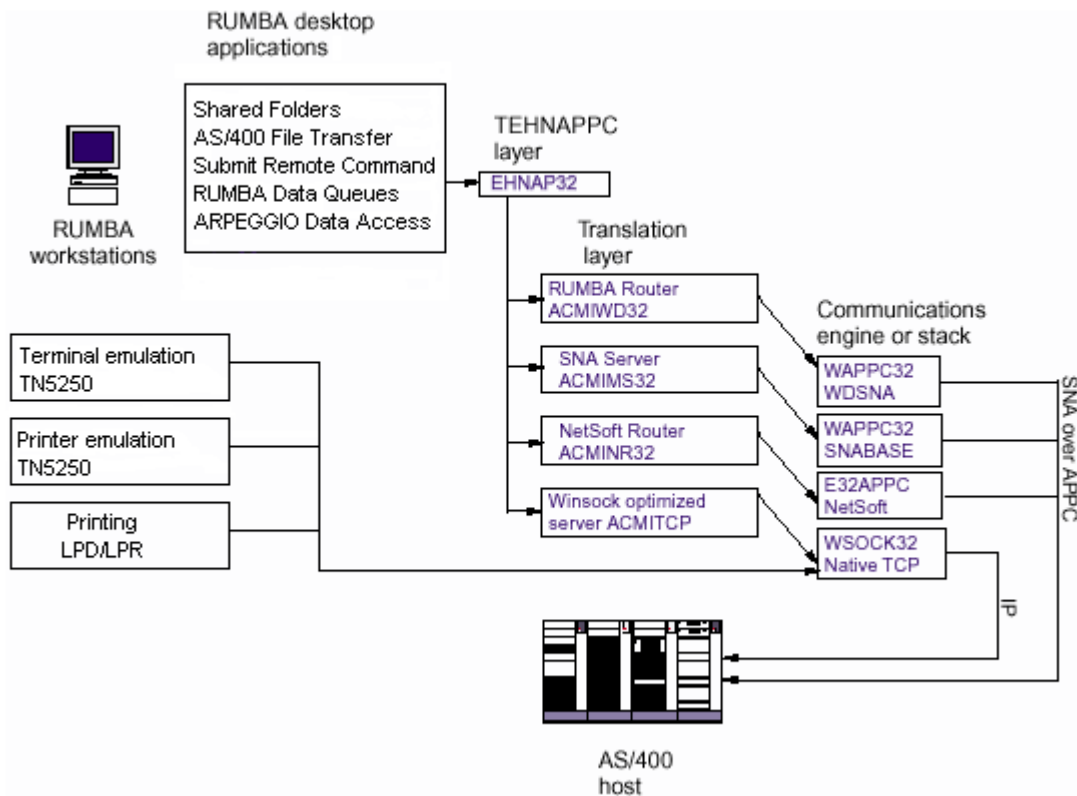
Rumba 's Winsock support allows you to use your existing TCP/IP connection as the link between the **AS/400 File Transfer** tool or **Submit Remote Command** and your host.

Rumba AS/400 Display and AS/400 Printer also provide display and printer support directly over TCP/IP.

Supported Rumba components

The table below describes the functions performed by the Rumba components that use native TCP.

Component	Function
Display	
Rumba AS/400 Edition Display & TN5250 interface	Presents the traditional AS/400 display terminal in a Microsoft Windows application: the Rumba window.
	Users can run multiple host sessions, displaying and working with host applications simultaneously.
Printing	
Rumba AS/400 Printer	Includes standard features such as AS/400 printer queues, PC printer queues, and print log.
	Note: Printing with TCP/IP is now available using TN5250.
File Transfer	
AS/400 File Transfer	Transfers files or selected data between the AS/400 host and your PC.

Configuring a TCP connection

The procedure for configuring your host connection depends on which desktop application you use:

- If you plan to use Data Access, use the 32 bit ODBC program, in Windows Control Panel, to define and manage a data source for the Rumba AS/400 Optimized Server ODBC driver. If you need to configure a Rumba APPC connection, click **Configure Rumba APPC**.
- If you plan to use TN5250, use the **Connection Configuration** dialog box to configure a Rumba Router TN5250 host link.

Trace support for TCP applications

To obtain trace data from your host connection, use the Rumba Trace tool.

For more information, see *Rumba Trace tool*.

Configuring an MPTN connection

The options you can specify on the **MPTN Configuration** page of the **Link Configuration** dialog box vary a great deal depending on which link type you use. This section discusses the options you specify to configure a host connection using the Rumba Router and an MPTN link type.

About MTPN

In this guide, the MPTN interface is described as a LAN direct connection type.

Multi-Protocol Transport Networking (MPTN) is an open architecture that allows integration of multiple-vendor and multiple-protocol environments.

The primary advantage of MPTN is that you can use either SNA or APPC to communicate over TCP/IP networks. This means that you can use the full capability of the Rumba APPC engine over your existing TCP/IP backbone. Effectively, this links your client workstation directly to the AS/400, providing transparent access to functionality, such as AS/400 printer and file transfer.

Rumba MPTN supports various APIs:

- EHNAPPC
- CPIC
- APPC

Various applications work with Rumba MPTN:

- Database applications
- Printer configurations
- AS/400 Display
- AS/400 File Transfer

AS/400 considerations

Before you can connect to an AS/400 using the MPTN interface, the AS/400 must be set up to use TCP/IP. For your AS/400 to communicate using TCP/IP, it must be running OS/400, Version 3.0, Release 1 (or later).

Note: This is an absolute requirement. Earlier versions of the operating system do not allow you to properly configure TCP/IP addresses for either your AS/400 or your client workstations.

MPTN parameters

The table below describes the parameters on the **MPTN Configuration** page of the **Link Configuration** dialog box.

Parameter	Description
TCP/IP Domain Name	
Remote Host Name	Type the remote host name or IP address you want to access.
Security	
Use Host APPN Remote Location List	When checked, the system name sent with your connection request is the **Link Name**/**Remote LU Name** you entered for this workstation. When unchecked, the system name sent with your connection request is **MPTN**.
	You can use this security option when you configure a connection to an AS/400 host that uses APPN Remote Location List security.
MPTN Options	
KeepAlive	Specifies whether host KeepAlive settings are negotiated as part of the connection process.

Configuring a Microsoft SNA Server connection

In this guide, the Microsoft SNA Server interface is described as a LAN gateway connection type.

Before running Rumba software with the Microsoft SNA Server interface, you must:

- Run the Microsoft-supplied configuration utility and configure the workstation as an APPC client.
- Run the supplied `snabase.exe` utility to start communication between the workstation and the Microsoft SNA Server.

Diagnostics

You can use the following tools to diagnose connection problems that your users might encounter when trying to establish a host connection:

Rumba Trace tool	Traces and records SNA and link layer communication sessions. These trace options can have an impact on your system performance, so it is best to use them as a filter to sift through your communication sessions when isolating a problem.
Rumba Event Viewer	Works in the background collecting information at timed intervals that you can set, and keeps an event log organized by Data, Time, Source, Event ID, Type, and Computer.
	Note: The Rumba Event Viewer is very similar in appearance and functionality to the Windows Event Logger.

Configuring the AS/400 Communications tool

You use the Rumba AS/400 Communications tool to configure host connections for:

- AS/400 File Transfer
- Shared Folders application and API
- Data Queues API
- Submit Remote Command API

- Virtual Print application and API

Configurations are saved as a file with an `.ehn` extension. You can create multiple configuration files for each interface that you use.

Connecting to the host

The connection is configured from the **Connection** menu.

1. Select the **Connection** > **Configure**.
2. Select the interface to use.
3. Click **Properties** and configure the interface, if required.
4. Click **Connect**. After connecting to the host, you can save your settings as a profile.

Interface page
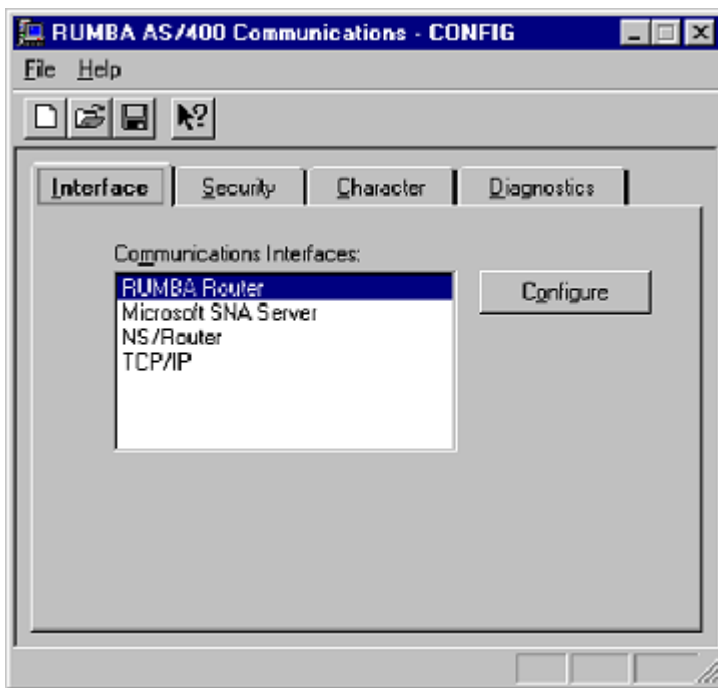
The table below describes the parameters on the **Interface** page of the **Rumba AS/400 Communications** dialog box.

Procedure	Description
Communications Interfaces	Lists the installed communication interfaces. Select which type of communication interface to use when connecting this profile to the host.
Configure	Configures the selected interface (displays the **Rumba Router** dialog box). If the interface has already been configured, you do not need to do so again, unless you want to make changes.

Procedures for using the Rumba Router and configuring other communication interfaces can be found in:

Using the RUMBA Router to Configure an APPC Link

Security tab

Security page
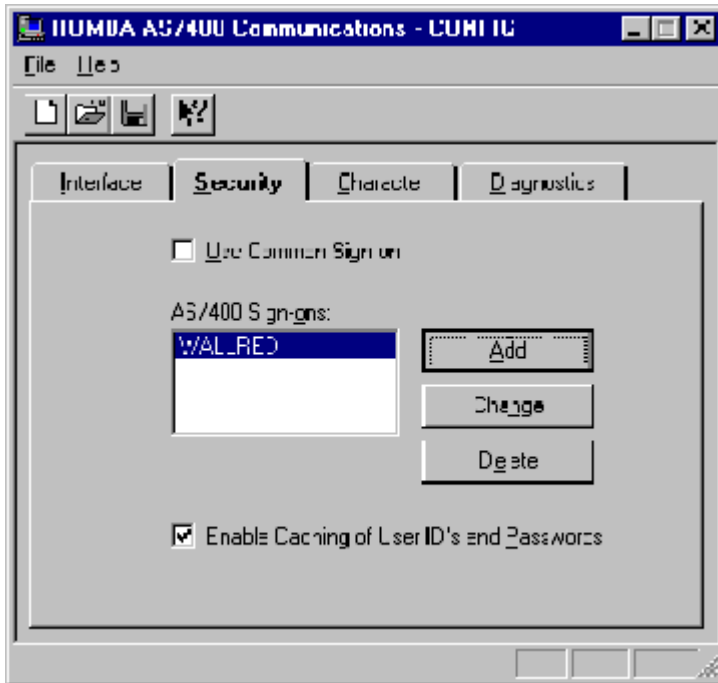
The table below describes the parameters on the **Security** page of the **Rumba AS/400 Communications** dialog box.

Parameter	Description
Use Common Sign-on	Uses the specified User ID and Password as sign-on information for all systems.
AS/400 Sign-ons	Lists the current AS/400 systems you have configured security settings for. You must add systems on the **Interfaces** page before you can configure security settings for them.
Add	Sets up security for a new system (displays the **Add System Sign-on** dialog box).
Change	Changes the security settings for the selected system.
Delete	Deletes the security settings for the selected system.
Enable Caching of User ID's and Passwords	Enables storing of the user ID and password for the currently configured system until you reboot your PC. If you check **Enable Caching of User IDs and Passwords**, you not have to enter your user ID and password for subsequent connections, unless you reboot.

Add System Sign-on dialog box

Parameter Name	Description
AS/400 System Name	Lists the selected system name.
User ID	Becomes the user ID that is required when signing on to the selected system.
Password	Becomes the password that is required when signing on to the selected system. This is optional, unless a

Parameter Name	Description
	password is required by the system you are connecting with.
Verify Password	Verifies the correct spelling of your password. An error message is displayed if this password does not match the first password you entered.
OK	Saves any changes you have made and closes the dialog box.
Cancel	Ignores any changes you have made and closes the dialog box.
Reset	Removes changes and restores the original settings.

Character page

The table below describes the parameters on the **Character** page of the **Rumba AS/400 Communications** dialog box.

Parameter	Description
Host Character and **Set/Code Page**	Displays all the character sets and corresponding code pages available for the AS/400 display.

Diagnostics page

The table below describes the parameters on the **Diagnostics** page of the **Rumba AS/400 Communications** dialog box.

Parameter Name	Description
Display EHNAPPC Error Messages	Displays additional diagnostic error message boxes when a communications error occurs. These do not replace standard error messages.

AS/400 File Transfer

This section provides a general overview of the Rumba AS/400 File Transfer program, together with reference information.

Rumba AS/400 File Transfer sends data between the host and the PC. Users can run AS/400 File Transfer to send host files or selected host data to the PC, work with the data in a PC application, then run AS/400 File Transfer to send the modified PC data back to the AS/400.

In addition to AS/400 File Transfer, you can also transfer files using Mainframe Passthrough.

For detailed information on configuring individual and batch file transfers, see the online Help.

AS/400 File Transfer features

AS/400 File Transfer provides multiple options for transferring data between the AS/400 host and the PC. You can:

- Create, open, and save file transfer settings in a profile
- Run individual or multiple (batch) file transfers
- Transfer text or database files from the PC to the AS/400
- Use drag-and-drop to transfer host data to the PC display, the printer, or to a PC file
- Transfer complete host files to the PC or use SQL to transfer only the information you need
- Display libraries you can access on the host, and add libraries for the current file transfer session
- Specify decimal, date, and time formats for the transferred data
- Remove End-of-File (EOF) characters from host files before sending to the PC
- Specify transfer options such as **Run minimized**, **Monitor percent complete**, **Notify when complete**, **Close when complete**, **Truncate records**, **Truncate spaces**, and **Auto Run**.
- Forward host files to an e-mail address

Installation

AS/400 File Transfer is included in the default installation of Rumba AS/400 Edition. When performing a custom installation of Rumba AS/400 Edition, you can select or deselect the **AS/400 File Transfer** option.

After installing AS/400 File Transfer on a workstation, you can create and distribute customized file transfer profiles that your users can run easily.

Using profiles

You can simplify the file transfer process for your users by creating and distributing profiles for all data transfers that need to be run regularly. For example, users might need to create weekly reports or spreadsheets based on host data. To automate the file transfer program, go through the following steps:

1. Analyze the problem.

   Which users transfer data to and from the host on a frequent or regular basis? What kinds of transferred data do they need? How would they best benefit from an automated file transfer procedure?
2. Design individual file transfer profiles.

   Based on your analysis, create customized file transfer profiles to meet the individual needs of your users. This includes configuring:

- The default communications interface for AS/400 File Transfer
- Transfer options such as **Run minimized**, **Notify when complete**, and Monitor percent complete
- The location and format of the data files on the PC and the host
3. Use SQL statements if appropriate.

   When sending AS/400 data to the PC, you can use SQL statements to transfer only the AS/400 data that users need, instead of transferring complete files. You can also sort, group, summarize, and perform mathematical calculations on the transferred data.
4. Create batch transfers that run multiple file transfers in a single step.

   You can group individual file transfer profiles in a batch profile so users can run the specified transfers in a single step. Batch transfers can also be scheduled to run once, or on a daily, weekly, or monthly basis.
5. Use the **Auto Run** option to run file transfers from the Windows Desktop.

   When you save individual or batch transfer profiles with the **Auto Run** option, users can run their file transfer profile by simply double-clicking a shortcut on the Windows Desktop.

User files

The table below describes the different types of user file that are associated with Rumba AS/400 File Transfer and which are stored in the user's private directory:

File name	Description
`.btf`	Batch transfer profile for multiple file transfers
`.fdf`	File description file (required when sending database files from the PC to the AS/400)
`.rto`	Transfer profile for a single file transfer
`.tfr`	PC Support/Client Access file transfer profile (upload)
`.tto`	PC Support/Client Access file transfer profile (download)

Changing the library list on the AS/400 host

When AS/400 File Transfer starts, Rumba reads the `QDFTJOBD` (default job description) file on the AS/400 and displays the list of libraries included in this file. To display other host libraries which your users have access to, you must change the `QDFTJOBD` file. You can change the file for the entire system or for a single user.

To change the QDFTJOBD file of a single user:

1. Copy the `QDFTJOBD` file into a new host library.
2. Change the library list as needed in the copied `QDFTJOBD` file.
3. Update the user profile to reflect the new `QDFTJOBD` file. Generally, the host uses the `QDFTJOBD` file in the QGPL library by default.

To change the `QDFTJOBD` file for the entire system, follow only the instructions *Changing the library list in the QDFTJOBD file*.

If you need to change the `QDFTJOBD` file during a file transfer session, select **Transfer** > **Request Libraries** to update the current library list.

Copying the QDFTJOBD file

1. At the AS/400 command line, type `WRKJOBD`, then press **F4**. The **Work with Job Descriptions** screen appears.

2. At **Job Description**, type `QDFTJOBD`.

3. At **Library**, type `QGPL`

4. Press **Enter**.

5. Under **Opt**, type `3` to copy the file, then press **Enter**.

   The **Create Duplicate Object** screen appears.

6. At **To Library**, type the new library name for the copied file, then press **Enter**.

Changing the library list in the QDFTJOBD file

1. At the AS/400 command line, type `WRKJOBD`, then press **F4**. The **Work with Job Descriptions** screen appears.

2. At **Job Description**, type `QDFTJOBD`.

3. At **Library**, type the name of the new library containing the copied `QDFTJOBD` file.

4. Press **Enter**.

5. Under **Opt**, type `2` to make changes to the file, then press **Enter**.

   Te **Change Job Description** screen appears.

6. Press **F10** and **Page Down**.

7. Under **Initial Library List**, at **+ for more values**, type the plus sign (**+**) and type the names of the host libraries you want to list.

8. Press **Enter** twice.

Updating the user profile

1. At the AS/400 command line, type `CHGUSRPRF`, then press **Enter**.

2. At **User Profile**, type the name of the profile to change.

3. Press **F10**, then press **Page Down** twice.

4. Under **Job Description**, at **Library**, type the name of the library containing the new `QDFTJOBD` file, then press **Enter**.

Creating a PC description file

When sending a PC database file to the AS/400, file and field descriptions for both the AS/400 system and the PC file are required. The host uses this information to correctly identify and interpret the field data.

The PC description file (`.fdf`) contains information describing the PC database file. It is a text file that identifies the file data type (such as text or DIF), and contains field data including the name, data type, and length of each field in the transferred file.

When transferring data from the AS/400, you can create a PC description file automatically by checking **Save description file** in the **To PC - Options** dialog box:
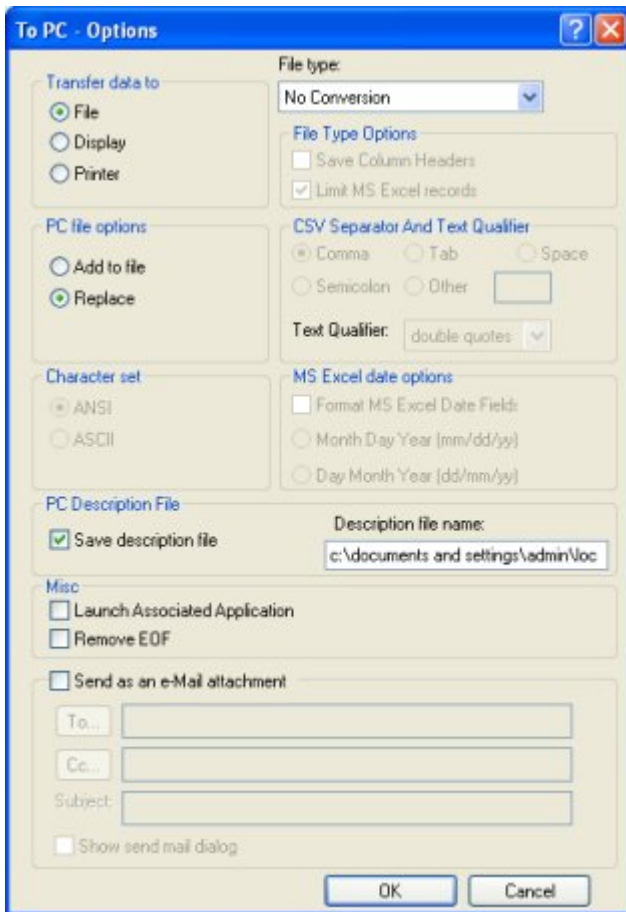
Conversely, the PC description file is required when sending a PC database file to the host. If necessary, you can use a text editor to create a PC description file, following these conventions:

- Use ASCII format.
- Use a carriage return and line feed to end each record.
- Tab characters can be used, but they will be treated as spaces.
- Place an End-of-File (EOF) character at the end of the file.
- Recommended naming convention: Use the same name as the PC database file you are transferring, but with an `.fdf` extension.

Structure of the PC description file

The PC description file contains four types of information that describe the PC database file:

- PCFDF
- PCFT file type
- PCFO (optional time, date, and decimal formats)
- PCFL entries (field name, data type, length) for each database field

These information types are described below.

PCFDF

   Identifies this file as a description file.

PCFT file type

   Identifies the file type of the PC database file.

Indicator	File type
1	ASCII
3	BASIC sequential (CSV)
5	Data Interchange Format (DIF)
6	No-conversion file
7	Reserved (DBCS version only)
8	DOS random type 2
9	Binary Interchange Format (BIFF)
10	Tab-delimited text

PCFO

Optional PC time, date, and decimal format entries.

Indicator	Time format name	Time format
1	HMS	hh:mm:ss
2	International Standards Organization (ISO)	hh.mm.ss
3	USA standard (USA)	hh:mm AM or PM
4	European (EUR)	hh.mm.ss
5	Japan Industrial Standard, Christian Era (JIS)	hh.mm.ss
6	DDS	Format given by AS/400 file attributes
7	DFT	Host job default
*	Unspecified	Host job default

Indicator	Time Separator
1	Colon (:)
2	Period (.)
3	Comma (,)
4	Blank ( )
5	Null (N)
6	Default (D) (host)
*	Unspecified (host job default)

Indicator	Date 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

Indicator	Date format name	Date format
7	EUR	dd.mm.yyyy
8	JIS	yyyy-mm-dd
9	DDS	Format given by AS/400 file attribute
10	DFT	Host job default
*	Unspecified	Host job default

Indicator	Date Separator
1	Slash (/)
2	Dash (--)
3	Period (.)
4	Comma (,)
5	Blank ( )
6	Null (N)
7	Default (D) (host)
*	Unspecified (host job default)

Indicator	Decimal Separator
1	Period (.)
2	Comma (,)
3	Default (D) (PC country default)
*	Unspecified (PC default used)

PCFL entries

Describes the database fields. Include one PCFL entry for each field in the database. Each entry consists of:

- Personal Computer Field Length (PCFL)
- The field name (maximum 10 characters)
- A number designating the data type
- The field length in characters
- PCFL entries

Indicator	Data type
1	ASCII
2	ASCII numeric
3	Hexadecimal
4	Binary
5	Zoned
6	Packed
7	BASIC integer
8	BASIC single-precision floating point
9	BASIC double-precision floating point

Indicator	Data type
10	EBCDIC
11	EBCDIC zoned
12	EBCDIC packed

Sample PC description file

The following is a sample PC description file with commentary:

File entry	Comment
PCFDF	PC description file
PCFT 9	PC file in BIFF format
PCFO 7,6,10,7,*	Host default date and time formats
Option Settings	
PCFL BINARY 9 7/2	The first field, BINARY, is double-precision floating point and 7 characters in length, precise to 2 decimal places.
PCFL CHAR 1 10	The second field, CHAR, is an ASCII field and has a length of 10 characters.
PCFL ZONED 9 8/4	(The other fields in the database are interpreted as above.)
PCFL PACKED 9 11/2	
PCFL HEX 3 10	
PCFL DATE 1 8	
PCFL TIME 1 8	

Using SQL: Examples

When transferring data from the AS/400 host to the PC, you can use the system default to transfer complete files, or you can use SQL (Structured Query Language) to send specific AS/400 data. SQL is a standardized query language for requesting information from a host database. You can use SQL to transfer data from specified fields and records in one or more host files, sort the transferred data, and perform simple calculations on the transferred data.

This section contains three short examples of how to use SQL to query data. The examples demonstrate various query keywords using sample data from an imaginary grocery store. There are three example queries:

- Example 1 is a simple query using SELECT, WHERE, and ORDER BY.
- Example 2 selects data from three files using JOIN BY.
- Example 3 transfers summary records using GROUP BY and HAVING.

To build a query, you must first specify the AS/400 data file(s) in the **Lib/File(Member)** box on the main AS/400 File Transfer screen. These examples use three sample AS/400 files named PRODUCT, SUPPLIER, and SUPPROD in a library named GROCERY. The first few records of each file are listed in the following tables.

ProdCat	ProdID	ProdDesc	ProdName	ProdQuant
100570265	129930113	Wild Berry	Summerbrook Wild Berry Seltzer	84

ProdCat	ProdID	ProdDesc	ProdName	ProdQuant
100570265	129930513	Apple-Cran	Summerbrook Apple-Cran Seltzer	-9
100570265	129935223	Black Currant	Olympia Black Currant Seltzer	125
100570267	129935423	Sparkling Lemonade	Olympia Sparkling Lemonade	7
100570262	761837634	Cherry Lime	Olympia Cherry Lime Soda	54
100570263	761831637	Cranberry Ice Tea	Crystal Falls Cranberry Ice Tea	302
100570263	761831635	Diet Peach Ice Tea	Crystal Falls Diet Peach Tea	79
100570263	761831636	Diet Rasp Ice Tea	Crystal Falls Diet Rasp Ice Tea	110
100570263	761831638	Lemon Ice Tea	Crystal Falls Lemon Ice Tea	218
100570100	165712111	Mineral Water	Whisper Rain Mineral Water	22
100570262	165712115	Mango Passion soda	Dave's Mango Passion soda	-22

Example 1: Using SELECT, WHERE, and ORDER BY

To transfer a list of all available types of tea drinks sorted alphabetically:

1. Specify the GROCERY/PRODUCT file in the **Lib/File(Member)** field on the main file transfer screen, then open the **From AS/400 - Options** dialog box:
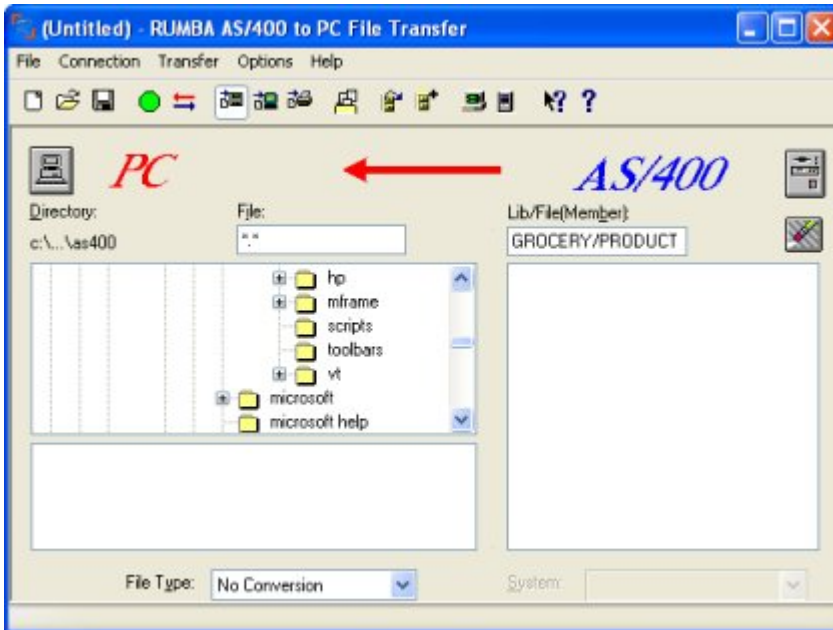
2. Verify that the **Select** field on the **Overview** page contains an asterisk (*). This indicates that all fields in the PRODUCT file will be transferred.

3. To transfer all records containing the string Tea, type the following string in the **Where** field:

PRODNAME LIKE '%Tea%'

The % character is a wildcard indicating any number of characters:
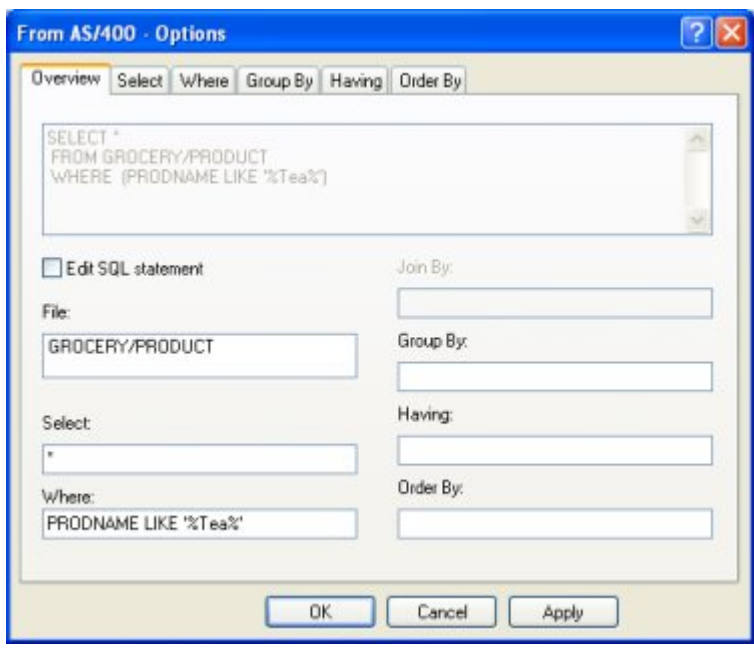
4. To sort the data in ascending order by the type of drink, type the following string in the **Order By** field:

PRODDESC

5. Click **Apply** to update the query fields:

```
SELECT *
FROM grocery/product
WHERE (PRODNAME LIKE '%Tea%')
ORDER BY PRODDESC
```

The resulting records are:

	ProdID	ProdDesc	ProdName	ProdQuant
100570263	761831637	Cranberry Ice Tea	Crystal Falls Cranberry Ice Tea	302
100570263	761831635	Diet Peach Ice Tea	Crystal Falls Diet Peach Tea	79
100570263	761831636	Diet Rasp Ice Tea	Crystal Falls Diet Rasp Ice Tea	110
100570263	761831638	Lemon Ice Tea	Crystal Falls Lemon Ice Tea	218

Example 2: Querying multiple files with JOIN BY

This query finds the suppliers and prices for a given product. Multiple files are joined so they can be queried as if they were one file. This is necessary because the GROCERY library stores data in separate files linked by common fields, rather than using one large file: The PRODUCT and SUPPROD files share the ProdID field, and the SUPPROD and SUPPLIER files share the SupID field.
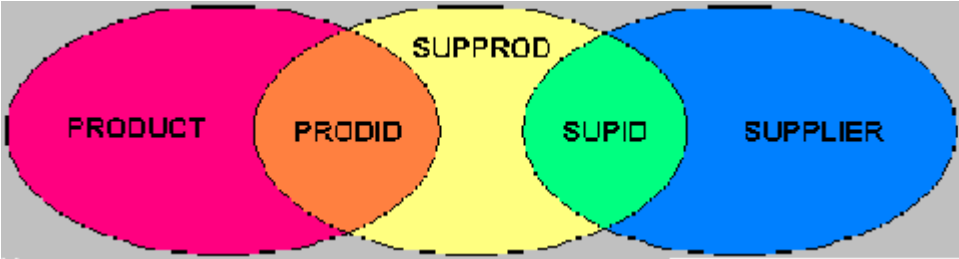
With `JOIN BY`, the relationships between files become part of the query. Without joining the files, you would need three queries to get the same result: one for product ID, one for product cost, and one for supplier name.

To find the suppliers and prices for Dave's Mango Passion soda:

1. Specify the `GROCERY/PRODUCT`, `GROCERY/SUPPROD`, and `GROCERY/SUPPLIER` files in the **Lib/ File(Member)** field on the main file transfer screen, then open the **From AS/400 - Options** dialog box.

2. To transfer only the product name, product cost, and supplier name fields, type the following string in the **Select** field on the **Overview** page:

   `T1.ProdName, T2.ProdCost, T3.SupName`

3. To join records in the three files by the product ID and supplier ID, type the following string in the **Join By** field:

   `T1.PRODID = T2.PRODID AND T2.SUPID = T3.SUPID`

4. To search all records with the string "Mango Passion", type the following string in the **Where** field:

   `T1.PRODNAME = 'Dave's Mango Passion soda'`

5. Click **Apply** to update the query fields.

   The **Query** box at the top of the **Overview** page displays the complete query statement:

   ```
   SELECT T1.ProdName, T2.Prodcost, T3.SupName
   FROM grocery/product, grocery/supprod, grocery/supplier
   WHERE (T1.PRODID=T2.PRODID AND T2.SUPID=T3.SUPID)
   AND (T1.PRODNAME='Dave's Mango Passion soda')
   ```

The resulting records are:

	ProdCost	SupName
Dave's Mango Passion soda	3.00	Kolb Bros.
Dave's Mango Passion soda	2.90	D & C Distributors
Dave's Mango Passion soda	2.85	Silves Wholesale

Example 3: Summarizing records with GROUP BY

What is the least expensive mineral water product? To answer this question, you must obtain the product costs from multiple suppliers for all mineral water products in each vendor's product line, then average the costs, group the data by vendor, and sort the data by cost.

With SQL, you can use summary records to obtain the answer by running a single query. The `PRODUCT` file includes a `ProdCat` (product category) field in which all mineral water products are identified by the value 100570100. The `ProdName` field contains the vendor name.

To transfer a summary record for each product name:

1. Specify the `GROCERY/PRODUCT`, `GROCERY/SUPPROD`, and `GROCERY/SUPPLIER` files in the **Lib/ File(Member)** field on the main file transfer screen, then open the **From AS/400 - Options** dialog box.

2. To transfer a summary record of the average cost and the product name field, type this string in the **Select** field on the **Overview** page:

   `AVG(T2.PRODCOST), T1.PRODNAME`

3. To join records by product ID and supplier ID, type the following string in the **Join By** field:

   `T1.PRODID = T2.PRODID AND T2.SUPID = T3.SUPID`

4. To search records that are mineral water products, type the following string in the **Where** field:

   `T1.PRODCAT =100570100`

5. To perform summary calculations on records of the same product name, type the following string in the **Group By** field:

```
T1.PRODNAME
```

6. To sort the results from lowest to highest product cost, type the following string in the **Order By** box:

```
AVG(T2.PRODCOST)
```

7. Click **Apply** to update the query fields.

   The **Query** box at the top of the **Overview** page displays the complete query statement:

```
SELECT AVG(T2.PRODCOST), T1.PRODNAME
FROM grocery/product, grocery/supprod, grocery/supplier
WHERE (T1.PRODID=T2.PRODID AND T2.SUPID=T3.SUPID)
AND (T1.PRODCAT='100570100')
GROUP BY T1.PRODNAME
ORDER BY AVG(T2.PRODCOST)
```

The resulting records are:

	ProdCost
Whisper Rain Mineral Water	3.00
Maple Falls Mineral Water	3.17
Mt. Baker Sparkling Water	3.35
Panda Mineral Water	3.62
Panda Sparkling Lemon Essence	3.81
Panda Sparkling Lime Essence	3.81

AS/400 Client Application

An additional feature available in Rumba Office and Rumba AS/400 Edition is the AS/400 Client Application: Rumba Submit Remote Command (application and API)

The AS/400 Client Application brings many powerful features and possibilities to your desktop. Rumba Submit Remote Command allows you to start and control non-interactive programs on your AS/400 host system without opening a display session.

The associated AS/400 Client APIs are development tools. With these APIs and a high-level programming language such as C, C++, or Visual Basic, you can develop custom PC applications which incorporate the functionality of the Rumba AS/400 Client Application. Before you can use the AS/400 Client APIs, you must install the Legacy API help and header files.

AS/400 Client installation and availability

The AS/400 Client Application is included under an Express installation of Rumba AS/400 Edition. However, you can select or deselect the application by performing a Custom installation. The AS/400 Client Application is listed under the AS/400 branch in the Select Features dialog box of the installation program.

Before starting

Before using the AS/400 Client application, you must configure an interface to connect with an AS/400 via the Rumba AS/400 Communications tool or the APPC Configuration tool. These tools reside in the Rumba Tools folder. For instructions on using these tools, see *Connection Tools*.

Note: If you have already configured an interface with the Rumba AS/400 Display or Printer, you do not need to reconfigure the interface with either of the tools.

Interfaces supported

The Rumba AS/400 Client Application requires an interface that supports full LU 6.2 capabilities. These interfaces include:

- Rumba Router (only MPTN supported).
- Microsoft SNA Server - All connection subtypes for the Microsoft SNA Server Client are supported.

    Note: The Microsoft SNA Server Client software must already be installed and configured, and the Microsoft SNA Server itself must be correctly configured and functional.
- TCP/IP - If your AS/400 operating system supports Optimized Server (V3 R1 or higher), you can connect Rumba AS/400 Client Applications over TCP/IP.

Rumba Submit Remote Command

Use the Rumba Submit Remote Command tool to start and control non-interactive programs on an AS/400 host from your PC without opening an AS/400 display session. Commands can be sent individually or together, and they can be saved in an ASCII file for re-submission. Return codes are received but program data cannot be returned from the host system.

For an example of using Submit Remote Command, imagine your accounts receivable department prints a group of reports every morning which requires them to enter a series of AS/400 commands. You can streamline the process in the following way:

1. Use the **Rumba Submit Remote Command** window to create an ASCII file of AS/400 commands.
2. Click **Send All**.

> ✏️ **Note:** If you already have a command file, open it in the **Rumba Submit Remote Command** window, then click **Send All**.

The Submit Remote Command utility also has a command line interface which allows you, for example, to execute Submit Remote Command from a PC batch file and send AS/400 commands in a specified `.txt` file.

The Rumba Submit Remote Command tool is located under **Start** > **Programs** > **Micro Focus Rumba** > **Rumba Tools**. Submit Remote Command is also a legacy API and available as an ActiveX control in the Rumba ObjectX Development Kit.

Mainframe File Transfer

This section provides a general overview of the Rumba Mainframe File Transfer program. For detailed information, refer to the online Help.

Mainframe file transfer features

Mainframe file transfer copies files between your PC and the mainframe. The **Transfer** > **Send** and **Transfer** > **Receive** commands open a setup window where you specify the file type, transfer protocol, and host environment for the file transfer. You can also choose to have Rumba software initiate the file transfer protocol.

Rumba for the mainframe works transparently with most IBM host operating systems, including CICS, TSO, and VM/CMS. This allows an application on one platform to access and use data on another platform.

Rumba simplifies the file transfer process by using typical Windows menus and dialog boxes. It reads and lists PC and host files, and lets you transfer files in the background. You can also perform file transfers using the configurable tool bar.

To start a transfer, Rumba submits a command to a host file transfer program. The most common host program used is called `IND$FILE`.

Mainframe file transfer provides a number of features, such as support for:

- ASCII or binary transfers
- Write Structured Field or Buffered data streams
- EBCDIC-to-ASCII translation
- Data compression
- DOS-based, batch, and drag-and-drop file transfer

Sending and receiving files

Transferring files between a PC and mainframe is straightforward once you have configured your PC. Unless you want to change the way you transfer files, you do not have to reconfigure your machine each time you want to send or receive files.

Configuring your PC

1. Connect to the host.
2. Click **Transfer** > **Configure** .
3. In the **File Transfer Configuration** dialog box, select **TSO**, **CMS**, or **CICS** as the destination host environment.
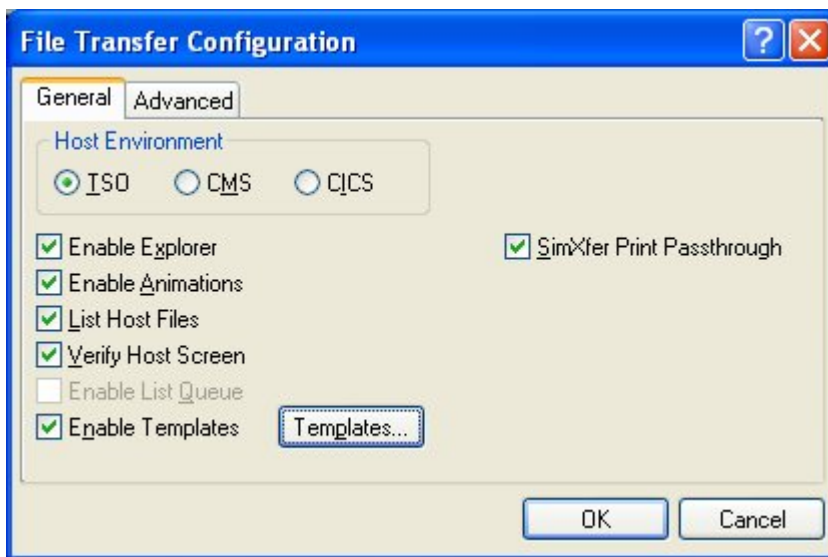
Note: If you click **Templates**, the **File Transfer Template Configuration** dialog box appears. For more information on how to set up a template, see *Using file transfer templates*.

After configuring your PC (and setting up a template, if required), you are ready to transfer files.

Sending a file to the host

1. Connect to the host.
2. Click **Transfer** > **Send** .

   The **Transfer - PC to Host** dialog box appears.
3. In the **Host File** frame, specify the required host file information. The fields vary, depending on the destination host environment.
4. In the **PC File** frame, specify the PC file to send to the host.
5. In the **Data Type** frame, click the appropriate data type options.
6. Click **Transfer** to send the file, or click **Accept** to save the settings for future use.

Receiving a file from the host:

1. Connect to the host.
2. Click **File** > **Receive** .
3. In the **Host File** frame, select the host file you to transfer to your PC.
4. In the **PC File** frame, select an existing destination file name or enter a new destination file name, and then select a PC directory.
5. In the **Data Type** frame, click the appropriate data type options.
6. Click **Transfer** to send the file, or click **Accept** to save the settings for future transfers.

Using file transfer templates

To maximize user productivity, you can use file transfer templates to set up and save frequently used file transfer settings rather than repeatedly configuring the same types of file transfer.

You may need to transfer files on a regular basis, and Rumba provides an efficient way of making the process easier. Use file transfer templates to set up and save detailed send or receive instructions for specific files. Saving these instructions as templates makes them available for repeated use. Rumba

supports DOS-type wildcard characters in your file names and extensions, increasing your ability to match larger sets of data on the host or PC with one template.

> **Tip:** When you set up a send template, it is good practice to set up a corresponding receive template that reverses the steps specified in the send template.

Creating a TSO template

To create a TSO file transfer template:

1. Open the **File Transfer Configuration** dialog box.
2. In the **Host Environment** frame, select **TSO**.
3. Ensure that **Enable Templates** is checked.
4. Click **Templates**.

   The **TSO File Transfer Template Configuration** dialog box appears.
5. Click **Send** or **Receive**, depending on the type of template you are creating.
6. In the **Name Conversion**frame, you can either modify a default template or create a new one. If you type a new PC file name, Rumba saves a new template with that name. If you do not change the PC file name, but modify its template settings, Rumba saves the settings changes in the default template.
7. When you have finished, click **OK**.

The template is now available when you perform a TSO file transfer operation.

Creating a CMS template

To create a CMS file transfer template:

1. Open the **File Transfer Configuration** dialog box.
2. In the **Host Environment** frame, select **CMS**.
3. Ensure that **Enable Templates** is checked.
4. Click the **Templates** button.

   The **CMS File Transfer Template Configuration** dialog box appears.
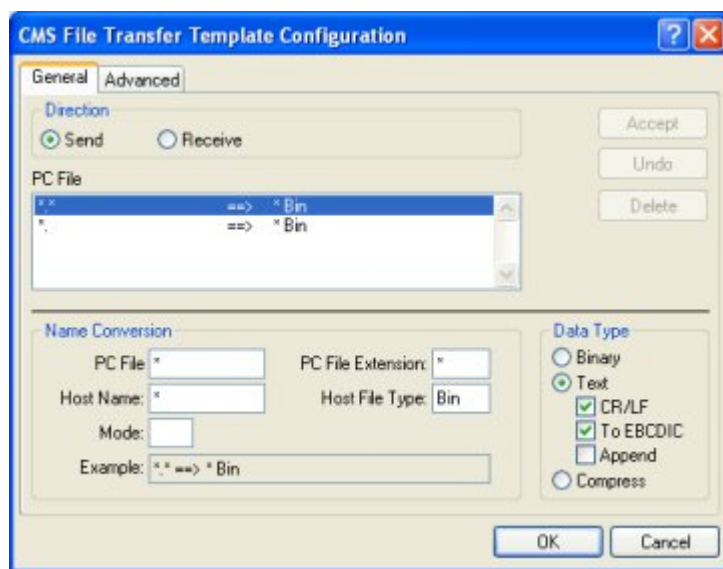
5. Click **Send** or **Receive**, depending on the type of template you are creating.

6. In the **Name Conversion** frame, you can either modify a default template or create a new one. If you type a new PC file name, Rumba saves a new template with that name. If you do not change the PC file name, but modify its template settings, Rumba saves the settings changes in the default template.

7. When you have finished, click **OK**.

The template is now available when you perform a CMS file transfer operation.

Creating a CICS template

To create a CICS file transfer template:

1. Configure a file transfer.
2. In the **Host Environment** frame, select **CICS**.
3. Ensure that **Enable Templates** is checked.
4. Click **Templates**.

    The **CICS File Transfer Template Configuration** dialog box appears.
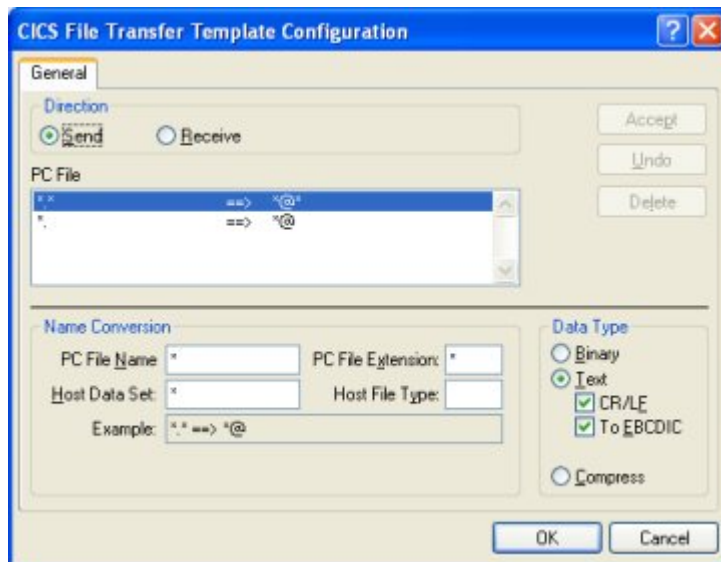
5. Click **Send** or **Receive**, depending on the type of template you are creating.
6. In the **Name Conversion** frame, you can either modify a default template or create a new one. If you type a new PC file name, Rumba saves a new template with that name. If you do not change the PC file name, but modify its template settings, Rumba saves the settings changes in the default template.
7. When you have finished, click **OK**.

The template is now available when you perform a CICS file transfer operation.

Batch file transfers

You can use batch files to transfer several files between your PC and the mainframe host. A batch file consists of two main components: a batch list, and send or receive commands. The batch list groups send and receive commands that you can run from your batch file. You can have several batch lists within one batch file, and several commands within each batch list.

If you only need one batch file, you can use the default batch file DEFAULT.FTB. You can also use drag and drop for multiple file transfers.

Creating a batch file

To create a batch file:

1. Select **Transfer** > **Batch** .

   If no batch files are defined, a **New Batch List** dialog box appears. Otherwise, in the **File Transfer Batch** dialog box, click **New** in the **Batch File** frame.

2. Specify a name for the new batch file, then click **OK.**

   The new batch file name appears in the **Name** list.
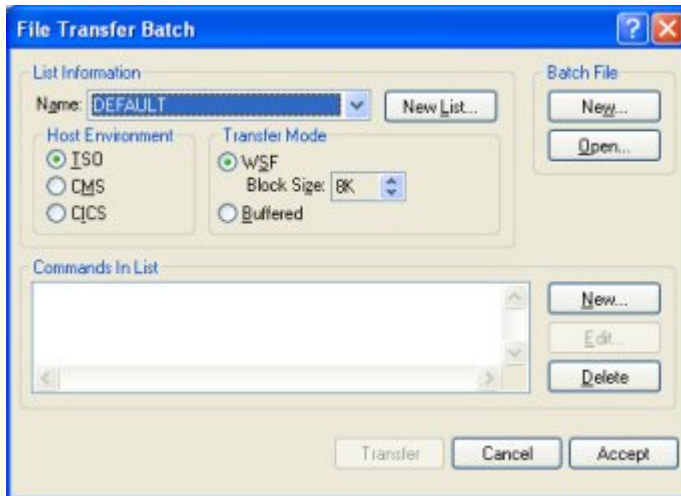
3. In the **Commands In List** frame, click **New**.

   The **File Transfer** dialog box appears.
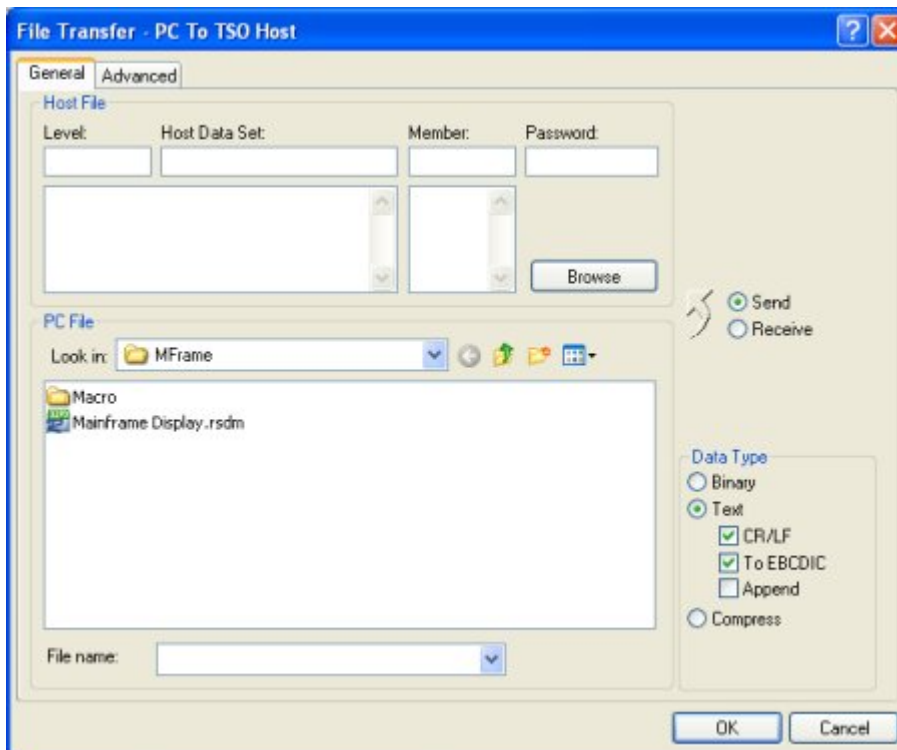
4. Click **Send** to send files to the host, or **Receive** to receive files from the host.

5. Select the files to transfer, then click **OK**. You can use file extensions and wild cards to transfer multiple files with a single command.

6. Repeat steps 3 - 5 to add transfer commands as needed.

7. Click **Accept** to save your settings, or **Transfer** to save your settings and run the batch file transfer.

Running a batch file transfer

To run a batch file transfer:

1. Open your Rumba profile and connect to the host.

2. Click **Transfer** > **Batch** .

3. Select the batch file you want to use. Batch files have an .ftb extension.

4. Click **Open.**
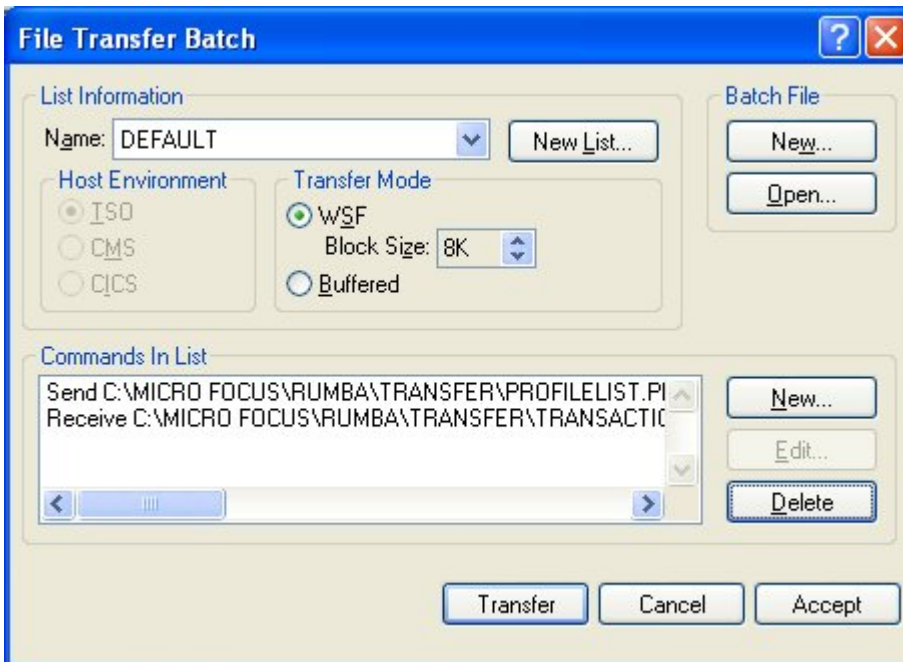
5. To run individual commands in a batch file, select the command you want to start, then click **Transfer**.

6. To run all commands in a batch file, click **Transfer**.

File transfer registry settings

When you install an upgrade version of Rumba for the Mainframe over an existing 16-bit version, the installation program converts settings that were previously in the rumba.ini file into Windows registry entries

The table below lists the old file transfer rumba.ini settings, the new Windows Registry entry, and describes the purpose of each setting.

This rumba.ini setting	Converts to this Windows registry entry	Purpose
[FTX] Blank Screen	BlankScreen (DWORD)	Specifies whether the screen should be blank while Rumba displays the **File Transfer Configuration** dialog box. When the dialog box first appears, the screen is frozen. When

This rumba.ini setting	Converts to this Windows registry entry	Purpose
		the dialog box closes, the screen refreshes with the current buffer contents.
[FTX] NOPA1	NOPA1 (DWORD)	Disables the PA1 key from being automatically sent to the host following a TSO file transfer or file list command. Some host applications, such as session managers, reserve the PA1 key for special functions. Enabling the NOPA1 setting prevents Rumba from interfering with these host applications. NOPA1=0 means disabled. NOPA1=1 means enabled. **Note:** For compatibility with older versions of Rumba, the value NOPA1=2 is supported as an alternative for the ListcatWithATTN setting.
[FTX] ListcatWith ATTN	ListcatWith ATTN (DWORD)	Overrides the default behavior of sending a PA1 following a LISTCAT command. Rumba issues the LISTCAT command internally to generate the file list in the **File Transfer** dialog box. When ListcatWithATTN is specified, the LISTCAT command is terminated by sending an Attention key, rather than a PA1, to the host. To disable the setting, assign the value 0. To enable the setting, assign the value 1.
[FTX]IND_Timeout	IND_Timeout (DWORD)	Adjusts file transfer responsiveness by specifying the amount of time to wait before timing out during a file transfer command. The setting specifies the timeout in milliseconds.
[FTX] List_Timeout	List_Timeout (DWORD)	Adjusts file transfer responsiveness by specifying the amount of time to wait before timing out during a file list command. The setting specifies the timeout in milliseconds.

Using macros to transfer files

You can use the Rumba Macro Editor to create macros to automate and simplify file transfer procedures for your users.

Creating a file transfer macro

To create a file transfer macro:

1. Select **Tools** > **Edit Macro** .

2. On the lower toolbar, click the **File Transfer Send** icon or the **File Transfer Receive** icon depending on whether you want to send or receive a file.

3. Configure the file transfer as needed, then click **OK.**

4. On the Macro Editor toolbar, select **File** > **Save**.

5. Give the macro a name, then click **OK**.

IND$FILE TRANSxx codes

The most common host file transfer program is IBM's `IND$FILE`. This program uses a set of TRANSxx codes, which convey information about particular key file transfer transactions. The table below describes the various codes used by the `IND$FILE` program.

TRANSXX code	Descriiption
`TRANS00 Error in file transfer: file transfer canceled`	Indicates that an error occurred in the file transfer operation that was detected by the CICS file transfer transaction. This may be an error in the data being transferred, or an unidentified system error. Examine the file for incorrect format or bad data, and check the installation procedure and the setup of all the components involved. Try the file transfer again.
`TRANS01 File Transfer command being processed`	Informs you that the file transfer transaction is currently being processed.
`TRANS02 Number of bytes of file transferred so far ===> xxxx`	The file transfer transaction is currently being processed, and this code informs you of the number of bytes of information transferred so far.
`TRANS03 File transfer complete`	The file transfer transaction has completed normally and the file has been transferred without any errors detected. Examine the data in the file to determine if the operation was implemented correctly.
`TRANS04 File transfer complete, with records segmented`	The file transfer transaction completed successfully, but at least one of the records of data sent to the host was longer than the maximum allowed (32767). The record has been broken into more than one segment, each of which appears as a separate item in the Temporary Storage queue. The error may have occurred because the CRLF option was used when the file did not contain logical records.
	Check the contents of the file being uploaded, and the options used, then try the file transfer transaction again. Alternatively, use the
	`BINARY`
	option to transfer the data "as is" from the 3270 to the host without interpretation. Interpret the data in the host.
`TRANS05 Personal computer filespec incorrect:`	File transfer has been canceled. The PC file specified is incorrect. Verify the file and try the file transfer transaction again.
`TRANS06 Command incomplete:`	File transfer canceled. The file transfer command entered was incomplete. Verify the command and try the file transfer transaction again.

TRANSXX code	Descriiption
TRANS07 Cannot link to host:	File transfer canceled A link cannot be established with the host. Check the status of the host, or refer to the user for more information.
TRANS09 Error reading file from damaged personal computer disk: file transfer canceled	The specified PC file resides on a damaged disk. Copy the file to an undamaged medium and try the file transfer transaction again. .
TRANS10 Host has not responded within timeout period	The host did not respond within the allotted amount of time. Check the status of the host, or refer to the user guide for more information.
TRANS11 Lost contact with host: file transfer canceled	Re-establish a host connection and try the file transfer transaction again.
TRANS12 Error writing to damaged or full personal computer disk: file transfer canceled	The specified PC disk is damaged or full. Examine the disk for damage and ensure that it contains enough space to receive the transferred data. Try the file transfer transaction again.
TRANS13 Error writing file to host: file transfer canceled	An error occurred while writing the file to the host. Refer to the user's guide for more information.
TRANS14 Error reading file from host: file transfer canceled	An error occurred while reading the file from the host. Refer to the user guide for more information.
TRANS15 Host storage unavailable: file transfer canceled	The CICS file transfer transaction could not obtain enough storage in which to place its own control blocks, I/O buffers, or save areas. Make more storage available to the transaction and try the operation again.
TRANS16 Incorrect request code: file transfer canceled	The command that started the file transfer operation did not specify GET or PUT. The command is rejected. Check the parameters used to invoke the file transfer operation and correct them. Re-attempt the File Transfer transaction.
TRANS17 Invalid file name: file transfer canceled	The file name specified is invalid. Verify the file name and try the file transfer transaction again.
TRANS18 Incorrect option specified: file transfer canceled	An incorrect file transfer option was specified. Verify all options selected and try the file transfer transaction again.
TRANS19 Error handling host file: file transfer canceled	An error occurred while handling the host file. Refer to the user guide for more information.
TRANS21 Not enough personal computer memory available: file transfer canceled	Not enough memory is available on your PC to proceed with the file transfer. Close one or more applications and try the file transfer transaction again.
TRANS22 Host session identifier incorrect: file transfer canceled	The host session identifier specified is incorrect. Verify the identifier and try the file transfer transaction again.
TRANS23 Activity specified not a host session: file transfer canceled	The activity specified is not a host session. Verify the activity and try the file transfer transaction again.
TRANS24	Autokey operation in progress: file transfer canceled Wait for the autokey operation to complete, then try the file transfer transaction again.

TRANSXX code	Descriiption
`TRANS25 Keyboard inhibited: file transfer canceled`	The keyboard has been inhibited. Refer to the user guide for more information.
`TRANS26 Unrecoverable system error: file transfer canceled`	An unrecoverable system error occurred. Check the installation procedure and the setup of all the components involved. Try the file transfer transaction again.
`TRANS27 Communication sequence with host disrupted: file transfer canceled`	Communication between your PC and the host was disrupted. Re-establish a host connection and try the file transfer transaction again.

HP and UNIX File Transfer

Rumba software, the PC-to-host connectivity system, gives you easy, complete, and transparent access to host applications. PC users working in the Microsoft Windows environment, and using Rumba HP and UNIX software, can communicate with HP and UNIX hosts and host applications. Rumba software combines the convenience, ease of use, and versatility of PCs with the power of HP and UNIX hosts and host applications.

How to transfer files using Rumba HP and UNIX

Using the Rumba HP file transfer feature, you can copy files from a PC to an HP host or from an HP host to a PC.

Similarly, using the Rumba UNIX file transfer feature, you can copy files from a PC to a UNIX or VMS host, or from a UNIX or VMS host to a PC. Rumba software works transparently with any UNIX or VMS host operating system.

Rumba software simplifies the file transfer setup process by using typical Windows menus and dialog boxes. It reads and lists PC and host files, and lets you transfer files in the background. You can also perform file transfers using the configurable toolbar.

Understanding the file transfer process

Rumba software uses protocol-driven transfer methods. This means that files are transferred using a set of rules known as a protocol. The specific details of the protocols are not important to users, but the protocols specified on the PC must match that used by the host computer. For example, Rumba software will not work if the host is attempting to transfer using the Kermit file transfer protocol and the user selects Text as the transfer protocol to be used. To minimize such conflicts, Micro Focus strongly recommends that users take advantage of the Rumba Initiated and Macro options.

Even using Rumba Initiated, the user must know which protocols are supported by the host. If the protocol chosen by the user is not supported by the host, the Rumba software will not work. Also, the host must be at a command prompt, ready to accept the transfer command before you attempt a file transfer.

Here is an overview of the steps you must follow to transfer files:

1. On the Home tab, click Send or Receive. (In the Classic interface, click **Transfer** > **Send** to send a file from your PC to the host, or **Transfer** > **Receive** to receive a file from the host.)
2. On the **Setup** tab, specify the file type, transfer protocol, and host environment. You can also have Rumba software initiate the file transfer operation by checking the **Rumba Initiated** box. For example, "sz" is used for sending files to the host via Zmodem.

   When this option is checked, you may specify the name of the host's transfer program which Rumba software will use when automatically initiating the transfer.

   If Rumba Initiated is selected, the software provides some standard commands based on the protocol selected. If the local setup is different, commands can be entered that are specific to the site. Whatever is entered in the To Host Program Name or From Host Program Name fields is sent to the host exactly as it is typed.

   📝 **Note:** If the host is a VMS system and the full program name is given, Rumba software defines a foreign command to start the program.

   The default setting for Rumba Initiated is Off. The Rumba Initiated option is not available for Text file transfers or when Other is selected in the Host Environment box.

When you choose a file transfer protocol, Rumba software displays a configuration tab corresponding to that protocol. For example, if you choose the Text transfer protocol, the Text Configuration tab is displayed.

3. Click the tab that is associated with the transfer protocol you have chosen, and then specify the information needed by the transfer protocol.

4. Click the **Files to Transfer** tab and select the files you want to transfer.

> ✎ **Note:** If Rumba Initiated is checked, and the correct host is not specified, you will not see any host file information when you view the **Files to Transfer** tab.

5. Click **OK**.

MPE file transfers

For Rumba HP only, file transfer capabilities are now available for systems using the MPE host environment. While you're connected in MPE mode, HPXFER is the only protocol available. MPE file transfers are always Rumba Initiated.

MPE file transfers require special setup by the host administrator. Follow these steps to set up MPE file transfers:

1. In the registry, create a new value:

   HKEY_LOCAL_MACHINE\SOFTWARE\WallData\

   Office\HP\MPEBOOTSTRAP

   The type of the value is DWORD and the value should be set to 1.

2. When the File Transfer dialog is invoked in Rumba software, a new button labeled **BootStrap** will be visible. Clicking this button will cause the WALLXFER.BIN file to be transferred to the MPE system.

3. After the WALLXFER.BIN file is on the host, the administrator should set its file permissions to execute ("X") and move it to the PUB.SYS group.

4. Once the file has been transferred for the first time, the BootStrap functionality should be disabled by deleting the registry key or changing its value to 0.

> ✎ **Note:** This procedure needs to be done once for each MPE system with which you will be transferring files.

Using compatible file transfer protocol versions

Rumba software may have trouble supporting certain freeware or shareware versions of a given file transfer protocol. For example, there are many public domain versions of Xmodem and Ymodem which are known to improperly implement the protocols. Make sure that you are using a file transfer protocol version that is current and well-supported by a known and reputable vendor.

8-bit binary transparent communications

It is necessary to perform file transfers using an 8-bit "binary transparent" protocol. A transparent channel is a communication method and medium that does not use any of the byte values for control of the channel itself. An 8-bit channel does not use any of the bits in an 8-bit character set for control of the channel (for parity, for example) and can transmit all 256 codes in the character set. An example of a channel that does not meet this criteria is a CompuServe" connection using 7 bits and Even parity. This is important when transferring data that contains anything that is not printable text in the data stream. If the channel you are using is not transparent, the Kermit protocol performs the most reliable file transfer; channels less than 8-bits wide are not supported.

Using Rumba macros to transfer files

Because many Rumba users are not familiar with file transfer procedures, it's a good idea to configure macros they can use for the file transfers they commonly perform. The Rumba macro utility allows you to automate many Rumba functions, including file transfers.

To create a file transfer macro:

1. Select **Tools** > **Edit Macro**.
2. Click **file transfer send** or **file transfer receive**.
3. Configure the file transfer and click **OK**.
4. Select **File** > **Save**, then give the macro a name and click **OK.**

AS/400 and Mainframe Printers

This section provides a general overview of Rumba for both the AS/400 and Mainframe printer.

AS/400 and mainframe printer features

With the Rumba printer, you can route jobs from your printer queue to the Windows printer of your choice. You can also:

- Preview documents before printing
- Save documents to print later
- Print to a file
- Change page layout, formatting, and printer properties
- Use the PC Printer Queues display, and the AS/400 Printer Queues display, to monitor and control printers and print jobs
- Use the Print Log display to monitor completed print jobs
- Use advanced printer features such as font mapping and character translation
- Install the Rumba AS/400 printer as a Windows system service

Printing AS/400 and mainframe documents on a PC printer

When you have configured and connected the Rumba printer, you can send host print jobs to your current Windows printer by simply opening the host application, and then printing the document as usual. You can also preview documents and save them to print later, print to a file, or change the page layout, formatting, and printer properties.

If you attempt a Rumba printer operation, but have not installed a Windows printer, the Printer Wizard automatically prompts you to install a Windows printer, and guides you through the installation process. Normally, you add printers by clicking the Printers icon in your Windows Control Panel.

Viewing documents before printing

You can use Print Preview to review page layout and formatting, change Windows printer properties, or save documents to print later. When Print Preview is active, a picture of each printed document displays on the screen before the document is sent to the printer.

Monitoring and controlling printers and print jobs

You can control the behavior of the Rumba mainframe printer by using Direct printing, Printer sharing, and End print job. You can also monitor your printers and print jobs with the Printer Queues display, and use the PC Printer Queues display to:

- Monitor and control current print jobs
- Control your Windows printers

AS/400 Printer Queues

The AS/400 Printer Queues feature is available only on hosts with operating systems of OS/400 V3R1 or later. This display lists the AS/400 printer queues to which you have access, along with the print jobs in each printer queue, and the status of each job.

You can:

- Customize the display to list only the print jobs you are interested in.
- Toggle this display on and off by choosing AS/400 Printer Queues on the View menu.

> **Note:** AS/400 Printer Queues will not work if the transaction program (TP) on the host has not been started. The command to start all host servers is: `strhostsvr *all` The command to start the Network Print Server TP is: `strhostsvr *netprt`

Direct printing

With Direct printing, you can print both unlimited characters per line, and graphic data streams without interpretation by the printer driver. Host data is translated from EBCDIC to ASCII and sent directly to the printer, bypassing the printer driver.

> **Note:** When Direct printing is active, Print Preview does not work. The print job will go to the printer without any preview.

Printer sharing and End print job

Use the Printer sharing option (default) or the End print job option to specify how Rumba software will recognize the end of a host print job.

With Printer sharing, the current print job is released when Rumba software receives the Change Direction Indicator (CDI) that is usually sent with the SNA End Bracket command to indicate the end of a print job.

You can use Printer sharing for most host print applications. However, some host applications do not send an Unbind command to signal the end of the print job. If you encounter pagination problems, or if host print jobs terminate before document printing is complete, then use the End print job option.

With End print job, if the host does not send an Unbind command at the end of the print job, Rumba software will wait for the user-specified timeout period, and then release the print job if data is not received.

PC Printer Queues

This display lists the printer queues for all network and local printers set up in your Windows Print Manager. Information is displayed for all print jobs (not only jobs from the host) and for all users. See the online help for procedures on using this display to:

- Pause, resume, restart, cancel, and purge print jobs
- Add, delete, and change printers
- Change Windows printer properties

Viewing print job history

The Print Log display lists documents that have printed on the PC printer, including documents printed from host as well as from PC applications, saved print previews, documents that did not print due to errors, and held or canceled documents. Information displayed includes:

- Name of the host system

- Device name of the PC printer
- PC printer type
- Status of the print job, or location if the print job was saved
- Number of pages
- Date and time the job was processed

Formatting printed documents

The way a printed document looks is affected by the capabilities of the target PC printer, the printer driver, Rumba software formatting options, and host print file values. You can use the format specified by the host, the default Windows format for the PC printer (LU1 print jobs only), or you can customize the format by specifying options such as text styles, page orientation, scaling, margins, spacing, and print quality.

Host Print Transform

The Host Print Transform feature is available only on hosts with operating systems of OS/400 V2R3 or later. Host Print Transform changes the AS/400 print data to the ASCII format needed by a particular PC printer before the data is sent to the Rumba AS/400 printer. The Rumba AS/400 printer then passes the data directly to the PC printer. In this way, the Rumba printer and Windows printer drivers are minimally involved in the process.

To use Host Print Transform, follow these steps before connecting the Rumba AS/400 printer:

1. From the **Connection** menu, choose **Configure**, then select **Rumba Router** from the list of installed interfaces.
2. On the **Rumba Router** tab, click the **Transform** button.
3. Click the **Host Transform Enabled** option, then change the configuration parameters as needed.

You can also configure Host Print Transform directly on the AS/400 host.

Font mapping

When a host document is sent to the PC printer, it includes both text and format information. The format information includes the font size and style of the printed document. When the document prints, the closest matching font is selected from those available on the PC printer.

From the **Options** menu, select **Font Mapping**.

From the **Font Mapping** dialog box, you can select a specific Windows font, type size, and style for each font sent from the host. You can also:

- Create multiple font map files and associate each with a different profile, to be used with different printers
- Use only PC printer fonts, to speed printing

Character translation

When printing host documents on a PC printer, host EBCDIC characters are translated to ANSI characters. Each national language has its own unique character translation table. By default, your character translation table is associated with the Host Character Set you selected when configuring your printer profile.

If your application requires that characters be translated differently than the default table, you can create a customized translation table to meet your needs. The customized table is saved in association with your printer profile. You can also create multiple customized translation tables to use with different profiles.

Printer Session Manager

Printer Session Manager is an alternative tool to the Rumba mainframe printer.

The tool is designed to run as a Windows service. It manages printer queues for a group of session profiles saved previously to a dedicated folder.

The Printer Session Manager:

- Does not have a user interface.
- Reports events into Windows Event Viewer under **Application and Services Logs** > **Micro Focus Printer Session Manager**. Each event displays session profile name in the Source column.
- Can be managed from the command line.

Using the command line

The Printer Session Manager can be managed from the command line

Syntax

```
PrinterService.exe [-Service] [-UnregServer] [-Auto] [-Account [domain/user]]
[-Pwd [password]] [-Profiles [profile_directory]] [-Help]
```

Parameters

Parameter	Description
-Service	Register service. [1]
-UnregServer	Unregister service. [1]
-Auto	The service is started automatically by the service control manager when the system starts. Used with -Service.
-Account [domain/user]	The name of the account under which the service should run. Used with -Service and -Pwd. [2] [3]
-Pwd [password]	The password to the account name specified by the account parameter. Used with -Service and -Account. [2]]
-Profiles [profile_directory]	The directory containing mainframe printer session profiles. [1]
-Help	Displays the Help.

🖉 **Notes:**

1. Administrator rights are required.
2. Failed if logon failed for the provided account and password.
3. The account must be granted "Log on as a service" rights:

   http://technet.microsoft.com/en-us/library/cc794944(v=ws.10).aspx

   http://support.microsoft.com/kb/266280

Examples

Register a service

A service is registered to start manually with Logon as local system account:

```
PrinterService.exe -Service
```

Register a service to start automatically

A service is registered to start automatically with Logon as local system account:

```
PrinterService.exe -Service -Auto
```

Register a service to start automatically with Logon Account of local user Administrator with password qwerty

A service is registered to start automatically with Logon as local administrator:

```
PrinterService.exe -Service -Auto -Account .\ Administrator -Pwd qwerty
```

Register a service to start automatically with Logon Account of user Alex from domain Corpdom with password 123456

A service is registered to start automatically with Logon as Domain\Alex:

```
PrinterService.exe -Service -Auto -Account Domain\Alex -Pwd 123456
```

Unregister a service

```
PrinterService.exe -UnregServer
```

Set the session profiles directory to C:\Working Directory\MFProfiles

```
PrinterService.exe –Profiles "C:\Working Directory\ MFProfiles"
```

Register a service to start automatically with Logon Account of user Alex from domain Domain with password 123456 and set the profile directory C:\Working Directory\ MFProfiles

```
PrinterService.exe -Service -Auto -Account Domain\Alex -Pwd 123456 -
Profiles "C:\Working Directory\ MFProfiles"
```

Using the Printer Session Manager

1. Create profiles for mainframe printer sessions and save them in a dedicated folder. For example:

   ```
   C:\Working Directory\ MFProfiles
   ```

2. Register Printer Session Manager with one of the command lines, containing the following options:

   ```
   –Service
   –Account
   –Pwd
   –Profiles
   ```

3. Start the service **Micro Focus** from the Windows Services Management Console.

   When it starts, the Printer Session Manager launches an instance of the Mainframe Printer (without interface) for each profile it discovers in the dedicated folder. In this way, the Printer Session Manager allows you to print mainframe files without explicitly running Rumba Mainframe Printer instances.

Rumba Telnet Edition

Rumba Telnet Edition represents the total TCP/IP solution. Rumba software supports every TCP/IP network application: connections to multiple hosts, printing, file transfer and file system sharing, and working in the Internet.

This section introduces you to Rumba Telnet Edition and provides an overview of core Rumba Telnet Edition capabilities.

Rumba applications for TCP/IP networks

The Telnet edition of Rumba supports every TCP/IP network application: connections to multiple hosts, printing, file transfer and file system sharing, and working over the Internet. The following tables describe the functions performed by the Rumba host applications (AS/400, Mainframe, UNIX, and HP).

For AS/400 hosts

With the Telnet edition of Rumba you can view AS/400 host applications using two different methods, print host applications, perform a variety of file transfers, and seamlessly share AS/400-based file systems with workstations.

The following provides brief descriptions of Rumba AS/400 host applications grouped by function.

- Display

  Rumba AS/400 Display presents the traditional AS/400 display in a Microsoft Windows application: the Rumba window. Includes support for the Text-Assist function of OfficeVision/400.

  Users can run multiple host sessions, displaying and working with host applications simultaneously.
- Printing

  Rumba AS/400 Printer (3812 emulation) enables you to route jobs from your AS/400 printer queue to a Windows printer of your choice.

For mainframe hosts

With Rumba Telnet Edition, you can view and print mainframe host applications using two different methods, perform a variety of file transfers, and seamlessly share mainframe-based file systems with workstations.

The following provides brief descriptions of Rumba applications grouped by function.

- Display

  Rumba Mainframe Display presents the traditional mainframe display (3278/79) in a Microsoft Windows application: the Rumba window. Includes GDDM.

  Users can run multiple host sessions, displaying and working with PC and host applications simultaneously.
- Printing

  Rumba Mainframe Printer (3287 emulation) presents the traditional IBM 3287 printer functionality in a Microsoft Windows application.

- File transfer

   Rumba Mainframe Display supports many host file transfer programs including IND$FILE.

   Supports ASCII or binary transfers, Write Structured Field or buffered data streams, EBCDIC-to-ASCII translation, and data compression.

For UNIX hosts

With the Telnet edition of Rumba you can view and print UNIX host applications using two different methods, perform a variety of file transfers, and seamlessly share UNIX-based file systems with workstations.

The following provides brief descriptions of Rumba applications grouped by function.

- Display

   Rumba UNIX Display presents the traditional VT display in a Microsoft Windows application supporting the whole range of VT terminals from VT 52 to VT 420 as well as VT340 ReGIS, Sixel, and Tektronix 4010/4014 graphics support.

   Users can run multiple host sessions, displaying and working with host applications simultaneously.
- Printing

   Rumba for UNIX Display/Terminal-routed printing prints the display screen, selected text, and full color graphics, or routes print jobs from the host printer through your Windows print driver.
- File transfer

   Rumba for UNIX Display supports Kermit, Text, and X-, Y-, Z-Modem file transfer protocols.

For HP hosts

With Rumba Telnet Edition you can view HP host applications, print HP applications using two different methods, perform a variety of file transfers, and seamlessly share HP-based file systems with workstations.

The following provides brief descriptions of Rumba applications grouped by function.

- Display

   Rumba for HP Display presents the traditional HP terminal types in a Microsoft Windows application.

   Users can run multiple host sessions, displaying and working with host applications simultaneously.
- File transfer and file system sharing

   Rumba for HP Display supports Kermit, Text, and X-, Y-, Z-Modem file transfer protocols for UNIX operating systems.

   MPE native file transfer provides a client server file transfer solution allowing users to transfer files to or from a HP 3000 using a proprietary Z-modem based protocol. The server side application can be downloaded to the host via the Rumba host link.

Getting connected

The procedure for connecting to a remote host or source depends on which application you use. Refer to the *Connecting to a host* section of the help for information on configuring a connection.

If you plan to use:	Reference
Rumba AS/400 Display	Use the Connection Configuration dialog box to configure a Rumba Router TN5250 host link.
Rumba Mainframe Display or Printer	Use the Connection Configuration dialog box to configure the TN3270E interface.
Rumba UNIX Display	Use the Connection Configuration dialog box to configure an interface.
Rumba HP Display	Use the Connection Configuration dialog box to configure the appropriate interface.

Automating Rumba

By automating complex and common tasks, both you and your users can make Rumba do more for you. You can use either macros or scripts to simplify your work. Macros record a series of actions, such as keystrokes and commands to automate tasks.

If you choose to install the separate Script Editor you can use scripts to automatically manipulate sessions. For example, you could write a script to open a new session and connect to a host.

This section provides you with background information on using macros. Detailed information on the Rumba Script Editor can be found in its online help.

> **Note:** One of the easiest and most effective ways to familiarize yourself with Rumba Script Editor/ Player capabilities is to run the demo scripts provided with this application. Run one of the script files (such as, demo.csf) to see how a script works. You can also copy a sample script file and edit it to suit your needs.

Use macros in conjunction with hotspots, custom toolbar buttons, and keyboard shortcuts to complete tedious log-on procedures and other common tasks with a single mouse click or key stroke.

For example, distributing macro files and Hotspot libraries to your users provides them with valuable production tools that they do not even have to take the time to create.

Creating macros

Macros record a series of actions, such as keystrokes and commands. Once you record a macro, you can play it back to automate routine tasks. For example, with the click of a button, a macro can connect your PC to a host, test the return strings, and then start an application.

There are two ways to create a macro:

- Choose **Record Macro** from the **Tools** menu or click the icon, and then perform your keystrokes and commands. Be sure to save the sequence as a macro file after recording.
- Choose **Edit Macro** from the **Tools** menu, and then build a sequence of commands with the Macro Editor.

To combine the speed of recording macros with the power of written command lines, record the basics, and then use the Macro Editor to edit and add to the macro sequence. For details about macro commands, see *Macro Commands*.

Connect and disconnect macros

Connect macros run automatically each time you connect to a host. For example, the connect macro could contain commands to enter a User ID and password, choose menu items, and open a host application. You can distribute to your users macros that automatically log them on and take them to the correct host application.

In order for a connect macro to work, the communications interface must be properly configured, and the macro must be saved from within the profile with which it is to be used.

Disconnect macros run automatically each time you disconnect from the host. For example, the disconnect macro could run another program on the connected host before logging out. It is important to note that, if you have a disconnect macro defined, you do not disconnect from the host until after the macro has been performed. If you are disconnected from the host because the physical connection has been broken, then the disconnect macro does not run.

Specifying connect and disconnect macros

Any given profile can have only one connect and one disconnect macro. To specify that a macro automatically runs upon connecting or disconnecting:

1. From the **Tools** menu, choose **Edit Macro**. This opens the Macro Editor dialog box.
2. Select a macro from the **Macro Name** box.
3. Under the **Play Macro On** area, check either the **Connect** or **Disconnect** box.
4. Select **Save** from the **File** menu to save the macro, and close the Macro Editor.
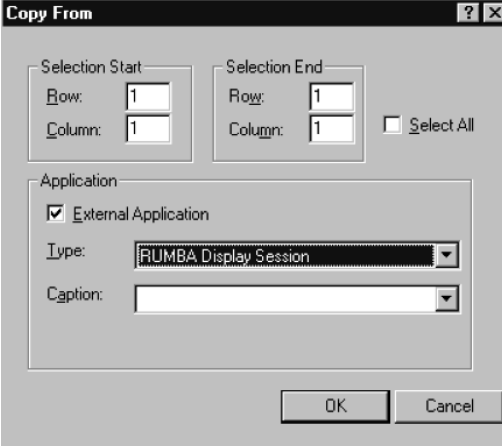
Rumba Macro Editor

Recording keystrokes and commands may not always produce the macros you need to accomplish complex tasks. When it does not, use the Rumba Macro Editor to work with all the macro commands and create complex macros using advanced macro features. *Macro Commands* shows the various commands available for each host.

You cannot record many of the macro commands, such as Pause Macro, Goto Label, Compare Values, Message, and Prompt, because they are not commands or keystrokes. For example, you may need a macro that pauses between two macro commands to give the host enough time to respond.

Copying data between applications

You can use Copy From and Paste To commands in the Macro Editor to designate an area of a display screen to be copied from one application and pasted into another application. In order to do this you need to specify the application and the location (in terms of rows and columns) from which the data should be copied, and the application and location to which it should be pasted.

> **Note:** If you select Microsoft Excel 97 as the application type when using either the Copy From or Paste To macro, the dialog box changes and you must provide additional information in the File and Worksheet fields.

Using variables

Variables are one of the most powerful macro features. Using variables is like writing your own macro-driven application. With variables, you can create interactive macros that respond to user input and perform specific sets of commands.

To open the **Variable Setup** dialog box:

1. Select **Tools** > **Edit Macro** to open the **Macro Editor** dialog box.
2. Right-click in the macro display area.
3. Choose **Variable Setup** from the popup menu.

In the **Variable Setup** dialog box, you can select a variable, such as a password or user ID, and define its value based on host screen information, static text, and user input. The variable values are available for the Type, Message, Compare Values, Goto Label, and Prompt commands.

For example, you could create a prompt dialog box to request a password from a user, and then save it as a variable. The variable containing the user's password is then available for any Type, Compare Value, and Message commands you have included in the macro sequence. For example, you could use a Compare Values command to check the password, and then end the macro or even disconnect the PC from the host

Example

The following example shows you how to define a screen variable and use it to execute specific actions based on a variable returned by PROFS.

This is a long example, which contains examples of creating several different kinds of macro commands. If you are unfamiliar with macros, we recommend that you create the macro contained in the example to familiarize yourself with the macro editor and macro command syntax.

1. Select **Tools** > **Edit Macro**.
2. Name the Macro *Flow Example*.
3. Right-click in the macro display area, and choose **Variable Setup** from the pop-up menu to open the **Variable Setup** dialog box. Click the **Variables** list box, then choose **Variable** for screen text.
4. From the **Variable Types** box, choose **Screen Variable**, if it is not already selected.
5. Click the **Screen Variable** tab.
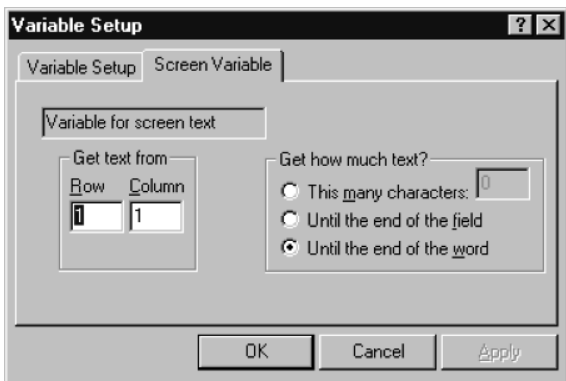
6. Change the **Get text from** Column value to 78.
7. In the **Get how much text?** area, click **This many characters**, and type 3 in the box.
8. Click **OK** to save the settings.

   This gets 3 characters from the screen at row 1, column 78 and saves them as a variable that your macro can use.

The following steps reference buttons on the toolbar above the macro display area.

1. Open the **Goto Label** dialog box. Select **Variable** for screen text from the label list, and click **OK.**
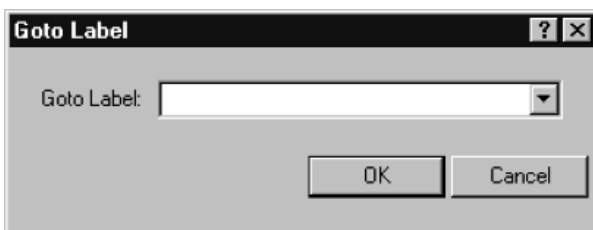
2. Open the **Message** dialog box. Type `Screen Unknown` and click **OK**.
3. Open the **Goto Label** dialog box. Type `Macro End` and click **OK.**
4. Open the **Label** dialog box. Type `A00` and click **OK.**
5. Open the **Message** dialog box. Type `PROFS Main Screen` and click **OK.**
6. Open the **Receive** dialog box.
7. Type `Mail Waiting` in the Find text box.
8. Under **Screen Position**, click **Only at row, column**, and type `24` and `69` in the **Row** and **Column** boxes.
9. Under **Timeout**, click **Seconds**, and type `30` in the box.
10. In the **Otherwise** box, type `NoMail`, and click **OK.**
11. Open the **Message** dialog box. Type `You have Mail` and click `OK`.
12. Open the **Label** dialog box. Type `NoMail` and click `OK`.
13. Open the **Goto Label** dialog box. Type `Macro End` and click `OK`.
14. Open the **Label** dialog box. Type `C00` and click `OK`.
15. Open the **Message** dialog box. Type `You are on the Open Mail Screen` and click `OK`.
16. Open the **Goto Label** dialog box. Type `Macro End` and click `OK`.
17. Open the **Label** dialog box. Type `E01` and click `OK`.
18. Open the **Message** dialog box. Type `You are reading a message` and click `OK`.
19. Open the **Label** dialog box. Type `Macro End` and click **OK**.
20. Add `Exit Macro` to end of macro.

Your completed macro should look like this:


```
Goto Label [Variable for screen text]
Message "Screen Unknown"
Goto Label "Macro End"
Label: A00
Message "PROFS Main Screen"
Receive at ( 24, 69 ) "Mail Waiting"
Message "You have Mail"
Label: NoMail
Goto Label "Macro End"
Label: C00
Message "You are on the Open Mail Screen"
Goto Label "Macro End"
Label: E01
Message "You are reading a message"
Label: Macro End
Exit Macro
```

21. Select **File** > **Save**, then and close the Macro Editor

This macro looks at the current **PROFS** screen and returns a message box containing a description of what was found. You can use this macro as a model to test different macro functions.

Running macros

This section assumes that you have created your macros. It tells you and your users how to run the standard Rumba macros and your custom macros in any of the several ways.

Running standard or custom macros

- From the **Tools** Menu, choose **Run Macro** or click the icon, and then select the macro you want to run.
- Associate a macro with a Hotspot, and then click the Hotspot text on the host screen to run the macro.
- Assign a macro to a toolbar button, and then click the button to run the macro.
- Link a macro to a keystroke, and then type the keys to run the macro.
- Macros can also be run from other macros. When editing the Run application command, select Macro Files from the Files of Type list. Choose the macro you want to run. When the macro comes to this command, it pauses, runs the new macro, and then runs the remainder of the originating macro.

Each host application type has a default macro directory.

Macros as Hotspots

Hotspots are words or sentences on the host screen that run macros when clicked. Hotspots add point-and-click capability to host screens, making them more like the other Windows applications. Instead of requiring users to master complicated command line statements, you can distribute Hotspot libraries to enable users to click through host screens with fewer errors and more efficiency.

You can find procedures for creating and editing Hotspots in the online help.

What are Hotspots

You can make any non-protected text on a host screen into a Hotspot and associate that text with any macro. The host screen shows Hotspots as 3D buttons (see below). When you click a Hotspot button, the macro runs.

For example, you could make the text in a list of options into Hotspots that automatically enter the appropriate option when clicked.
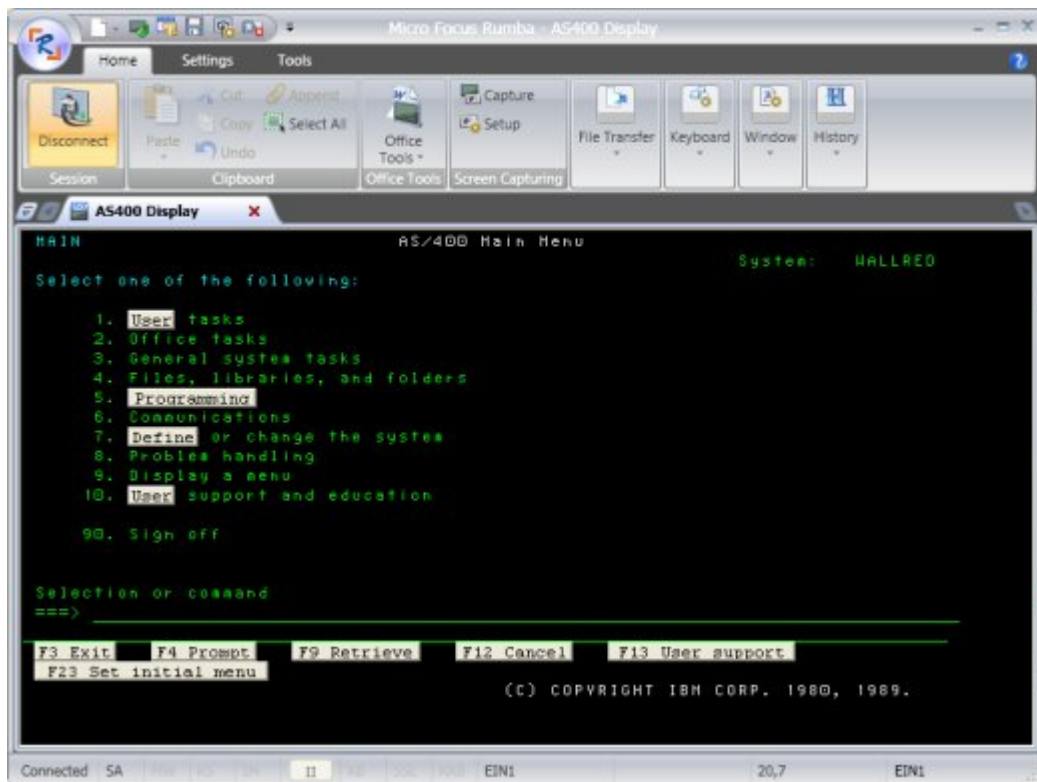
Hotspots are saved in libraries with an `.hsp` extension. You can create multiple libraries with distinct configurations. To use different collections of Hotspots, or to run different macros with the same Hotspot text, users just select the corresponding library.

Creating and distributing Hotspot libraries

You can create whole libraries of Hotspots and distribute them to users. When the user enables Hotspots, the macro runs when the user clicks the Hotspot. The macros used in your Hotspots are saved with the Hotspot library, so you do not need to distribute the .hsd files to your users.

Macros on your toolbars

Macros can be linked to toolbar buttons, and incorporated into your standard Rumba Toolbar or placed on custom toolbars. With custom toolbar macros, your users can accomplish complex key sequences and commands without errors and without even touching their keyboards.

Keyboard macros

If your users do not like switching between keyboard and mouse as they work, they can link macros to keystrokes. Without even taking their hands off the keyboard, your users can run the macros you distribute which create shortcut keys for complex key sequences.

Distributing Keyboard files

The keyboard files are located in the private `directory\hostname\keyboard` directory as `.map` files. To distribute keyboard files, you must distribute the `.map` file and any corresponding macro files (`.rmc` & `.mac`) that are linked to mapped keys. The macro files are located in the private directory\hostname \macros directory.

Macro commands

The macro commands available for each host type differ slightly. The following tables list the macro commands available for each host. Each table lists the macro command, followed by a brief description of its function.

Mainframe macro commands

An asterisk (*) indicates macro commands that you cannot record, but can create and edit in the Macro Editor.

Macro Command:	Use to:
Connect	Connect the PC to the mainframe host
*Compare Values	Compare two character strings or variables, and perform the specified commands based on the results
Cursor Position	Position the cursor on the host screen
Disconnect	Disconnect the PC from the mainframe host
Exit Macro	End the macro sequence
File Transfer Receive	Receive the specified file transfer
File Transfer Send	Send the specified file transfer

Macro Command:	Use to:
*Goto Label	Jump to the specified Label command in the macro sequence
Keystroke	Send a function key to the host application
*Label	Create a marker in the macro sequence to which a Goto Label command can jump
*Message	Show a dialog box with the specified message
*Pause Macro	Pause the macro sequence for the specified number of seconds
*Prompt	Create a dialog box that prompts the user for input
Receive	Search the host screen for the specified characters
*Run Application	Run an application on the local PC
Copy From	Copy selected areas of a display screen to the clipboard for pasting into an application or display screen
Paste To	Paste clipboard information to an open application or session.
Type	Send the specified characters or variable to the host screen

AS/400 macro commands

An asterisk (*) indicates macro commands that you cannot record, but can create and edit in the Macro Editor.

Macro command:	Use to:
Connect	Connect the PC to the AS/400 host
*Compare Values	Compare two character strings or variables, and perform the specified commands based on the results
Cursor Position	Position the cursor on the host screen
Disconnect	Disconnect the PC from the AS/400 host
Exit Macro	End the macro sequence
*Goto Label	Jump to the specified Label command in the macro sequence
Keystroke	Send a function key to the host application
*Label	Create a marker in the macro sequence to which a Goto Label command can jump
*Message	Show a dialog box with the specified message
*Pause Macro	Pause the macro sequence for the specified number of seconds
*Prompt	Create a dialog box that prompts the user for input
Receive	Search the host screen for the specified characters
Copy From	Copy selected areas of a display screen to the clipboard for pasting into an application or display screen

| Macro command:	Use to:
Paste To	Paste clipboard information to an open application or session.
*Run Application	Run an application on the local PC
Type	Send the specified characters or variable to the host screen

HP and UNIX macro commands

An asterisk (*) indicates macro commands that you cannot record, but can create and edit in the Macro Editor.

Macro Command:	Use to:
Connect	Connect the PC to the UNIX or HP host
*Compare Values	Compare two character strings or variables, and perform the specified commands based on the results
Disconnect	Disconnect the PC from the UNIX or HP host
Exit Macro	End the macro sequence
File Transfer Receive	Receive the specified file transfer
File Transfer Send	Send the specified file transfer
*Goto Label	Jump to the specified Label command in the macro sequence
Keystroke	Send a function key to the host application
*Label	Create a marker in the macro sequence to which a Goto Label command can jump
*Message	Show a dialog box with the specified message
*Pause Macro	Pause the macro sequence for the specified number of seconds
*Prompt	Create a dialog box that prompts the user for input
Receive	Search the host screen for the specified characters
*Run Application	Run an application on the local PC
Type	Send the specified characters or variable to the host screen

Express Macro Language

An alternative to using Rumba macros is to use Express Macro Language (EML) macros. EML is a new, code-free macro language that can be shared across mainstream Micro Focus emulators. You can therefore create an EML macro in one product and use it in another.

To create an EML macro, you use the EML Editor which is a visual editor where you drag and drop blocks of code onto a canvas and plug them together to obtain the function you want. For example:
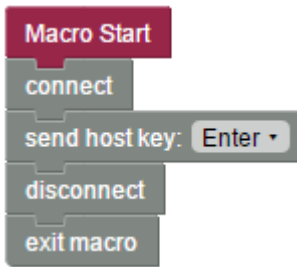
An advantage of using the EML Editor is that you cannot make syntax errors. If an error would be caused by joining two blocks together, the editor would not allow you to join the blocks. You can simply concentrate on the logical flow of the macro.

After you have created a macro, you save it as a JavaScript (.js) file.

For more information about EML and how to create EML macros, see the EML online Help.

Recording a macro

1.
   Select **Tools** > **Record Macro**, or click .
2. Perform the steps you want to record, such as logging on and opening a host application.
3.
   To end the recording, select **Tools** > **Record Macro** again, or click .

   The **Save As Macro File** window appears.
4. In the **File name** field, type a name for the macro.
5. From the **Save as type** list, select **EML Macro (*.js)**.

   The macro file is saved with a file extension of .js and the EML Editor appears. The macro is shown in the form of connected blocks.

   🖉 **Note:** If you do not want the editor to appear automatically, you can change this in the **Rumba Options** dialog box.
6. Select **File** > **Exit**.

🖉 **Note:** It is not possible to record Rumba FTP actions.

Running a macro

1.
   Select **Tools** > **Run Macro**, or click .

   The **Select Macro** dialog box appears.
2. Select the .js macro you want to run, then click **Open**.

   The macro runs.

🖉 **Tips:**

   - To interrupt a macro while it's playing, click .
   - To assign runtime properties to your macros, select **Tools** > **Macro Properties**.

Connection Tools

With Rumba Tools, you have a set of utilities designed to boost your productivity. This section provides overviews and tips for the most commonly used Rumba Tools.

- Rumba APPC Configuration utility
- Administrator Override File
- Rumba AS/400 Communications utility
- Rumba Communication Monitor

Note: For details on additional Rumba Administrative Tools, see the Rumba Utilities and Advanced sections in the online help.

The following summarizes the purpose and supported hosts for each of these administrator utility.

- Rumba APPC Configuration utility

  Simplify the process of creating connections throughout your network.

  Applies to mainframe and AS/400.
- Rumba Administrator Override File

  Implement a single configuration file for multiple users and reduce the time required for setting up and troubleshooting workstations.

  Applies to mainframe and AS/400.
- Rumba AS/400 Communications utility

  Simplify the process of creating communication sessions throughout your network.

  Applies to AS/400.
- Rumba Communication Monitor

  A workstation-connection monitor you can use to analyze the current status of each configured link.

  Applies to mainframe and AS/400.
- Rumba Trace Console

  Simplify the mechanics of tracing communication sessions to improve your efficiency in troubleshooting and solving network problems.

  Applies to mainframe, AS/400 and HP UNIX.

Rumba APPC Configuration tool

The APPC Configuration tool assists you in building a configuration file for the Rumba SNA engine. This configuration file defines the elements listed below. Each of these elements is discussed in this section.

- Connections
- AS/400 modes
- Remote and local logical units
- CPI-C side information file
- Transaction programs
- Conversation security

The following figure illustrates the Rumba APPC Configuration utility main display:
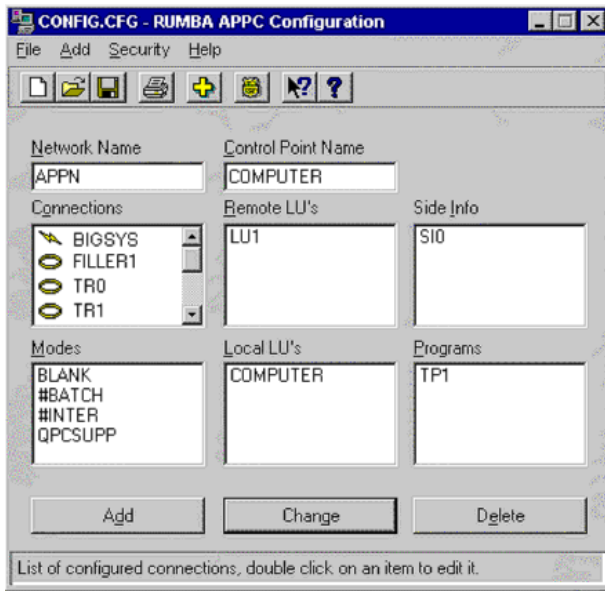
Because the Rumba APPC Configuration utility uses a feature-rich graphical user interface, when you make an addition, change, or deletion you only need to select a box and click a button.

The Rumba APPC Configuration utility saves configuration parameters in a file with a .cfg extension. You can save as many configurations as you need for distribution later, or you can build an administrator override configuration file for your users.

With the Rumba APPC Configuration utility, you can set values for different local and remote environments. For Common Programming Interface-Communications, you can add or modify side information entries. You also have the option of assigning User IDs and Passwords to incoming remote transaction programs.

Connections

Use the Rumba APPC Configuration utility to name connections and add interfaces quickly and without error.

Working in the background, the Rumba APPC Configuration utility creates a local LU name (logical unit name) for your connection from your network name and control point name. As a default, the Rumba APPC Configuration utility inserts your computer name for the control point name.

The Rumba APPC Configuration utility allows you to easily add an interface connection. The Rumba APPC Configuration utility's Connections box uses a simple graphics scheme that allows you to visually identify the interface used for your connection.
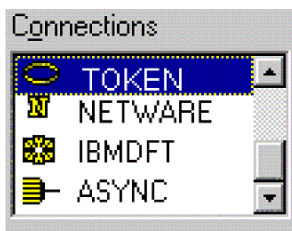
International character set translation

The Rumba APPC Configuration utility works with the international character set specified in the regional settings for your workstation. This translation is accomplished using tables that reside in the Program Files \Micro Focus\Rumba\System\Charsets directory after Rumba software is installed.

These tables (referred to as codepage tables) work with Rumba software as follows:

- The character set that Rumba software uses is determined by the country that is specified on the Regional Settings Properties dialog box. Access this dialog box via the Windows Control Panel.
- EBCDIC to ANSI translation (or mapping) for your workstation is based on the codepage table currently in use.
- The `Program Files\Micro Focus\Rumba\System\Charsets` directory contains files named chr2exxx.gen, where "XXX" is the locale code (for example, 037 is the code for USA).

To modify the character mapping for a specific language, modify the corresponding chr2exxx.gen file using a regular text editor. There are no registry settings or .ini file settings to make or change.

AS/400 modes

AS/400 mode descriptions define a number of session characteristics. These characteristics include the:

- Maximum number of sessions
- Maximum number of conversations
- Pacing values for request and response units

By using the mode description as an APPC system object, you can reduce the number of separate configuration definitions. The Rumba APPC Configuration utility offers several pre-configured modes. When in doubt, QPCSUPP is the AS/400 default, although #BATCH offers the best compromise in communication performance and cost.

Remote and local logical units

The Rumba APPC Configuration utility simplifies the task of managing remote and local logical units. A local logical unit (LU) represents the local workstation, while the remote logical unit represents the local workstation conversation partner.

- Logical unit properties

  Because the concept of the LU has evolved, it is easier to understand the LU by looking at its properties:

  - Sessions have LUs at each end
  - To establish a session, a client must have the server's LU name, which is known as the Remote LU
  - In a hierarchical network, an LU has a defined global address that it uses to route data
  - You can assign different LU names to a host's different subsystems

    Note: An LU name does not signify a machine or program name, it designates the role a machine or program has within a network. For example, an LU name can designate a network server.
- Logical unit configuration tips

  Here are a few tips to consider when you use the Rumba APPC Configuration utility to add or change logical units:

  - For a computer client, the LU name default can be the computer name. The Rumba APPC Configuration utility does this automatically in the configuration process.
  - For a server, consider using an LU name alias based on the function, rather than the computer name. This minimizes disruption if the function is moved to another computer.
  - When you are connected to a subarea SNA network, finding an LU name is problematic because you do not have network node services. In this case, you must configure a remote LU and associate it with a route.

CPI-C side information file

The Rumba APPC Configuration utility gives you a powerful tool that you can use to easily create and edit side information files. Previously, when using APPC/LU 6.2 verbs, application programmers had to supply

the parameters that described the network environment. With CPI-C, these parameters are called out from a side information file.

The Rumba APPC Configuration utility uses a straightforward process to create a side information file. One thing to remember is that the side information file is the last part of the configuration process. First you must define the:

- Network environment
- Session characteristics
- Remote LU information

Transaction programs

You can add a transaction program by naming and associating it with an executable file or command line. This transaction program (TP) can then have:

- A path to the executable program
- An auto-start option
- Optimum security
- Queueing
- Timeouts
- PreLaunch Counts

The transaction programs you add with the Rumba APPC Configuration utility should fulfill the APPC/LU 6.2 requirements of:

- Being invoked by the LU 6.2 ATTACH call
- Using conversations to communicate with a remote transaction program
- Establishing many conversations with one or more remote transaction programs using distinct sessions

Conversation security

If you have secured the local transaction program (TP), use the Rumba APPC Configuration utility's Conversation Security feature to assign user IDs and passwords to incoming TPs.

When an incoming TP attempts to allocate a conversation with the local TP, the local TP reads the User ID and password. If there is a match, the incoming TP is allowed to access the local TP.

Administrator override file

A high priority for any system administrator is the standardization of system configurations. When you create an administrator override file, you provide a set of well-defined parameters that each user can access. This tool can significantly reduce the time spent setting up and troubleshooting workstations.

Administrator override file definition

How does the administrator override file work? An administrator override (Admin Override) file is a configuration file you create and then place on the network. Your users can access this administrator override file for connection information to SNA-type machines. If the connection information changes, you can use the administrator override feature to make one central change.

For example, when you introduce a new AS/400 to your network, you only have to add the AS/400 information to one administrator override configuration file on your network. The next time a user pulls up the configuration dialog in the AS/400 display or Rumba APPC Configuration utility, the new AS/400 appears, configured and ready to go.

Note: The administrator override file does not contain the control point name that will be used for the connection; this name comes from the user's machine.

Administrator override file function

How does the administrator override file work? The Rumba SNA engine, AS/400 Display, and Rumba APPC Configuration utility all have the ability to look at two configuration files at the same time and extract, as necessary, pertinent information from each file.

Note: Mainframe connection definitions are not supported by the administrator override file. To understand why this is the case, consider the following:

- Connection definitions for the Rumba Router and AS/400 Display are sharable objects. That is, the parameters for an AS/400 connection definition and the AS/400 Remote LU Definition can be common to all users globally. The configuration parameters that are specific to a single workstation are located in the local LU definition.
- Connection definitions for a mainframe connection are not sharable objects. As an example, for a Mainframe Display DLC connection, each user must have an individual and unique PUID as part of the connection definition. If you tried to create an administrator override file for this connection, all users would share a single block ID and PUID, and only one user could connect at a time.

Basically, when you configure connections to an SNA-type server, the Control Point Name (also known as the "PC Name" in the AS/400 environment) is what makes one workstation unique from another workstation. The administrator override file contains the link information; this includes the network name, destination address, remote LU, side information, and more. Use the local configuration file to store information applicable only to the workstation.

Local connection configuration

If the administrator override file is set, can you still configure connections locally? Yes! You can enable Admin Override through the Rumba APPC Configuration utility and still create link configurations that you store on the workstation in a local configuration file. However, there are restrictions. You are able to configure local connections only when they have a different connection name and remote LU alias from any other link residing in the administrator override file. The reason for this is described in the following section on the precedence of configuration files.

Also, similar to document printing, the AS/400 display shows only connections configured in the administrator override file, and the user can connect only to these administrator override configured connections.

Configuration files precedence

The locally-administered configuration file is referenced in the registry as `HKEY_LOCAL_MACHINE \Software\WALLDATA\ConfigFile`

The configuration files exist so users can provide their configuration preferences within the communications environment.

The system-administered configuration file is referenced in the registry key as `HKEY_LOCAL_MACHINE \Software\WALLDATA\Sna\AdminOverrideConfigFile`.

You can use the administrator override configuration file as a set of configuration elements that will override all other configurations. Any configuration element you specify within `AdminOverride-ConfigFile` will then take precedence over the corresponding element contained in `ConfigFile`. This means that if collisions occur between the elements of the `ConfigFile` and `AdminOverrideConfigFile`, the winner will be the `AdminOverrideConfigFile`.

For example: if `ConfigFile` contains an MPTN link definition named NETRED and `AdminOverrideConfigFile` contains an SDLC link definition named NETRED, then `AdminOverrideConfigFile`'s SDLC link definition will replace the `ConfigFile`'s MPTN link definition.

Administrator override file creation

Creating an administrator override file that multiple users can access is not very different than creating a local configuration file for one user. The only difference is that you store the administrator override file on a shared network drive.

1. From any machine that already has Rumba software installed, configure the necessary connections using the Rumba APPC Configuration utility.
2. Save the file as you normally would for a local machine (note the file name in the title bar).
3. Make sure that each connection is valid by connecting to each link configured.
4. Once you are confident that you have a working configuration file, copy the file to a shared directory on the network. Make sure that each intended user has access to this drive.

> **Note:** Change the Administration Override configuration file's attributes to Read Only: this will prevent a user from accidentally corrupting the file.

Editing existing administrator override files

If an administrator override configuration file is already in place and is currently being accessed by users, we recommend the following as the safest editing method.

1. Using a file manager utility, copy the current administrator override configuration file and rename it.
2. Take off the Read-only attribute if it is present, so that the changes you make can be saved.
3. From the Rumba APPC Configuration utility menu bar, choose Open. Select your newly created file. Make changes as necessary.
4. Save your new administrator override configuration file.
5. Once you are confident that you have a working configuration file, copy the file to a shared directory on the network and replace the existing administrator override configuration file.

> **Note:** Change the Administration Override configuration file's attributes to Read Only: this will prevent a user from accidentally corrupting the file.

Setting user access to administrator override file

1. From the Rumba Administrative Tools program group, start the Rumba APPC Configuration utility.
2. From the **File** menu, choose **Advanced.**
3. Click **Use Administrator Override Configuration**.
4. Enter the full path of the Configuration file that you want the users to access. You can also use the Browse feature to find the file and its path name.

    You can ensure that the next person will get the same configuration by using the setting "Make configuration available to all users of this machine." This will use the `\\HKEY\LOCAL_MACHINE\` registry instead of the `\\HKEY\CURRENT_USER` registry.

5. From the **File** menu, choose **Save.**
6. Close the Rumba APPC Configuration utility.

When you specify the path, we strongly recommend that you use the Universal Naming Convention (UNC) instead of a mapped drive letter. If you use the UNC, the connection is not reliant on a mapped drive letter that could be overwritten or corrupted.

> **Note:** If you use the Browse feature, the path is returned in UNC format.

If the Rumba APPC Configuration utility cannot find the file you specified in the path for the administrator override file, a warning is displayed. The situation may be that the workstation is not currently attached to the network, or that the user does not have permission to access the server that you specified in the path. If this is the case and you are confident that the path is correct, choose to continue and exit.

From the operating system's ComputerName, the Rumba APPC Configuration utility extracts a default Control Point Name for the local configuration file. If you wish, you can specify another Control Point Name by simply altering the field. Any changes you make to the Control Point Name are also automatically made to the Local LU name.

It is necessary that you use a `.cfg` extension on the file name. If you do not specify an extension, the Rumba APPC Configuration utility will append `.cfg` to the file name. The Rumba APPC Configuration utility supports both 9.5 format file names and long file names.

Assessing workstation setup

Is the workstation you just set up configured properly? The best tool to use in determining if you have set up the workstation correctly is the Rumba Communications Monitor.

If you selected the APPC Client package install, the Rumba Communications Monitor is installed with a program icon in the Rumba Administrative Tools program group. If the APPC Client was not installed, the program will still be installed, but no icon is placed in the group. If this is the case, you can run the program from the Program Files\Micro Focus\Rumba\System subdirectory. The file name is `nof.exe`.

The Rumba Communications Monitor starts the Rumba SNA engine and displays the information that the Rumba SNA engine interprets. If the Rumba Communications Monitor displays the connections that you configured in the administrator override file, then you are off to a good start.

Administrator override file limitations

While the Rumba SNA engine has the ability to use Rumba administrator override configuration files, the Rumba AS/400 display and AS/400 printer are the only display-type services that currently use this feature.

The AS/400 Connection Configuration dialog allows the user to see and connect only to links that you configured in an administrator override configuration file. Local files cannot be accessed.

Note: If your company creates transaction programs that use the Rumba SNA engine, then the administrator override file should work fine.

Configuration-free workstations

With the Rumba Administrative Tools, you can have self-configuring workstations. By using SMS or a batch file, each new workstation will look for its:

* Administrator override file
* Local configuration file

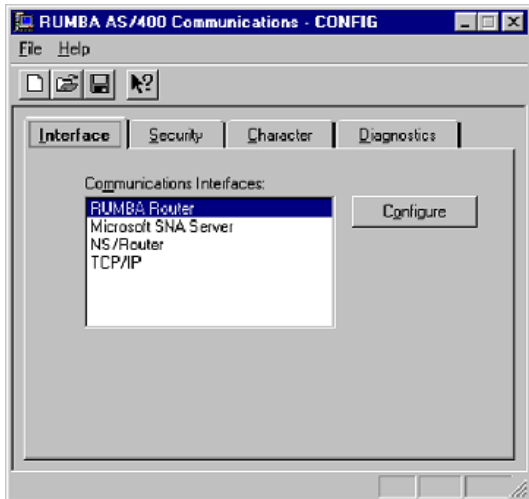# Rumba AS/400 Communications Utility

You can use the Rumba AS/400 Communications Utility to locate other AS/400 systems and to configure or change host connections and link interfaces for the:

* Application Program Interface (API)
* Submit Remote Command application and API

AS/400 Communications Utility overview

With this utility, the configuration settings are saved in a .ehn file. You can save multiple configurations under different .ehn file names for later use. The .ehn configuration file can be especially useful when you are setting up data queues for end users who do not have Rumba AS/400 display or printer capability.

The figure below illustrates the main dialog box of the AS/400 Communications Utility.

The table below maps the major tasks you perform with the AS/400 Communications Config utility to the appropriate tab on the dialog box.

Table 2: Configuration summary

To:	Go to the:
Configure communication interfaces	Interface tab
Configure link interfaces	Interface tab
Add AS/400 sign-ons	Security tab
Select a foreign language's host character set	Character tab
Display APPC error messages	Diagnostics tab

> **Note:** If you use Rumba AS/400 Communications to configure EHNAPPC, then all non-display and non-printer sessions will use the same configuration parameters.
>
> For example, if you configure EHNAPPC to connect to BIGSYS via APPC token ring for Data Queues, all other hosts (except for Display, Printer, and possibly RDA) will connect via APPC token ring.

AS/400 Communications Utility link configuration

To use the Rumba AS/400 Communications Utility to configure links:

1. Click the **Interface** tab.
2. Select a communications interface, then click **Configure**.

   For specific information on configuring, refer to the help section for the selected interface, found under Connecting to a host.
3. Select a system name, then click **Add**.
4. Select a link.

5. Click the **link interface** tab, then add or change configuration values as needed.

**6.** From the **File** menu, choose **Save**, or click the Save icon.

By default, the configuration is saved as an `.ehn` file.

Click the **Advanced** tab to specify a logical unit with a name different than your system's.

Communication Monitor

The Rumba Communication Monitor is a valuable utility that helps you monitor, troubleshoot, and optimize your network. Also, you can view non-configured system links.

The Rumba Communication Monitor displays what the Rumba SNA engine finds. The left pane shows that the APPN link is not connected and the right pane gives detailed information about the link.
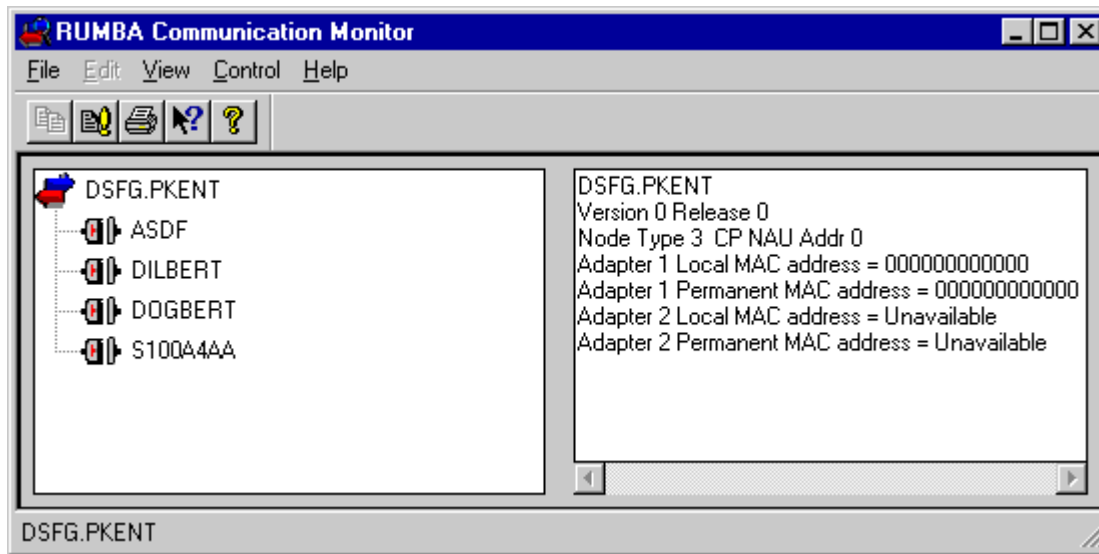
> *Note:* Even though you can use the Rumba Communication Monitor to start and stop links (either by the menu or by right clicking your mouse), we recommend that in normal circumstances (when you have a transaction program in process or an open Rumba Display session) you start and stop links and sessions from your Rumba display session.

MSS Automated Sign-On

You can configure Automated Sign-On for mainframe 3270 sessions. To do this, you:

- Create a connect macro and distribute it to the appropriate users.
- Create a session profile.
- Configure the session on the MSS.
- Authorize access to the session.

Prerequisites

- MSS is installed.
- The MSS Automated Sign-On Add-In is installed.
- Centralized management is enabled in **Rumba Options**.

Creating a connection macro

To create an Automated Sign-On macro:

1. Select **Tools** > **Record Macro**.
2. Perform the necessary steps to log onto a host application.
3. To stop recording, select **Tools** > **Record Macro**.

   The **Macro Editor** appears.
4. In the **Name** field, type a name for the macro.
5. In the **Key Name** field, type a 4-digit identifier for the macro.
6. Optional: Under **Play Macro On**, check **Connect**.
7. Insert the **Call External Provider** command before a **Type** statement where the user ID is provided.

   🖊 **Note:** The exact position in the macro for the command depends on the application concerned and how the application ID is provided:

   - As a string.
   - Taken from a screen.
   - As a prompt.

   The **Call External Provider** dialog box appears.
8. Select **[MSS-AutoSignOnProvider]** from the **Select Provider** list.
9. Provide the application ID for **[MSS-AutoSignOnProvider]** as an input parameter.
10. Click **OK**.
11. Double-click the **Type** statement that recorded the user name.

    The **Type** dialog box appears.
12. Select **[MSS-AutoSignOnProvider: Username]** from the **Characters to send** list.
13. Click **OK**.
14. Delete the **Type** statement that recorded the password.
15. Right-click and select **Add** > **Type**.

    The **Type** dialog box appears.
16. Select **[MSS-AutoSignOnProvider: Password]** from the **Characters to send** list.
17. Click **OK**.
18. On the **Macro Editor** toolbar, select **File** > **Save**.

    The macro is saved with the file name and a file extension of `.RMC`.
19. Distribute the macro to the appropriate users.

Creating a session profile

1. Start a mainframe session.
2. Configure the host.
3. Select **File** > **Save Session Profile**.

   The **Save Session Profile** window appears.
4. Type a name in the **File name** field.

   Make a note of the file name you used.
5. Click **OK**.

Configuring the session on the MSS

1. Start the MSS Administrative WebStation.
2. Click **Session Manager** in the navigation pane.
3. In the **Session Manager** pane, click **Add**.
4. In the **Add New Session** pane, select **Rumba** in the **Windows-Based** list.
5. Type a name for the session in the **Session name** box. Use the same name as the saved session profile.
6. Click **Continue**.
7. In the **Configure a Windows-Based Rumba session** pane, click **Browse** in the Rumba session profile section.
8. Select the saved session profile, then click **OK**.
9. Click **Save Settings**.

Authorizing access to the session

The way you authorise users for a session depends on the access control configuration of the MSS.

To authorise a user or group, click **Access Mapper** in the navigation pane, then use the online Help to associate them to the session you want.

When users next start Rumba, the session profile is automatically downloaded to their local machine. They can then open the profile in the normal way.

> **Note:** By default, session profiles are downloaded to:
>
> `C:\<user_name>\AppData\Local\Micro Focus\Rumba\MSS Sessions`

Diagnostic and Development Tools

This section gives you overviews and tips for the Rumba diagnostic and development tools.

Diagnostic tools

The diagnostic tools available in the **Rumba Administrative Tools** folder provide you with a set of powerful and cost-effective features that are designed to boost your productivity.

Rumba Trace tool

The Rumba Trace tool is a valuable tool that combines several trace options in a common utility. When you are troubleshooting a problem for a user, you can ask the user to trace and record communication sessions with the Rumba Trace tool. Later, you can selectively play back the binary data file to troubleshoot the problem. This section discusses the following aspects of working with the Rumba Trace tool:

- Application program interface options
- Difficult troubleshooting tasks
- Display software code version
- Playback display and printer data stream trace

By default, all application program interfaces (APIs) are selected for trace, but you can limit trace data and select only the APIs you want to trace. There are two guidelines for this decision:

- Trace options can have an impact on system performance, so you may want to use the trace APIs as filters when isolating a known problem.
- When diagnosing a problem, it often requires many types of trace information to effectively investigate the system state. You may want to select all APIs (default) to capture all available information and direct output to an unformatted file to send to Micro Focus Technical Support.

Application program interface options

The Rumba Trace Console gives you the following API options:

- Mainframe Display Data Stream (DSA)
- Mainframe Printer Data Stream (DSA)
- AS/400 Display Data Stream (DSA)
- AS/400 Printer Data Stream (DSA)
- UNIX/HP Display Data Stream (DSA)
- SNA Communications (includes APPC, CPI-C, LU 6.2, ACSSVC, and SNA)
- Micro Focus EHNAPPC (a standard AS/400 interface for APPC)
- Link Layer Communications for all network architecture drivers
- Rumba Display APIs (includes DDE, EHLLAPI, and WinHLLAPI)
- Host System APIs (includes Submit Remote Command)
- Rumba Event Viewer

Note: To view trace data while running the trace, be sure that the Display option is checked on the Options menu. This is the default setting.

Be aware that only 1000 lines of the most recent data is displayed. Also, note that the formatted file is distinct from the display file and carries more data. When large amounts of trace data are being saved, a larger trace buffer size tends to prevent trace buffer overruns. The default size is 8 MB.

Difficult troubleshooting tasks

When there is a network problem, but you do not know what to trace, you can trace all APIs and save the data in a binary file. Later, this data can be replayed, and you or your support engineers can analyze the sequence of data from API-to-API. With this sequence in hand, you can isolate where the problem is occurring and then replay the specific API data.

1. Open the Rumba Trace utility.
2. From the Rumba Trace Options menu, choose **Configuration**.

    The Configure Tracing Parameters dialog box will be displayed.
3. Click **Select All**.
4. Click the **Output** tab, and then select the **Binary File** box.
5. In the text box under **Binary Output**, type the path and file name to which you want to save the API data flow (use the extension `*.bin`).
6. Click **OK** to close the Configure Tracing Parameters dialog box.
7. Reproduce the problem that you are trying to troubleshoot. (For instance, attempt to run the Rumba software application that is failing to connect properly.)
8. From the Rumba **Trace Options** menu, choose **Configuration**.
9. On the **Output** tab, clear the **Binary File** box (turns off the binary option).
10. Click the **API Selection** tab, choose the APIs you want to analyze.
11. Click the **Advanced** tab. Under **Binary File Replay**, type in the name of the file you saved (or you can use the browse button to navigate to it).
12. Click **Replay Now** to view the API data flow.

Display software code version

The Rumba Trace Console contains a useful feature that can give you an all-inclusive software code version report that covers .cfg, .dll, .exe, and .ini files.

1. From the **Trace** menu, choose **Options**.
2. Choose **Code Version Snapshot**.

If you use this feature, you may want to run it either before or after you complete your trace, because this report is extensive and takes time to run.

Playback display and printer data stream trace

You can use Rumba Trace Console to record the AS/400 or Mainframe Display and Printer Data Streams, and then analyze and change this data, including adding rules of behavior on key actions and changing labels. This feature is especially useful for troubleshooting problems, or when you need to create a demonstration host for training or marketing purposes.

1. Open the Rumba Trace utility.
2. From the Rumba Trace Options menu, choose **Configuration**.
3. On the **API Selection** tab, select a single Display or Printer Data Stream to record.
4. Click the **Output** tab, select the **Formatted File** box, and then type in the path and name of the file you want to create (use the extension `*.fmt`)
5. Click **OK** to close the **Configure Tracing Options** dialog box.
6. Reproduce the problem or demonstrate the Rumba software feature you want to record.
7. Close the Rumba Trace utility and then start a Rumba Display or Printer session.
8. From the **Connection** menu, choose **Configure**, select **Demo Host** from the Installed Interfaces box, and then click the **Demo Host** tab.
9. In the **Script file** box, type in the path and name of the file you saved (or you can use the **Browse** button to navigate to it).

10. Click **Reconnect** to demonstrate the feature or problem you recorded.

> **Note:** The Demo Host can play back only one output file at a time, so make sure that you only choose one at a time on the API Selection tab of the Configure Tracing Options dialog box.

Rumba Tracer

Rumba Tracer allows users to record information about a program's execution. This information is typically used by programmers for debugging purpose. Depending on the type and detail of information contained in a trace log, the information is also used by experienced system administrators or technical support personnel, and software monitoring tools to diagnose common problems with software.

> **Hidden fields:** By default, the contents of hidden fields do not appear in trace results. This is useful to protect security-sensitive information, such as passwords.
>
> To disable this feature:
>
> **1.** Create a TraceSecurity value of type DWORD under the appropriate registry key:
>
> - WallData\Office\Mframe\ for mainframe sessions.
> - WallData\Office\AS400\ for AS/400 sessions.
>
> **2.** Set the value of TraceSecurity to 1.

Trace files can be produced by Rumba software components which have been provided with additional code for generating binary output trace files. The file can be viewed using the `TraceView` and `tracefmt` utilities, which are part of the *Windows Driver Kit (WDK),* formerly known as the Driver Development Kit (DDK). Only `TraceView` and `tracefmt` need to be installed from the WDK.

TraceView A graphical viewer for viewing trace files. In addition to trace files, it requires debug information in the shape of `.tmf` files generated by the Rumba.

Tracefmt Converts binary trace files to text files which can be viewed by any text viewer.

To start tracing:

1. Open a command window.
2. At the system prompt, enter:

```
RumbaTrace.cmd start
```
3. To stop tracing, enter:

```
RumbaTrace.cmd stop
```

The binary output trace file is stored in the `%temp%` folder.

The `RumbaTrace.cmd` script is part of the Rumba installation and is stored in the Rumba `System` folder.

You do not need to restart Rumba after running `RumbaTrace.cmd`.

Development tools

This section introduces the Rumba Development Edition for .NET FrameWork (RDE .NET) and Rumba Development Add-in for Visual Studio.

Rumba Development Edition for .NET FrameWork (RDE .NET)

A set of controls and components that enable developers to build applications which incorporate Rumba displays. The Rumba Developer Edition object module was restructured to be more intuitive. It is now fully integrated into the .NET C# managed code environment exposed through pre-built .NET binaries. The

interface introduces a much clearer objects hierarchy with some enumerations extensions for stronger parameters typing, shorter paths to methods and more sharing between session types.

✎ **Note:** Although RDE controls appear in the Visual Studio Toolbox for Visual Basic, C#, and Visual C++ projects, they can be used only for C# and Visual Basic projects .

Rumba Development Add-in for Visual Studio

This add-in is provided separately and provides objects for developers to use in .NET code to add mainframe and AS/400 printing, HLLAPI, macros and FTX functionality to your Visual Studio projects. RDE .NET controls appear in the Microsoft Visual Studio Toolbox as Mainframe, AS/400, UNIX and HP displays.

The add-in provides objects for developers to use in .NET code to add mainframe and AS/400 printing, HLLAPI,macros and FTX functionality to your Visual Studio projects. The add-in also provides:

* Rumba project templates for different session types
* Integration with the Visual Studio Toolbox for adding Rumba RDE.NET functionality with easy drag and drop.
* A menu item that allows one-click loading of dozens of examples with full source code included.
* Visual Studio code snippets for writing program code more quickly, easily, and reliably.
* 'Intellisense' support which provides auto-completion help information and short descriptions for all properties, methods, interfaces, and enumerations.
* RDE Tracer to monitors calls to RDE properties, methods, and events for easy application tracking. A tracer class has been added to each display control, which has two methods: start and stop. Once started, all RDE methods, properties, and events used by developers are logged to a log file, together with their respective parameters and returns values, for easy understanding of the RDE procedures at run-time.
* Full documentation of the RDE.NET API with a description of building you first RDE.NET application.

Installing the add-in

1. Close all open instances of Visual Studio.
2. Run `RumbaDevelopersEditionVisualStudioAddIn.msi` and follow the instructions in the Setup wizard.
3. When prompted, select at least one version of the **RDE Add-In for Visual Studio**.

You can remove the add-in from the machine by choosing **Start** > **Control Panel** > **Add or Remove Programs**.

Silent install

Installing to all versions of Visual Studio

To install the add-in to all versions of Visual Studio installed on the machine and supported by the add-in, use the following command line:

```
msiexec /i RumbaDevelopersEditionVisualStudioAddIn.msi /qn
```

Installing to a specific version of Visual Studio

To install the add-in to specific versions of Visual Studio installed on the machine, use the following command line:

```
msiexec /i RumbaDevelopersEditionVisualStudioAddIn.msi ADDLOCAL=
<version_list> /qn
```

where `<version_list>` is a comma separated list of Visual Studio versions. The list can contain all or some of the following items:

RumbaDE_VSD_2012

RumbaDE_VSD_2013

RumbaDE_VSD_2015

Example:

```
Msiexec /i RumbaDevelopersEditionVisualStudioAddIn.msi ADDLOCAL=
RumbaDE_VSD_2012, RumbaDE_VSD_2013, RumbaDE_VSD_2015 /qn
```

Silent uninstall

To uninstall the add-in, use the following command line:

```
Msiexec /x RumbaDevelopersEditionVisualStudioAddIn.msi
```

Contacting Micro Focus

Our Web site gives up-to-date details of contact numbers and addresses.

Further information and product support

Additional technical information or advice is available from several sources.

The product support pages contain a considerable amount of additional information, such as:

- The *Product Updates* section of the Micro Focus SupportLine Web site, where you can download fixes and documentation updates.
- The *Examples and Utilities* section of the Micro Focus SupportLine Web site, including demos and additional product documentation.
- The *Support Resources* section of the Micro Focus SupportLine Web site, that includes troubleshooting guides and information about how to raise an incident.

To connect, enter *http://www.microfocus.com/SupportLine* in your browser.

Note: Some information may be available only to customers who have maintenance agreements.

If you obtained this product directly from Micro Focus, contact us as described on the Micro Focus Web site, *www.microfocus.com*. If you obtained the product from another source, such as an authorized distributor, contact them for help first. If they are unable to help, contact us.

Also, visit:

- The Micro Focus Community Web site, where you can browse the Knowledge Base, read articles and blogs, find demonstration programs and examples, and discuss this product with other users and Micro Focus specialists.
- The Micro Focus YouTube channel for videos related to your product. .

Information we need

However you contact us, please try to include the information below, if you have it. The more information you can give, the better Micro Focus SupportLine can help you. But if you don't know all the answers, or you think some are irrelevant to your problem, please give whatever information you have.

- The name and version number of all products that you think might be causing a problem.
- Your computer make and model.
- Your operating system version number and details of any networking software you are using.
- The amount of memory in your computer.
- The relevant page reference or section in the documentation.
- Your serial number. To find out this number, look in the subject line and body of your Electronic Product Delivery Notice email that you received from Micro Focus.

Contact information

Our Web site gives up-to-date details of contact numbers and addresses.

Additional technical information or advice is available from several sources.

The product support pages contain considerable additional information, including the *Product Updates* section of the Micro Focus SupportLine Web site, where you can download fixes and documentation updates. To connect, go enter *http://www.microfocus.com/SupportLine* in your browser.

If you are a Micro Focus SupportLine customer, please see the *Welcome to Customer Care* document that includes information about downloading and licensing your product, contacting Customer Care, and about reporting an incident. Support from Micro Focus may be available only to customers who have maintenance agreements.

Tell Us What You Think

We welcome your feedback regarding Micro Focus documentation.

Send feedback about this Help documentation.

Click the above link to email your comments to Micro Focus.

Index

C

contact information 132

E

EML macros 115

M

macros
    EML 115

recording 115
running 115

P

product support 132

R

recording macros 115
running macros 115