



Rumba+ Desktop 10.2 SP2

Rumba+ Desktop 10.2 SP2 System
Administrator Guide

© 1984-2024 Rocket Software, Inc. or its affiliates. All Rights Reserved.

Rocket is a registered trademark of Rocket Software, Inc. For a list of Rocket registered trademarks go to: www.rocketsoftware.com/about/legal. All other products or services mentioned in this document may be covered by the trademarks, service marks, or product names of their respective owners.

IBM and CICS are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide.

2024-09-27

Contents

About This Guide	8
Additional information	8
Conventions used in this guide	8
Introducing Rumba+	9
About Rumba+	9
Rumba+ products	9
Host-specific features	10
Mainframe features	10
AS/400 features	11
UNIX and HP features	11
Core features	13
Multiple HLLAPI sessions	13
Enhanced security through SSL/SSH	13
Certificate Express Logon	14
Host sessions	14
Rumba+ printer	14
Copy and paste	14
File transfer	14
Screen displays	15
International keyboards	15
Supported VT features	15
Supported HP features	16
Automating your work	16
Rumba+ tools	16
Productivity tools	16
Configuration and maintenance	18
Developer tools	19
Planning Installation and Deployment	20
Windows terminology	20
Windows Installer service	20
Software requirements	21
Supported operating systems	21
Prerequisite software	21
Hardware requirements	21
Things to consider before installation	21
Deployment strategies	22
Migration issues and directory locations	22
Microsoft Host Integration Server (HIS) 2000 Server and Client	22
Changes to Rumba+ components	22
Installing Rumba+	23
The installation process	23
Default path of user profile	23
Using the Setup wizard	23
Using the command line	24
Rolling back an installation	27
Changing Rumba+	27
Repairing Rumba+	27
Removing Rumba+	27
Customizing setup	28
Using transforms	28

Windows Installer command line arguments	30
Distributing Rumba+ with SCCM	32
Uninstalling previous versions of Rumba+	32
Preparing client computers for Windows Installer	32
Creating a Rumba+ deployment package	32
Advanced deployment issues	33
Rumba+ file locations	33
Using Rumba+	36
Starting a session from the command line	36
Starting a session from a URI	37
Rumba+ folders	37
Using Rumba+ menus, toolbars, and the status bar	38
Enabling and disabling global commands	38
Customizing menus and toolbars	38
Mainframe status bar	39
AS/400 status bar	39
HP status bar	40
UNIX status bar	40
Rumba+ session profiles	40
About session profiles	41
Supplemental and interface configuration files	41
Migrating session profiles from Rumba+ 6.x or 7.x	41
Configuring session profiles	42
Rumba+ desktop profiles	46
About desktop profiles	46
Session profile locations	46
Creating and deploying sessions and settings with the MSS Administrative Console	46
Creating or modifying a centrally managed session	47
Making centrally managed sessions available to users	47
Using centralized management to deploy MSI Packages	47
HP keyboard key codes	47
Keys mapped to active default settings	47
Keys used to send 7-bit control characters	48
Codes sent by editing keys	49
Codes sent by arrow keys	49
Codes sent by numeric keypad keys	50
VT keyboard key codes	50
VT codes sent by top-row function keys	50
VT keys used to send 7-bit control characters	51
VT codes sent by editing keys	52
VT codes sent by arrow keys	52
VT codes sent by numeric keypad keys	52
Configuring mainframe applications with Rumba+ Desktop and Enterprise Server	53
Using Plus	54
Using code	54
Using the Screen Designer	54
How it works	54
Customization projects	55
Deploying a customization project	55
Using the Screen Designer command line	55
Generating Plus archives	56
Importing project files	57
Migrating customization projects from earlier versions of Plus	57
Mainframe File Transfer	59
Mainframe file transfer features	59

Sending and receiving files	59
Configuring your PC	59
Sending a file to the host	59
Receiving a file from the host:	60
Batch file transfers	60
File transfer registry settings	60
Using macros to transfer files	61
Creating a file transfer macro	61
IND\$FILE TRANSxx codes	61
AS/400 Getting Connected	64
Connecting to the host	64
Starting the connection process	64
Selecting an interface	65
Configuring the interface	66
Successful connection indicator	66
Other helpful features	66
Using the Rumba+ Router to configure an APPC link	67
Overview of Rumba+ Router	67
Connection Configuration dialog box	67
Setting up the connection	67
Rumba+ Router link types	70
Native TCP support	72
Rumba+ TCP architecture	72
Supported Rumba+ components	73
Configuring a TCP connection	73
Trace support for TCP applications	74
Configuring an MPTN connection	74
About MTPN	74
AS/400 considerations	74
MPTN parameters	75
Configuring a Microsoft SNA Server connection	76
Diagnostics	76
Configuring the AS/400 Communications tool	76
Connecting to the host	76
Interface page	77
Security page	78
Character page	79
Diagnostics page	79
AS/400 File Transfer	80
AS/400 File Transfer features	80
Installation	80
Using profiles	80
User files	81
Changing the library list on the AS/400 host	81
Copying the QDFTJOBDB file	81
Changing the library list in the QDFTJOBDB file	82
Updating the user profile	82
Creating a PC description file	82
Structure of the PC description file	83
Sample PC description file	86
Using SQL: Examples	86
Example 1: Using SELECT, WHERE, and ORDER BY	87
Example 2: Querying multiple files with JOIN BY	88
Example 3: Summarizing records with GROUP BY	89
AS/400 Client Application	91

AS/400 Client installation and availability	91
Before starting	91
Interfaces supported	91
Rumba+ Submit Remote Command	91
HP and UNIX File Transfer	93
How to transfer files using Rumba+ HP and UNIX	93
Understanding the file transfer process	93
MPE file transfers	94
Using compatible file transfer protocol versions	94
8-bit binary transparent communications	94
Using Rumba+ macros to transfer files	94
AS/400 and Mainframe Printers	95
AS/400 and mainframe printer features	95
Printing host documents on a PC printer	95
Viewing documents before printing	95
Monitoring and controlling printers and print jobs	95
AS/400 Printer Queues	96
Direct printing	96
Printer sharing and End print job	96
PC Printer Queues	96
Viewing print job history	96
Formatting printed documents	97
Host Print Transform	97
Font mapping	97
Character translation	97
Printer Session Manager	98
Using the command line	98
Examples	99
Using the Printer Session Manager	99
Administrative installation	101
Using Macros	102
Creating macros	102
Connect and disconnect macros	102
Specifying connect and disconnect macros	102
Rumba+ Macro Editor	103
Copying data between applications	103
Using variables	103
Running macros	105
Running standard or custom macros	106
Macros as Hotspots	106
Macros on your toolbars	106
Keyboard macros	106
Macro commands	106
Mainframe macro commands	107
AS/400 macro commands	108
HP and UNIX macro commands	108
Express Macro Language	109
Recording a macro	110
Running a macro	110
VBA macros	110
Connection Tools	112
Rumba+ APPC Configuration	112
Connections	113
International character set translation	113
AS/400 modes	113

Remote and local logical units	114
CPI-C side information file	114
Transaction programs	114
Conversation security	115
Administrator override file	115
Description	115
Function	115
Local connection configuration	116
Creating a file	116
Editing existing files	116
Setting user access	117
Assessing workstation setup	117
Rumba+ AS/400 Communications	118
Overview	118
Configuring links	119
Rumba+ Communication Monitor	119
MSS Automated Sign-On	120
Prerequisites	120
Creating a connection macro	120
Creating a session profile	121
Configuring the session on the MSS	121
Authorizing access to the session	121
Diagnostic and Development Tools	122
Diagnostic tools	122
Rumba+ Trace	122
Rumba+ Tracer	124
Development tools	124
Rumba+ Development Edition for .NET FrameWork (RDE .NET)	125
Rumba+ Development Extension for Visual Studio	125
Rumba+ Desktop 64-bit applications support	126
Notices	127
ROCKET SOFTWARE END USER LICENSE AGREEMENT	129

About This Guide

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

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



Note: IBM and CICS are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide.

Additional information

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+.

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+](#) on page 23.

To open the Help, select **Help > Rumba+ Desktop Help Topics**,

Conventions used in this guide

The following conventions are used in this guide:

Item	Description
File names	File names and extensions appear in <code>monospace</code> 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 <code>monospace</code> type.
Blue text	Cross-references to additional information appear blue text.

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, and HP. Rumba+ can 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. You have a choice of protocols, such as TCP/IP and Rumba+ Router (MPTN).

All Rumba+ products operate on Windows 8.1 and 10, WTS, and Citrix Presentation Server. In addition to providing multiple access solutions, the 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. Rocket Software produces and packages these 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:

Product	Description
Rumba+ Office	<ul style="list-style-type: none">• 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. <p>Requires multiple network protocols.</p>
Rumba+ Citrix Edition	<ul style="list-style-type: none">• 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. <p>For Citrix XenApp, Microsoft App-V, Windows Terminal Server.</p>
Rumba+ Mainframe Edition	<ul style="list-style-type: none">• Supports IBM mainframe.• Supports multiple servers and network protocols.

Product	Description
Rumba+ AS/400 Edition	<ul style="list-style-type: none"> • Supports display, printing, and file transfer capabilities for the mainframe host. • Provides browser-to-host connections and an integrated set of Internet applications. • 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	<ul style="list-style-type: none"> • 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.

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.

Standard features and interfaces of Rumba+ Mainframe Edition:

Feature	Description
Display	<ul style="list-style-type: none"> • Presents the traditional mainframe display terminals (3278/79) in a Microsoft Windows application: 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.

Feature	Description
File transfer	<ul style="list-style-type: none"> • 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).

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.

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.

Standard features for UNIX and HP:

Standard feature	Description
Display	<p>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.</p> <p>Users can run multiple host sessions, displaying and working with host applications simultaneously.</p>
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 (Transport Security Layer) 1.0, 1.1, 1.2, 1.3, 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.

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.

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.

Standard features for HP:

Standard feature	Description
Display	<p>Presents the traditional HP terminal types in a Microsoft Windows application.</p> <p>Users can run multiple host sessions, displaying and working with host applications simultaneously.</p>
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

Standard feature	Description
	transferred, and Rumba+ provides the underlying commands to successfully transfer the file.

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.

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.

You can use Rumba+ to:

- 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, 1.3 (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 (PUIs) 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.

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 Rocket Software.

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.

- VBA support

VBA support provides the ability to write more complex and powerful automation. VBA support uses standard VBA functionality, IDE and allows you to run, record, and edit macros from and to a common Macro Module shared between all sessions.

Rumba+ tools

All Rumba+ products contain the following tools:

- Productivity tools
- Configuration and maintenance tools
- Developer tools

More complex and powerful task automation is also provided by Rumba+ VBA support.

Productivity tools

Rumba+ applications contain tools that help end users eliminate 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) enable .NET managed code to interact with an Office application's COM-based object model.

Office version	PIA auto installer	Rumba+ support
2016	Yes	Supported
2019	Yes	Supported
365	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 customizing the display.

The following table describes the more important Rumba+ tools:

Tool	Description
Rumba+ Desk	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.

Tool	Description
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.
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.
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.
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, as described in the following table:

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: <ul style="list-style-type: none"> 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
	<ul style="list-style-type: none"> Host system APIs, such as Data Queues, Shared Folders, AS/400 file transfers, and Submit Remote Command.
Rumba+ Communication Monitor	<p>Displays an in-depth look at the connections of a workstation.</p> <p> Note: Not available for Rumba+ UNIX-HP Edition.</p>

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 can 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	<p>The Windows Installer replaces previous installation methods, and enables cleaner installs and uninstalls, as well as repairs.</p> <p>The patch install package file (.msp) allows you to apply hot fixes to your Rumba+ product.</p> <p>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 Customizing setup.</p>
Publishing and assigning with Intellimirror	<p>Administrators can publish Rumba+ to users or assign Rumba+ to users or computers.</p> <p>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.</p> <p>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 Control Panel > Programs and Features.</p> <p> Note: You can only publish to users, not computers.</p>
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.

Software requirements

The following software is required:

Supported operating systems

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

- 64-bit Windows 10 and 11
- 64-bit Windows Server 2016 and 2019, Windows 2022
- Windows Terminal Server (WTS)
- Citrix XenApp (formerly Presentation Server)
- Microsoft Application Virtualization (APP-V)

Prerequisite software

The following software is required:

- Microsoft .NET Framework 4.8
- Microsoft Visual C++ 2015-2022 Redistributable (x86)
- Microsoft Visual C++ 2015-2022 Redistributable (x64)
- Microsoft Visual C++ 2010 SP1 Redistributable (x86)
- Microsoft Visual Basic for Applications 7.1 (x86)
- Microsoft Visual Basic for Applications 7.1 (x86) English
- Microsoft Visual Basic for Applications 7.1 (x86) German
- Microsoft Visual Basic for Applications 7.1 (x86) French

To install these, run `RumbaPrerequisites.exe`.

Hardware requirements

The following hardware is required:

- 1 GB RAM
- Disk space (minimum): 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 Microsoft System Center Configuration Manager (SCCM) for your distributed installs?
- Will you be using SCCM 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+](#) on page 23 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	<p>These installation options allow you to control which Rumba+ features are installed. The features available depend on your Rumba+ edition.</p> <ul style="list-style-type: none">• 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](#).

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 SCCM deployments, it is strongly advised that you create separate SCCM 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.

Installing Rumba+

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

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

- Installing, modifying, repairing, and uninstalling Rumba+.
- 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\<user_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 components for Rumba+ 64-bit Windows support.
- 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. For example, `Rumba_Office.exe`. Provide the path for where the file contents should be extracted.
2. Follow the instructions in the Setup wizard.
3. 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.

4. Choose a directory for Rumba+ to be installed to. We recommend that you use the default directory. You can also choose the location for application data files.
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 **Rumba+ Desktop** entry on the Windows **Start > All Programs** menu from which you can access all the Rumba+ products and related tools that you have installed.

Administrative installation

1. Double-click the appropriate `Rumba_<...>.exe` file. For example, `Rumba_Office.exe`. Provide the `<absolute_path>` to where the file contents should be extracted.
2. Cancel all following wizards.
3. To install Rumba+ as an administrative installation, use the following command:

```
msiexec /a <absolute_path>\<Rumba_<...>.msi [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

```
msiexec <INSTALLFLAG> <absolute_path>\<Rumba_install_file>.msi <msiFlags>
MRILANG=<Rumba_UI_language> INSTALLDIR=<Rumba_installation_folder>
PRIVATEDIR=<user_profile_folder> INTERFACEPRIVA
TEDIR=<interface_configurations_folder> CUSTOMIZEDDIRS=1
```

Parameters

Parameter	Description	Required
<code><INSTALLFLAG></code>	Specifies whether to install or uninstall Rumba+: <ul style="list-style-type: none"> • Specify <code>/i</code> to install. • Specify <code>/X</code> to uninstall. 	Yes.
<code><absolute_path></code>	Absolute path to the Rumba+ installation file.	Yes.
<code><Rumba_
+_install_file>.msi</code>	The type of installation file in the form of: <code>Rumba_<package>.msi</code> for a licensed installation. <code>Rumba_<package>_eval.msi</code> for an evaluation installation. where <code><package></code> is one of: UNIX Office Mainframe Citrix	Yes.

Parameter	Description	Required
	AS400	
<msiFlag>	<p>Specifies the installation mode. Either or both of silent and with a log file.</p> <p>Silent:</p> <p>/q {n,b,r,f}</p> <p>This sets the user interface level:</p> <p>n - No UI. b - Basic UI. r - Reduced UI. f - Full UI (default).</p> <p>Log file:</p> <p>/l[i w e a r u c m o p v x + ! *] <log_file></p> <p>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.</p> <p>/log <log_file> - Equivalent of /l* <log_file>.</p>	No. For either silent mode or to create a log file.
MRILANG	<p>Specifies the installation language:</p> <p>English: MRI2924 French: MRI2928 German: MRI2929</p>	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

```
msiexec /i C:\Users\Admin\Desktop\Rumba_Office.msi
```

Standard uninstall

```
msiexec /x C:\Users\Admin\Desktop\Rumba_Office.msi
```

Silent installation with non-default Rumba+ INSTALLDIR

```
msiexec /i C:\Users\Admin\Desktop\Rumba_Office.msi /qn INSTALLDIR=C:\MyRumbaInstallDir
```

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

```
msiexec /i C:\Users\Admin\Desktop\Rumba_Office.msi /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

```
msiexec /i C:\Users\Admin\Desktop\Rumba_Office.msi /qn /lv 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

1. Double-click Rumba_Office.exe to extract Rumba+ prerequisites to a temporary location.
2. Dismiss any following wizards.
3. Run `<absolute_path>\RumbaPrerequisites.exe /s`.

Silent installation with a log file

1. Double-click Rumba_Office.exe to extract Rumba+ prerequisites to a temporary location.
2. Dismiss any following wizards.
3. Run `RumbaPrerequisites.exe /s /l <log_file>`.

Example

```
<absolute_path>\RumbaPrerequisites.exe /s /l C:\instLog.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.

Changing Rumba+

To change a Rumba+ installation:

1. Select **Control Panel > Programs and Features > Change**.

The Rumba+ Setup wizard starts.

2. Click **Next**.

The **Change, repair, or remove installation** window appears.

3. Click **Change**.

The **Custom Setup** window appears.

4. Make the changes you want.

Repairing Rumba+

To repair a Rumba+ installation:

1. Select **Control Panel > Programs and Features > Change**.

2. Click **Next**.

The **Change, repair, or remove installation** 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 > Programs and Features > Uninstall**.

Ensure that Rumba+ installation folders (normally `C:\Program Files (x86)\Micro Focus\Rumba` and `C:\Program Files\Micro Focus\Rumba`) were deleted during removal.



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 Rocket Software products installed on your computer, RSS Client files are not removed during the uninstall process. Removing them could cause another Rocket Software product to stop functioning.

Customizing setup

The **Custom** option of the Rumba+ Setup wizard allows you to select the features you want to install and customize your Rumba+ installation for individual, group, and enterprise-wide distributions. 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](#) on page 32.

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+, the 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, Rocket Software 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:

```
msiexec /a <absolute_path>\<Rumba_install_file>.msi [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: To extract an MSI, double-click the appropriate Rumba_<...> .exe file. For example, Rumba_Office.exe. Provide the <absolute_path> to where the file contents should be extracted.

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 InstallShield, 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:** To extract an MSI, double-click the appropriate Rumba_<...> .exe file. For example, Rumba_Office.exe. Provide the <absolute_path> to where the file contents should be extracted.
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.

Creating a transform using WIX


Run the following command using the WIX tool `torch.exe`:

```
torch.exe <Full_name_of_base_MSI> <Full_name_of_MSI_Containing_Updates> -out  
<Full_name_of_MST_file>
```

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. 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.

Option	Parameters	Definition
		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][+][!] <i>logfile</i>	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	<i>PatchPackage</i>	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. 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 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 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 before a Rumba+ installation, on an on-demand basis.

You can download the Microsoft Installer SDK from <https://learn.microsoft.com/en-us/windows/win32/msi/platform-sdk-components-for-windows-installer-developers>.

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 > Programs and Features**. This allows SCCM to remove the application automatically.

Rumba+ file locations

Preinstalled files

default.mnc	C:\ProgramData\Micro Focus\Rumba
SmpLeAs4.rmc	C:\Program Files (x86)\Micro Focus\Rumba\As400\Macro
SmpLeHP.rmc	C:\Program Files (x86)\Micro Focus\Rumba\Hp\Macro
SmpLeTSO.rmc	C:\Program Files (x86)\Micro Focus\Rumba\Mframe\Macro
SmpLeVM.rmc	C:\Program Files (x86)\Micro Focus\Rumba\Mframe\Macro
SmpLeVT.rmc	C:\Program Files (x86)\Micro Focus\Rumba\Vt\Macro

Run-created files

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

Macros

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

Plus project files

Plus 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

Help files

.chm

English C:\Program Files (x86)\Micro Focus\Rumba\Help
\MRI2924

French C:\Program Files (x86)\Micro Focus\Rumba\Help
\MRI2928

German C:\Program Files (x86)\Micro Focus\Rumba\Help
\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 MFrame and AS400. Optional.
<IP_or_host_name>	specifies the IP address or name of the host to connect to. Can be used with /Display:MFrame or /Display:AS400 for which it is required.
<port_number>	the port number to connect to. Can be used with /Display:MFrame or /Display:AS400 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.

Starting a session from a URI

As a system administrator, you can create a web page that has links to session profiles that you provide for your users.

Users click one of the provided links. Rumba+ Desktop starts automatically and loads the selected session profile.

Syntax:

```
mfrumbalaunch://file:///<PATH>[?CreateLocalCopy=true/false]
```

HTML examples:

```
<a href="mfrumbalaunch://file:///C:\Users\<username>\AppData\Local\Micro Focus
\Rumba\MFrame\test.rsdm?CreateLocalCopy=TRUE">Local Link Full Path AND
CreateLocalCopy=true</a>
```

```
<a href="mfrumbalaunch://file:///C:\Users\<username>\AppData\Local\Micro Focus
\Rumba\MFrame\test.rsdm?CreateLocalCopy=false">Local Link Full Path AND
CreateLocalCopy=false</a>
```

```
<a href="mfrumbalaunch://file:///C:\Users\<username>\AppData\Local\Micro Focus
\Rumba\MFrame\test.rsdm">Local Link Full Path</a>
```

```
<a href="mfrumbalaunch://file:///\\<network_path>\test.rsdm">Local Link
Network Path</a>
```

```
<a href="mfrumbalaunch://file:///<username>Session.rsdm">Local RELATIVE
Path</a>
```

```
<a href="mfrumbalaunch://http://<ip_address>/test/
<username>Session.rsdm">Remote Link File Name</a>
```

```
<a href="mfrumbalaunch://http://<username>/test/
MFDspFor<username>.rsdm">Remote Link File Name for <username></a>
```



Note: If you specify the optional parameter `?CreateLocalCopy=true`, a local copy of session profile is created and stored in the user's `<RumbaPrivateDir>\Temp` folder. This local copy is deleted when the sessions closes.

Rumba+ folders

You can access Rumba+ programs from the **Rumba+ Desktop Start** menu folder. This includes the **Rumba+ Desktop Tools** folder. Exactly what programs are available depends on which components you installed.

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

- Rumba+ Desktop APPC Configuration
- Rumba+ Desktop AS400 Communications
- Rumba+ Desktop Communication Monitor
- Rumba+ Desktop Engine
- Rumba+ Desktop Submit Remote Command
- Rumba+ Desktop Trace

Using Rumba+ menus, toolbars, and the status bar

You can Rumba+ commands 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.

Enabling and disabling global commands

Global commands are all Rumba+ Desktop 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 **Plus 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.
Short Name	Displays the HLLAPI short name letter currently being used.
Host Name	Displays the host name.

AS/400 status bar

Status bar area	Function
Description area	<ul style="list-style-type: none"> • 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. • 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.

Status bar area	Function
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.
Cursor coordinates	Lists the row and column position of the cursor.
Host name / IP	The name or IP address of the connected host.
Bell	Indicates whether modern controls available on a given screen..

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.

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).

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

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

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.

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+](#) on page 23.

The supplemental files and their default extensions are:

File type	Extension
Interface configuration	.cfg
Keyboard map	.map
Macro script	.rmc, .js, .vb
Menu configuration	.mnc
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+ 10.2 SP2. 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+ 10.2 SP2.

Configuring session profiles

Any type of session profile can be partially re-configured outside of Rumba+.

The table below shows the parameters that can be configured using the RumbaConfigurator.

M	= Mainframe display	H	= HP
A	= AS/400 display	MP	= Mainframe printer
U	= HP display	AP	= AS/400 printer

Parameter	Desc	Value type	M	A	U	H	MP	AP
RumbaSessionOverriddenParameters								
OverriddenParameters property of RumbaSessionConfiguration object.								
Rumba+ session profile overridden parameters object. All properties are write-only.								
HostName	Overrides configured host name / address	String	Y	Y	Y	Y	Y	Y
Port	Overrides configured host port number	Integer	Y	Y	Y	Y	Y	Y
SSL	Overrides configured connection security (for Rumba+ Security Services only)	Boolean	Y	Y			Y	Y
DeviceName	Overrides configured device / LU name	Integer	Y	Y			Y	Y
InteractiveMode	Overrides configured flag Explicitly Interactive Mode	Boolean	Y	Y			Y	Y
RumbaSSLSessionOverriddenParameters								
SSLOverriddenParameters property of RumbaSessionConfiguration object.								
Rumba+ SSL session profile overridden parameters object. All properties are write-only.								
Applicable to Rumba+ Security Services only.								
If the SSL parameter is set to FALSE, then all parameters are ignored.								
RequireStrictServerAuthentication	Overrides configured flag Require Strict Server Authentication	Boolean	Y	Y			Y	Y
RequireTrustedChain	Overrides configured flag Require Trusted Chain	Boolean	Y	Y			Y	Y
EnforceValidityDates	Overrides configured flag Enforce Validity Dates	Boolean	Y	Y			Y	Y
ExplicitSSL	Overrides configured flag Explicitly Negotiate About Security	Boolean	Y	Y			Y	Y
ClientCertificateName	Overrides configured client certificate name	String	Y	Y			Y	Y
RumbaSessionOverriddenConnectionParameters								
OverriddenConnectionParameters property of RumbaSessionConfiguration object.								

Parameter	Desc	Value type	M	A	U	H	MP	AP
Rumba+ session profile overridden mainframe connection parameters object. All properties are write-only.								
SendKeepAlive	Overrides configured Send Keep Alive	Enum 0 - NoOperation 1 - System 2 - SendNOP	Y					Y
KeepAliveNOPInterval	Overrides configured Send Keep Alive NOP interval	Integer	Y					Y
RumbaSessionOverriddenPlusParameters								
OverriddenPlusParameters property of RumbaSessionConfiguration object.								
Rumba+ session profile overridden Plus parameters object. All properties are write-only.								
AutoComplete	Overrides configured Auto-Complete	Boolean	Y	Y				
SpellCheck	Overrides configured Spell Check	Boolean	Y	Y				
HyperlinksDetectWebAddresses	Overrides configured Detect HTTP and HTTPS Addresses	Boolean	Y	Y				
HyperlinksDetectWebAddresses	Overrides configured Detect Email Addresses	Boolean	Y	Y				
HyperlinksDetectFileAddresses	Overrides configured Detect File Addresses	Enum 0 - NoOperation 1 - ToLastNonSpace 2 - ToFirstSpace	Y	Y				
RumbaSessionOverriddenAPIParameters								
OverriddenAPIParameters property of RumbaSessionConfiguration object.								
Rumba+ session profile overridden API parameters object. All properties are write-only.								
ShortName	Overrides configured Session Short Name	String	Y	Y	Y	Y		
MultipleSessions	Overrides configured Multiple Sessions per Profile	Boolean	Y	Y	Y	Y		
RumbaSessionOverriddenEditParameters								
OverriddenEditParameters property of RumbaSessionConfiguration object.								
Rumba+ session profile overridden Edit parameters object. All properties are write-only.								
TypeAhead	Overrides configured Type Ahead	Boolean	Y	Y				
KeyboardAutoReset	Overrides configured Keyboard Auto Reset	Boolean	Y	Y				
DoubleClickAction	Overrides configured Double-Click Option	Enum 0 - NoAction 1 - Selection 2 - SelectionAndExtra	Y		Y			
WordWrap	Overrides configured Word Wrap	Boolean	Y					
PCInsertMode	Overrides configured PC Insert Mode	Boolean	Y					
PasteInsert	Overrides configured Paste Insert	Boolean	Y					

Parameter	Desc	Value type	M	A	U	H	MP	AP
ParsingCutCopy	Overrides configured Parsing CutCopy	Enum 1 - ParseDataOnFieldBoundaries 2 - ParseDataOnWordBoundaries 3 - DataOnly	Y	Y	Y	Y		
MoveCursorAfterPaste	Overrides configured Move Cursor After Paste	Boolean	Y	Y				
RumbaSessionOverriddenLightPenParameters OverriddenLightPenParameters property of RumbaSessionConfiguration object. Rumba+ session profile overridden LightPen parameters object. All properties are write-only.								
LightPen	Overrides configured Light Pen	Enum 0 - Disabled 1 - Enabled 2 - EnabledAndVisible	Y					

Not all parameters are relevant for all session types. Irrelevant parameters or parameters with invalid values are rejected on load.

Exposed objects, methods, and properties are described in the **Rumba Configuration Objects Library**.

Configuration objects

RumbaConfigurator

The root object creatable by **ProgId MicroFocus.RumbaConfigurator**.



Note: The configurator does not verify the relevance or value of parameters.

Methods:

RumbaSessionConfiguration **OpenSessionProfile(String profile)**

If succeeds, returns Rumba+ session profile object for the specified session profile. Otherwise Nothing / null.

If the **profile** parameter has one of the following values:

- MainframeDisplay
- MainframePrinter
- AS400Display
- AS400Printer
- UnixDisplay
- HPDisplay

the corresponding default session profile is opened (if exists).

If the **profile** parameter does not contain a full path (file name only), the profile is assumed to be found in the default Rumba+ private folder according to the file extension. For example, MFrame for .rdsm for and .rspm, VT for .rsdu.

Sample test case

```
file = <path to existing session profile or profile type>

Set configurator = CreateObject("MicroFocus.RumbaConfigurator")
Set profile = configurator.OpenSessionProfile(file)

If Not profile Is Nothing Then

Set overrides = profile.OverriddenParameters
overrides.HostName = <Name of host>
overrides.Port = <Port>
overrides.SSL = True
overrides.DeviceName = <name of device>

Set SSLOverrides = profile.SSLOverriddenParameters
SSLOverrides.RequireStrictServerAuthentication = True
SSLOverrides.EnforceValidityDates = True
SSLOverrides.ExplicitSSL = False
SSLOverrides.RequireTrustedChain = True
SSLOverrides.ClientCertificateName = <Name of certificate>
SSLOverrides.InteractiveMode = True

Set ConnectionOverrides = profile.OverriddenConnectionParameters
ConnectionOverrides.SendKeepAlive = 2
ConnectionOverrides.KeepAliveNOPInterval = 1000

Set PlusOverrides = profile.OverriddenPlusParameters
PlusOverrides.AutoComplete = False
PlusOverrides.SpellCheck = False
PlusOverrides.HyperlinksDetectWebAddresses = False
PlusOverrides.HyperlinksDetectEmailAddresses = False
PlusOverrides.HyperlinksDetectFileAddresses = 0

Set APIOverrides = profile.OverriddenAPIParameters
APIOverrides.ShortName = "V"
APIOverrides.MultipleSessions = False

Set EditOverrides = profile.OverriddenEditParameters
EditOverrides.TypeAhead = True
EditOverrides.KeyboardAutoReset = True
EditOverrides.DoubleClickAction = 0
EditOverrides.WordWrap = True
EditOverrides.PCInsertMode = False
EditOverrides.PasteInsert = False
EditOverrides.ParsingCutCopy = 1
EditOverrides.MoveCursorAfterPaste = False

Set LightPenOverrides = profile.OverriddenLightPenParameters
LightPenOverrides.LightPen = 0

MsgBox("Done")

Else

MsgBox("Can't open file " + file)

End If
```

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\\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.

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 the relevant session profiles should be contained within the same folder as the desktop profile.

Creating and deploying sessions and settings with the MSS Administrative Console

You can use the Rumba+ Desktop Management and Security Server (MSS) to define terminal emulation sessions, configure and save session settings, then manage and configure secure settings for those sessions.

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

- Session profiles
- Desktop profiles
- Keyboard mapping
- Macros



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 MSS Package Manager to upload the file.
3. Use MSS Assign Access 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.

Creating or modifying a centrally managed session

You can create and manage Rumba+ sessions from the MSS Administrative Console. Sessions that you create in this way are saved to the server and can be made available to users from the server and modified at any time.

For instructions about adding or modifying sessions, see the *Management and Security Server Administrator Guide*, in the [Host Access Management and Security Server Documentation](#).

Making centrally managed sessions available to users

After you have saved sessions to the centralized management server, you can make these sessions available to users who have Rumba+ installed on their workstation. Users can launch these sessions in the same way they launch locally created sessions. When Centralized Management is configured, each time a user opens the workspace:


Rumba+ contacts the server and prompts for user credentials (if required by the server).

Sessions that are available to the user are downloaded to the user data directory.

For instructions about making sessions available to users, see the *Management and Security Server Administrator Guide*, in the [Host Access Management and Security Server Documentation](#).

Using centralized management to deploy MSI Packages

Use the Administrative Console to upload companion install packages (.msi) to the Administrative Server for deployment to specified users. Companion install packages can be created in the Installation Customization Tool or other MSI creation tools, and can include toolbars, macros, keyboard maps, and settings files.

 **Note:** You can simplify deployment of MSI packages by saving your customized sessions as compound session document files.

These packages are automatically deployed to a user's desktop when the user logs on to the Management and Security Server or starts a Rumba+ session with Centralized Management enabled.

For instructions about configuring, uploading, or updating a package, see the *Management and Security Server Administrator Guide*, in the [Host Access Management and Security Server Documentation](#).

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

Legend	Key
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
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

Control character mnemonic	ASCII value	Key pressed with Ctrl (all modes)
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

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~

Legend	Key number	Code sent
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
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)

Control character mnemonic	ASCII Value	Key pressed with Ctrl (all modes)
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
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

Key	Numeric	Application
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).

Configuring mainframe applications with Rumba+ Desktop and Enterprise Server

You can use Enterprise Server 9.0 to specify that particular fields in a mainframe application are required or are sensitive fields.

Required fields do not allow the application to go to the next screen unless data is entered in those fields. Data in sensitive fields is redacted.

You can create required and sensitive fields when using Enterprise Server and Rumba+ Desktop 10.2 SP2.

Using Plus

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

You can use Plus 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 code

Plus controls are created using the Rumba+ Screen Designer by adding controls to captured screen and selecting relevant properties, see below.

Another way to create Plus controls is to use code.

You can use the Rumba+ VBA API to create and access Plus controls, and modify their properties. In addition, events are fired for each Plus control. In this way, you can create more complex and better responsive modern applications by using custom capabilities that are not currently available in the Screen Designer. Validating user input can be performed on specific Rumba+ screens as declared by the user.

This functionality can be an extension for Plus sessions whether or not a Plus archive (.rdar file) is associated with the session. Plus mode must be enabled to support this.

Using the Screen Designer

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

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

Using individual screens

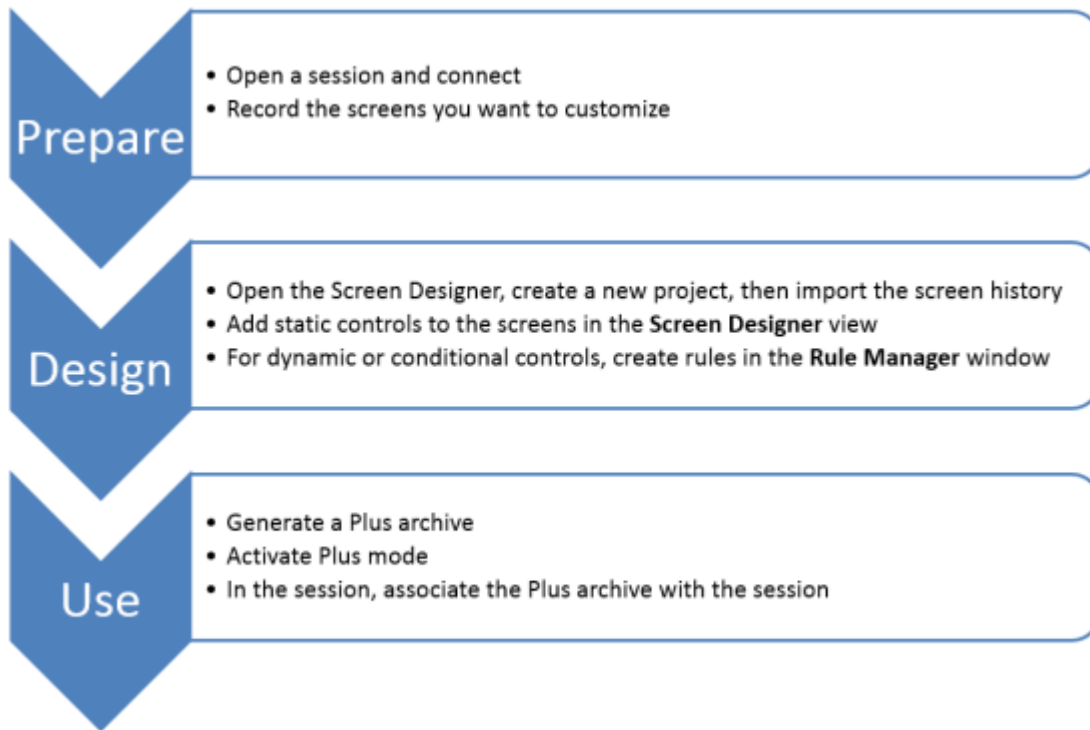
You can identify each screen of the specific host session uniquely and customize each screen individually.

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:



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 Plus 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 generate a customization package file (`.rdar`) from the Screen Designer and associate it with a Plus host session.

Using the Screen Designer command line

You can use the Screen Designer command line to:

- Create one or more Plus archive 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:

<code><Install_Dir></code>	is the folder where the Screen Designer is installed.
<code>/P <project_file_path></code>	is the fully qualified path to a single project file.
<code>/L <project_file_list_path></code>	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.
<code>/O <output_dir></code>	is the fully qualified path of where to export the Plus archives. Optional. If not specified, the <code><Install_Dir>\Plus\ScreenDesigner</code> folder is used.
<code>/S</code>	saves the project after the Plus archive is created.
<code>/I <config_file></code>	is the fully qualified path of the configured list of projects to import.
<code>/?</code>	shows this information on the screen.

Generating Plus archives

Syntax

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

Example: Generating a single Plus archive

```
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 Plus archives

```
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\ul\ul.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  
\al\al.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 Plus

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**.

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+ 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.

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 **OK** 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 **OK** to save the settings for future transfers.

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.

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 <code>rumba.ini</code> 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 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.

This rumba.ini setting	Converts to this Windows registry entry	Purpose
		 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** , then select a Rumba+ 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	Description
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	<p>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.</p> <p>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.</p>
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.
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.

TRANSXX code	Description
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.
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.

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](#) on page 112.

There is more to connecting to a host than just having the physical elements in place. When you start a Rumba+ AS/400 session, 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 > All Programs > Micro Focus Rumba+ Desktop > Rumba+ Desktop Tools > Rumba+ Desktop 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/400 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.

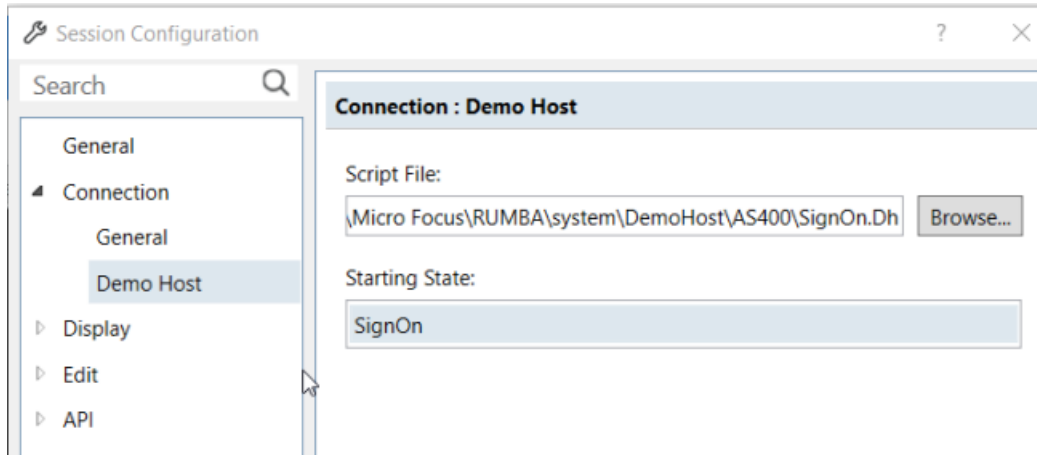
Procedures for configuring each interface are described in the online Help.

The Demo Host

The Demo Host lets you communicate with a simulated host through a series of Rumba+ 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.

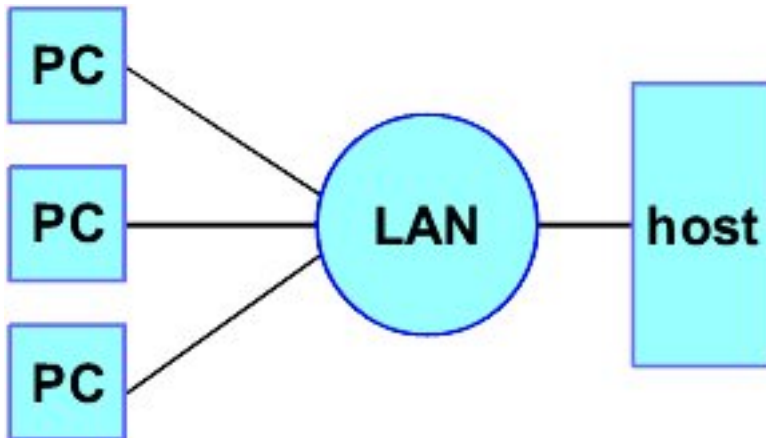
The Demo Host is especially useful for training purposes. You do not have to configure the host or the session parameters to use the Demo Host.



! **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.

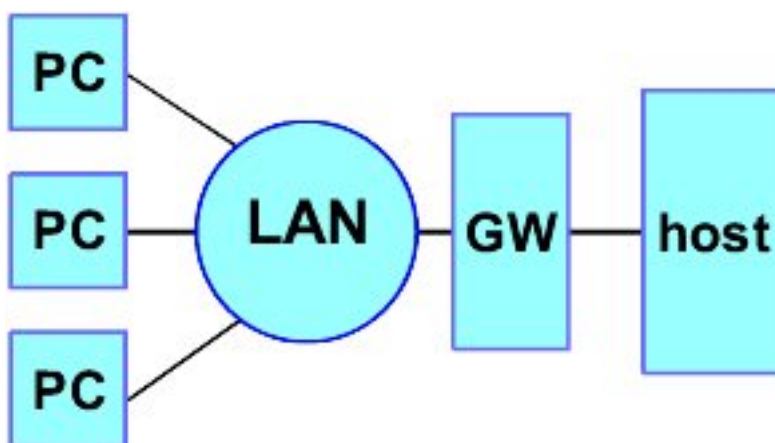
Selecting an interface

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

When 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:

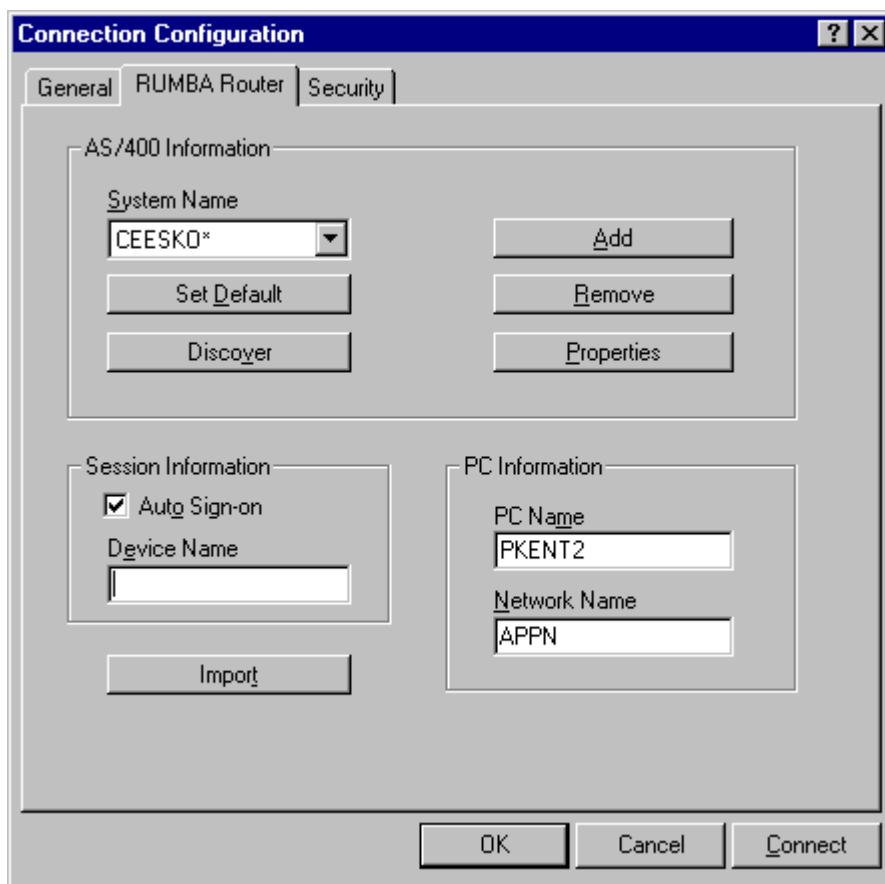
- Demo Host
- Rumba+ Router
- TN5250
- Microsoft SNA Server (if the SNA client is installed)

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.

Parameter	Description
	<p>Remove Deletes a configured connection. If you delete the default system, you must select or configure a new system before connecting.</p> <p>Properties Opens the Link Configuration dialog box for the selected connection. You can edit the available parameters and save the changes to the configuration.</p>
Session Information	<p>Use the controls in this group to specify how user information for this session is specified.</p> <p>Auto sign-on When you check this box, your connection information is used for the host sign-on.</p> <p>Device Name You can add an optional device name that will override the default device name used for this session.</p> <p>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.</p>
PC Information	<p>Lists the information that is used when connecting your PC with the AS/400.</p> <p>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 \$.</p> <p>Network Name Add the name of the network you are using to connect. For example, you may be using APPN.</p> <p>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.</p>

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	
	<p>Use Common Sign-on Uses the specified user ID and password as sign-on information for all systems.</p> <p>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.</p>

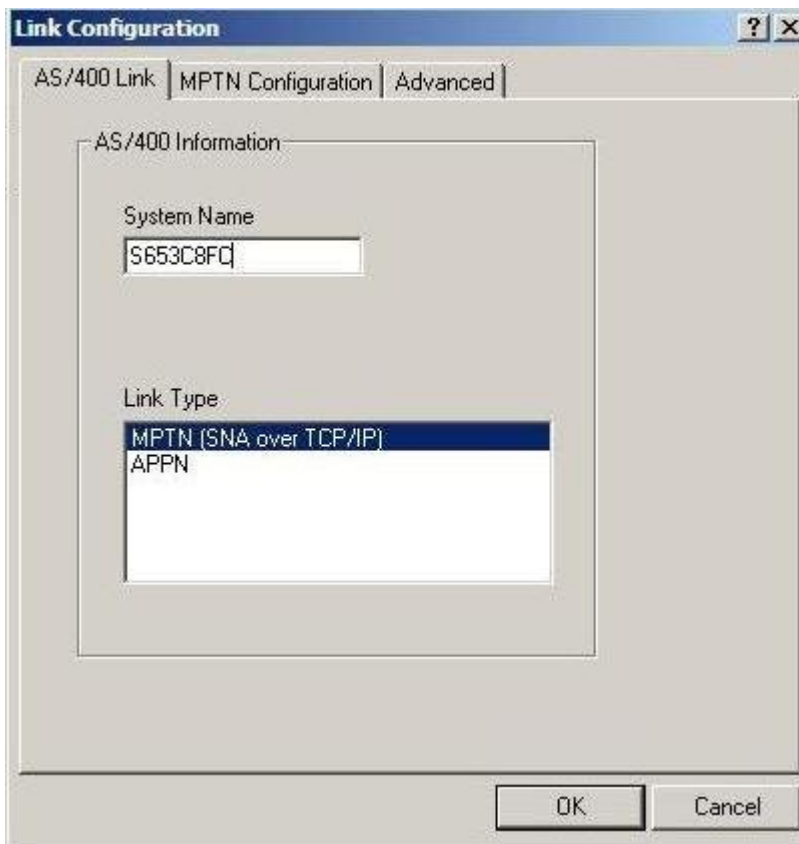
Parameter	Description
Add	Sets up security for a new system.
Change	Changes the security settings for the selected system.
Delete	Deletes the security settings for the selected system.
Enable Caching of User IDs and Passwords	Enables the storing of the user ID and password for the currently configured 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 **Rumba+ Router**.
3. On the **Rumba+ 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	

Parameter	Description
System Name	<p>Displays the alias for the AS/400. This name is added to the list of configured connections on the Rumba+ Router page.</p> <p>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.</p> <p>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.</p>
Link Type	<p>Displays the link types you can use to connect. You must select a link type to complete the connection configuration.</p> <p>Your choice of link type depends on your hardware and software configuration, as well as the type of connection you need. If you do not know which link type to use, check with your system administrator.</p>

Advanced page

The screenshot shows a window titled "Link Configuration" with three tabs: "AS/400 Link", "MPTN Configuration", and "Advanced". The "Advanced" tab is active. Inside the window, there are two text input fields. The first is labeled "LU Name" and contains the text "S653C8FC". The second is labeled "Network Name" and contains the text "APPN". At the bottom of the window, there are two buttons: "OK" and "Cancel".

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

Parameter	Description
LU Name	<p>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.</p> <p>This name can be up to eight characters, including A-Z, 0-9, @, #, and \$.</p>
Network Name	<p>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 \$.</p> <p>This name is used to identify the AS/400 on the network.</p>

Link types

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

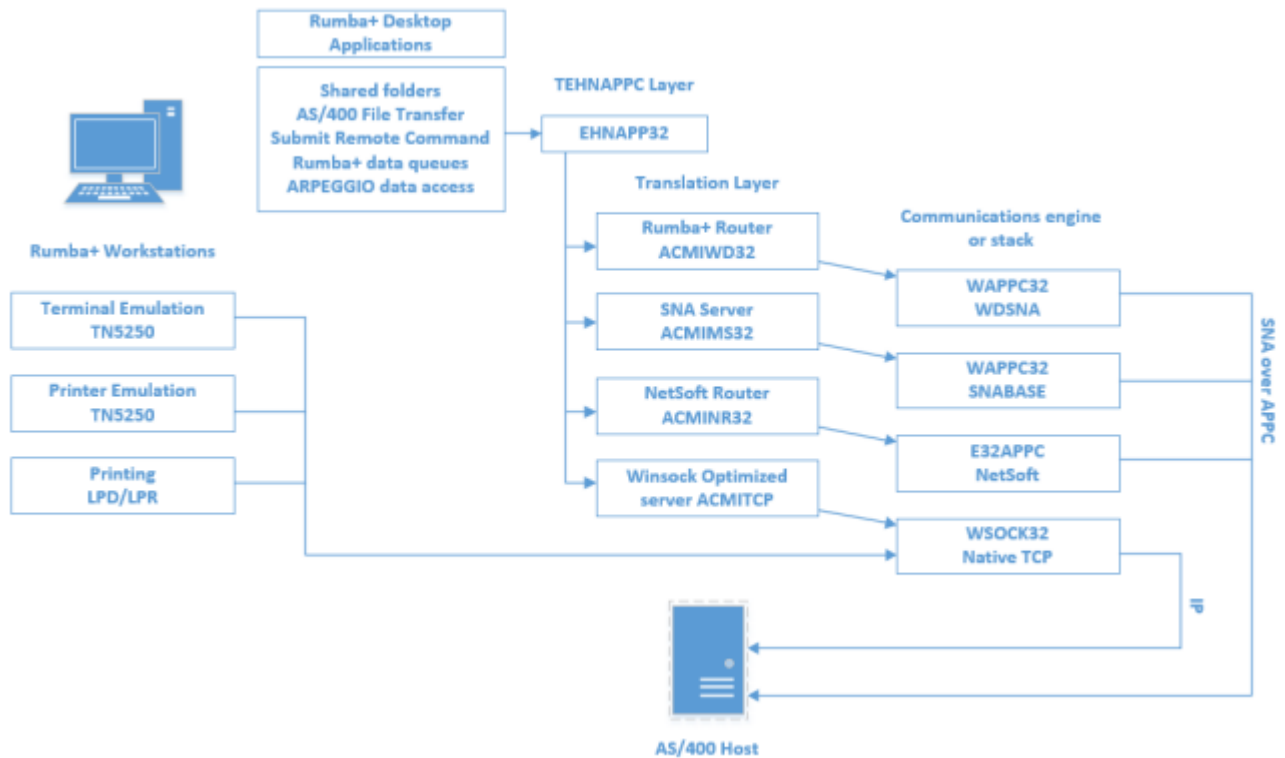
Link Type	Function
APPN	Connects to an AS/400 host by using an existing network connection as a node. For example, an APPN connection between your PC and an AS/400 host can be used to access another AS/400 host that does not have a direct physical connection to your PC.
MPTN (SNA over TCP/IP)	Connects 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+ supports AS/400 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+ 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+ desktop. 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](#) on page 122.

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 MPTN

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

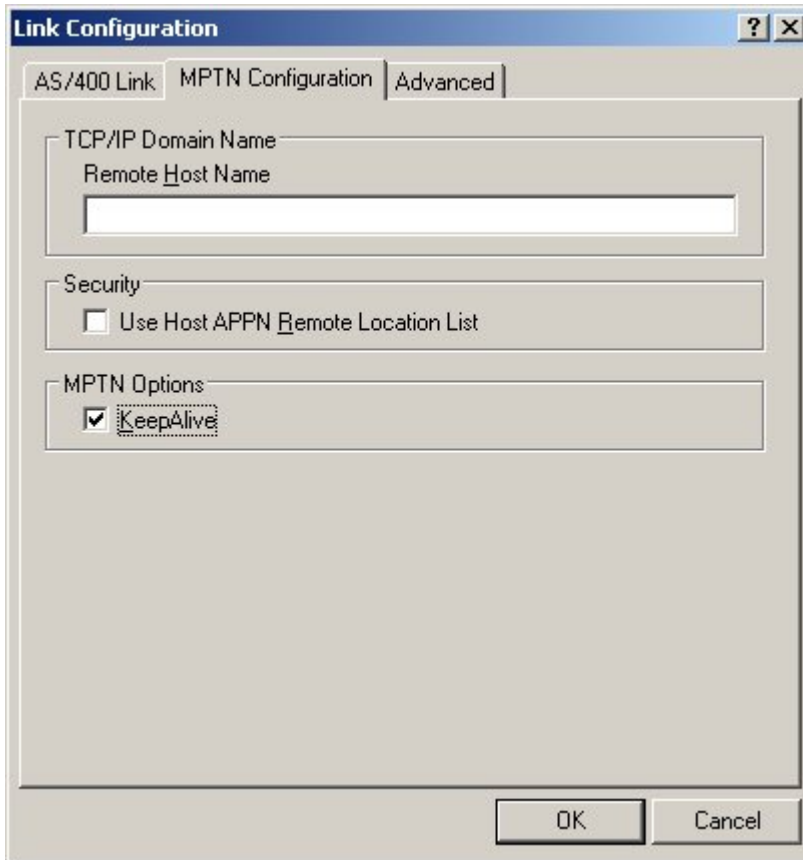
AS/400 considerations

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+ 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 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

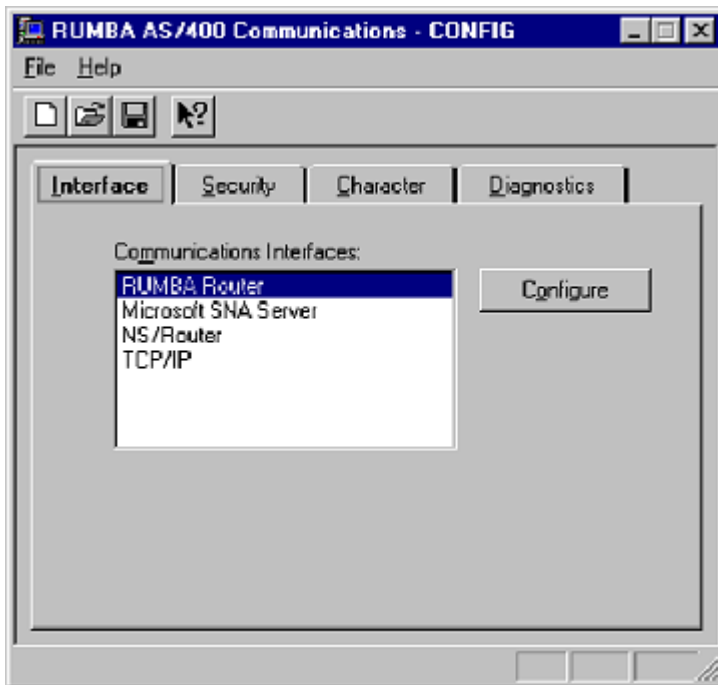
Configurations are saved as a file with an `.ehm` extension. You can create multiple configuration files for each interface that you use.

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.

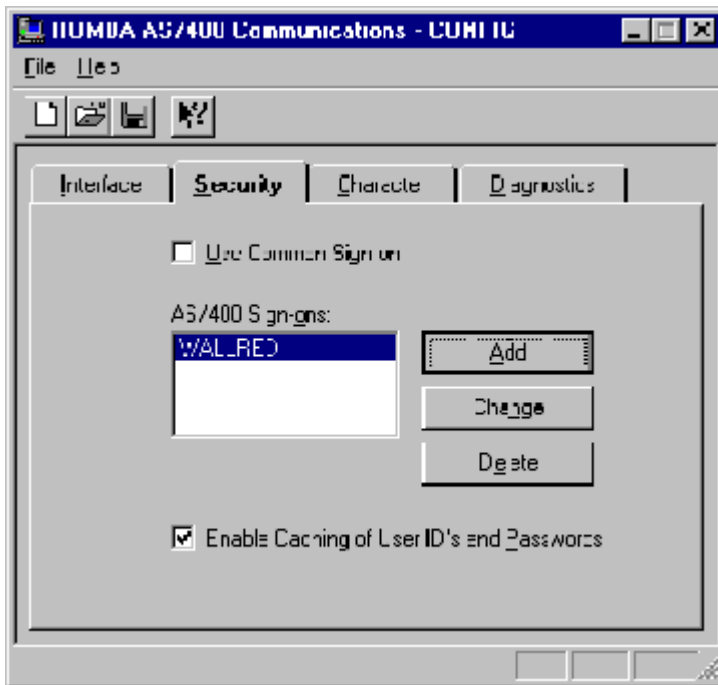
Procedure	Description
Communications Interfaces	Lists the installed communication interfaces. Select which type of communication interface to use when connecting this profile to the host.
Configure	Configures the selected interface (displays the Rumba+ Router dialog box). If the interface has already 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](#) on page 67

[Security page](#) on page 69

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 QDFTJOB (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 QDFTJOB file. You can change the file for the entire system or for a single user.

To change the QDFTJOB file of a single user:

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

To change the QDFTJOB file for the entire system, follow only the instructions [Changing the library list in the QDFTJOB file](#).

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

Copying the QDFTJOB file

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

2. At **Job Description**, type QDFTJOBDB.
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 QDFTJOBDB file

1. At the AS/400 command line, type WRKJOBDB, then press **F4**. The **Work with Job Descriptions** screen appears.
2. At **Job Description**, type QDFTJOBDB.
3. At **Library**, type the name of the new library containing the copied QDFTJOBDB file.
4. Press **Enter**.
5. Under **Opt**, type 2 to make changes to the file, then press **Enter**.

The **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

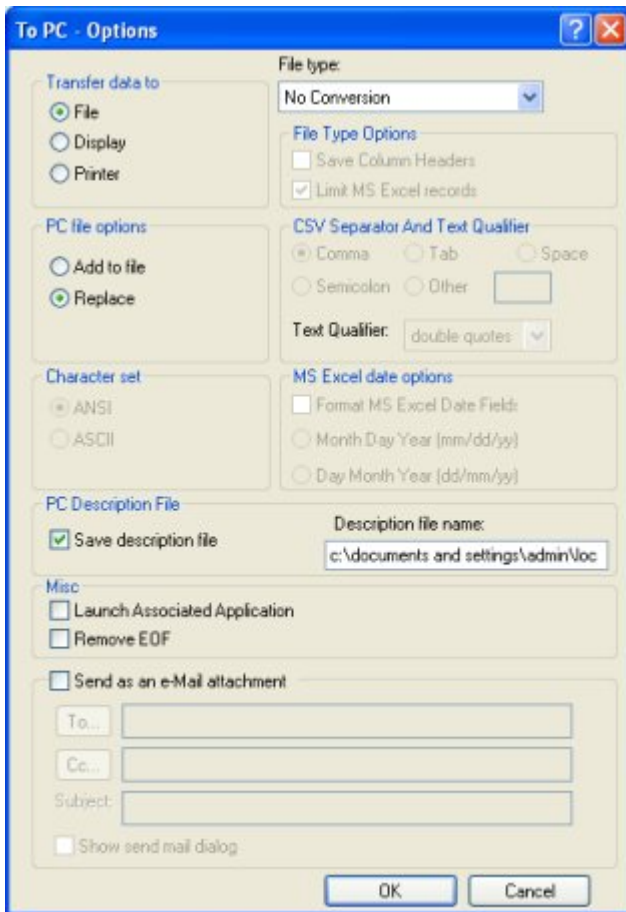
1. At the AS/400 command line, type CHGUSRPRF, then press **Enter**.
2. At **User Profile**, type the name of the profile to change.
3. Press **F10**, then press **Page Down** twice.
4. Under **Job Description**, at **Library**, type the name of the library containing the new QDFTJOBDB file, then press **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,* Option Settings	Host default date and time formats
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 to send specific AS/400 data. 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.

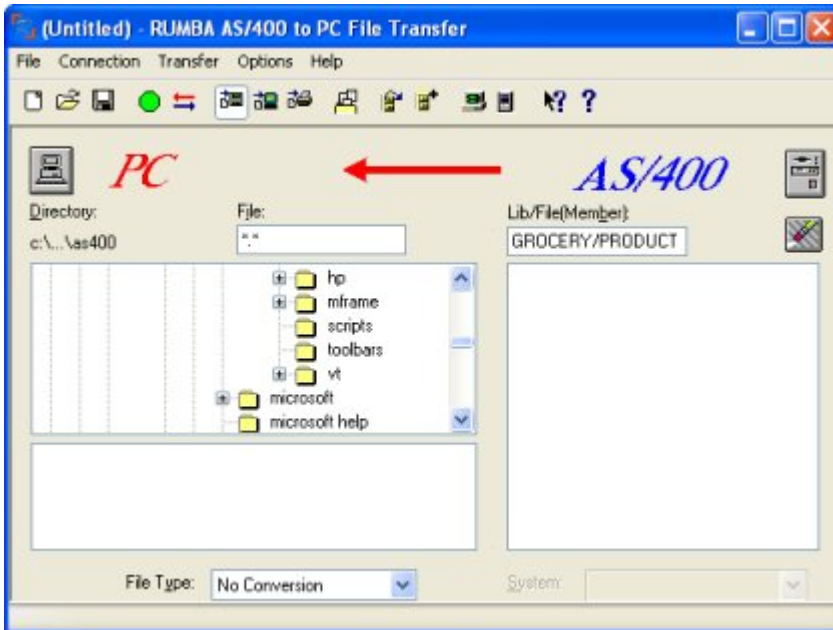
ProdCat	ProdID	ProdDesc	ProdName	ProdQuant
100570265	129930113	Wild Berry	Summerbrook Wild Berry Seltzer	84

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:

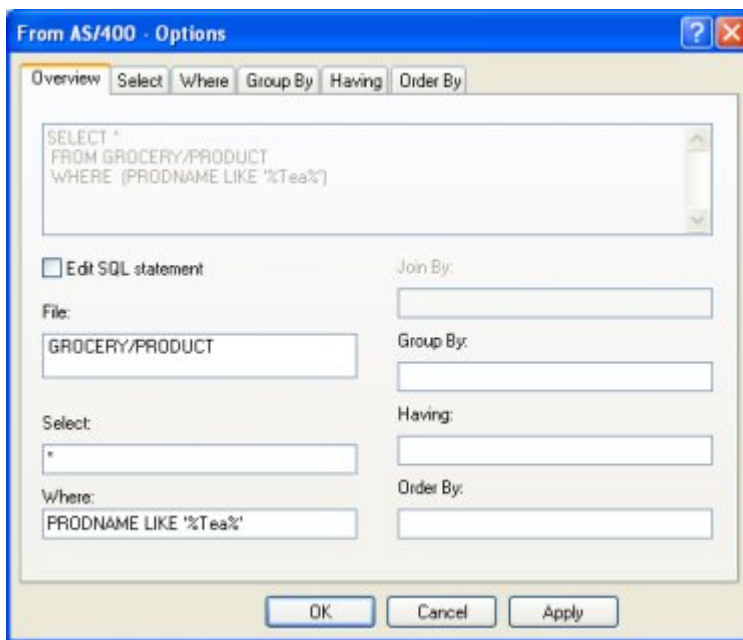
1. Specify the GROCERY/PRODUCT file in the **Lib/File(Member)** field on the main file transfer screen, then open the **From AS/400 - Options** dialog box:



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:

PRODDDESC

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

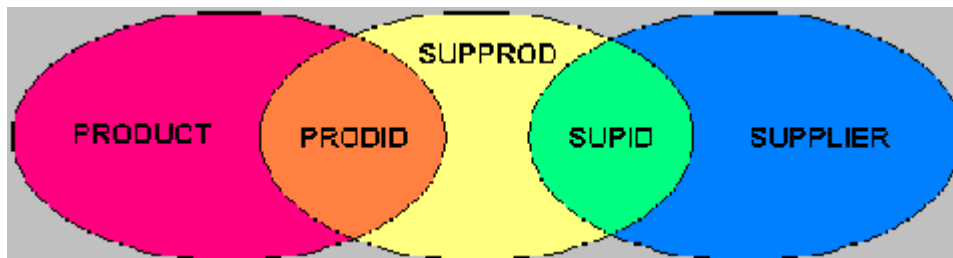
```
SELECT *
FROM grocery/product
WHERE (PRODNAME LIKE '%Tea%')
ORDER BY PRODDDESC
```

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+ Desktop 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 > All Programs > Micro Focus Rumba+ Desktop > Rumba+ DesktopRumba+ Tools**. Submit Remote Command is also a legacy API and available as an ActiveX control in the Rumba+ ObjectX Development Kit.

HP and UNIX File Transfer

Rumba+ gives you easy, complete, and transparent access to HP and UNIX host applications.

How to transfer files using Rumba+ HP and UNIX

You can use the Rumba+ HP file transfer feature, to copy files from a PC to an HP host or from an HP host to a PC.

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

Rumba+ 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+ uses protocol-driven transfer methods. The specific details of the protocols are not important to users, but the protocols specified on the PC must match those used by the host computer. For example, Rumba+ does not work if the host attempts to transfer using the Kermit file transfer protocol and the user selects **Text** as the transfer protocol to be used. To minimize such conflicts, we strongly recommend using the **Rumba+ Initiated** option (see below) and using macros (see [Using Rumba+ macros to transfer files](#) on page 94).

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

1. Select **Transfer > Send** or **Transfer > Receive**.
2. On the **Setup** page, specify the file type, transfer protocol, and host environment. You can also have Rumba+ initiate the file transfer operation by checking **Rumba+ Initiated** (see below). For example, `sz` is used for sending files to the host via Zmodem.

When you choose a file transfer protocol, Rumba+ provides a configuration page 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, then specify the information needed by the transfer protocol.
4. Click the **Files to Transfer** tab and select the files you want to transfer.
5. Click **OK**.

About the Rumba+ Initiated option:

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

When the option is checked:

- You can specify the name of the host transfer program for Rumba+ to use when automatically initiating the transfer.
- Rumba+ 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.
- If the correct host is not specified, you will not see any host file information when you view the **Files to Transfer** tab.

The **Rumba+ Initiated** option is not available for **Text** file transfers or when **Other** is selected in the **Host Environment** list.



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

MPE file transfers

For HP sessions only, file transfer capabilities are now available for systems using the MPE host environment. When you are 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 box appears, a new button labeled **BootStrap** will be visible.
3. Click **BootStrap** to transfer the `WALLXFER.BIN` configuration file to the MPE system.
4. After the `WALLXFER.BIN` file is on the host, set its file permissions to execute (`x`) and move it to the `PUB.SYS` group.
5. 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 must be done once for each MPE system you will use to transfer files.

Using compatible file transfer protocol versions

Rumba+ might have trouble supporting particular 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 implement the protocols incorrectly. Ensure 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

File transfers are performed 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. 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.

AS/400 and Mainframe Printers

This section provides a general overview of the Rumba+ AS/400 and mainframe printers.

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 host 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, 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 try 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 selecting **Devices and Printers** from the Windows **Start** menu or 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

This describes how you can:

- 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

Mainframe only

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 goes to the printer without any preview.

Printer sharing and End print job

Mainframe only

Use the **Printer sharing** option (default) or the **End print job** option to specify how Rumba+ recognizes the end of a host print job.

With printer sharing, the current print job is released when Rumba+ 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+ waits for the user-specified timeout period, then releases 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+ 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

AS/400 only

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. Select **Connection > Configure**, then select **Rumba Router** from the **Installed Interfaces** list.
2. Click the **Rumba+ Router** tab.
3. Click **Transform**.
4. In the **Host Transform** frame, select **Enabled**, then change the configuration parameters as needed.
5. Click **OK**, then **OK** in the **Connection Configuration** dialog box.

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

-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://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.

Administrative installation

1. Double-click the appropriate Rumba_<...> .exe file. For example, Rumba_Office.exe. Provide the <absolute_path> to where the file contents should be extracted.
2. Cancel all following wizards.
3. To install Rumba+ as an administrative installation, use the following command:

```
msiexec /a <absolute_path>\<Rumba_<...>.msi [TARGETDIR=<path>] [/q[n|b|r|f]]
```

Using Macros

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

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. Plus

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.

VBA support provides the ability to write more complex and powerful automation. VBA support uses standard VBA functionality, IDE and allows you to run, record, and edit macros from and to a common Macro Module shared between all sessions.

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:

- Select **Tools > Record Macro**, then perform your keystrokes and commands. Be sure to save the sequence as a macro file after recording.
- Select **Tools > New Macro > EML / RMC / VBA**, then build a sequence of commands with the appropriate editor.

To combine the speed of recording macros with the power of written command lines, record the basics, 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.

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.

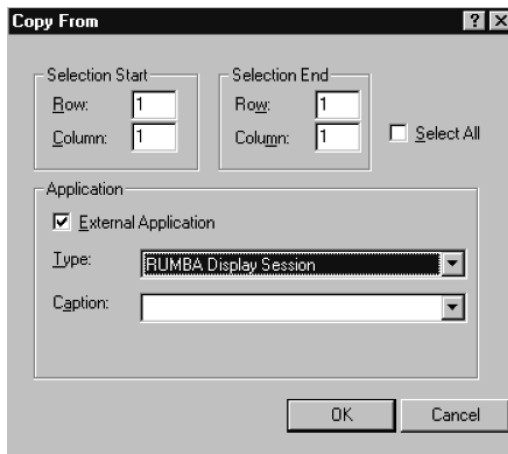
Rumba+ Macro Editor


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. 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 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.

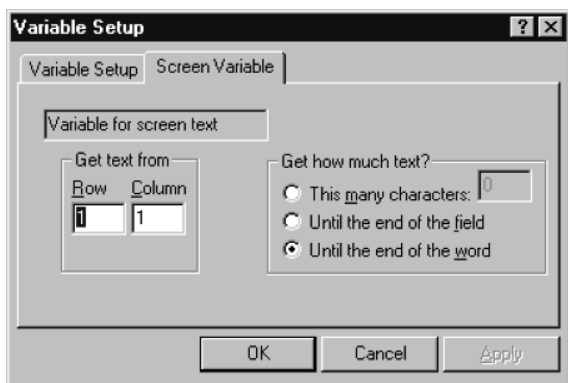
For example, you could create a prompt dialog box to request a password from a user, then save it as a variable. The variable containing the user password is then available for any **Type**, **Compare Values**, and **Message** commands you have included in the macro sequence. For example, you could use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

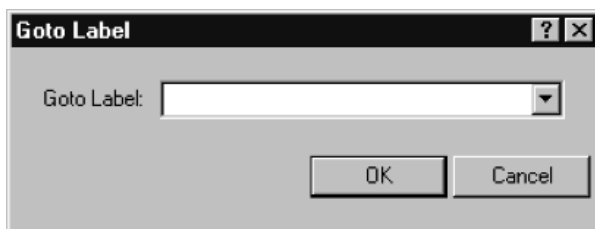
1. Select **Tools > New Macro > RMC**.
2. In the **Macro Name** field, type `Flow Example`.
3. Right-click in the macro display area, and choose **Variable Setup** from the pop-up menu to open the **Variable Setup** dialog box.
4. Select **Variable for screen text** from the **Variables** list.
5. Select **Screen Variable** from the **Variable Types** list.
6. Click the **Screen Variable** tab.



7. Change the **Get text from Column** value to 78.
8. In the **Get how much text?** frame, select **This many characters**, and type 3 in the box.
9. 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.

10. Open the **Goto Label** dialog box.



11. Select **Variable for screen text** from the list, then click **OK**.
12. Open the **Message** dialog box. Type `Screen Unknown` and click **OK**.
13. Open the **Goto Label** dialog box. Type `Macro End` and click **OK**.
14. Open the **Label** dialog box. Type `A00` and click **OK**.
15. Open the **Message** dialog box. Type `PROFS Main Screen` and click **OK**.

16. Open the **Receive** dialog box.
 - a) Type `Mail Waiting` in the **Find text** box.
 - b) Under **Screen Position**, select **Only at row, column**, and type 24 and 69 in the **Row** and **Column** boxes.
 - c) In the **Timeout** frame, select **Seconds**, and type 30 in the box.
 - d) In the **Otherwise** box, type `NoMail`.
 - e) Click **OK**.
17. Open the **Message** dialog box. Type `You have mail` and click **OK**.
18. Open the **Label** dialog box. Type `NoMail` and click **OK**.
19. Open the **Goto Label** dialog box. Type `Macro End` and click **OK**.
20. Open the **Label** dialog box. Type `C00` and click **OK**.
21. Open the **Message** dialog box. Type `You are on the Open Mail Screen` and click **OK**.
22. Open the **Goto Label** dialog box. Type `Macro End` and click **OK**.
23. Open the **Label** dialog box. Type `E01` and click **OK**.
24. Open the **Message** dialog box. Type `You are reading a message` and click **OK**.
25. Open the **Label** dialog box. Type `Macro End` and click **OK**.
26. Add `Exit Macro` to end of macro.
27. Select **File > Save**, then close the Macro Editor.

Your completed macro should look like this:



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

- Select **Tools > Run Macro**, then select the macro you want to run.
- Associate a macro with a Hotspot, then click the Hotspot text on the host screen to run the macro.
- Assign a macro to a toolbar button, then click the button to run the macro.
- Link a macro to a keystroke, 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, 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.

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.

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 stored in the `%LocalAppData%\Micro Focus\Rumba\hostname` folder as `.map` files. To distribute keyboard files, you must distribute the `.map` file and any corresponding macro files (`.rnc` and `.mac`) that are linked to mapped keys.

Macro files are stored in `%LocalAppData%\Micro Focus\Rumba\hostname\Macro` folder

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.

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.

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.

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.

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.

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.

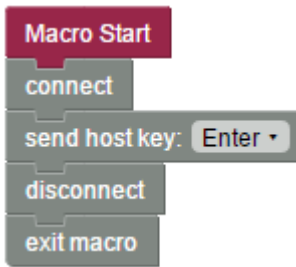
Type

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 Rocket Software 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**.
2. Perform the steps you want to record, such as logging on and opening a host application.
3. To stop recording, select **Tools > Record Macro** again.

The **Save Macro** window appears.

4. In the **Macro Name** field, type a name for the macro.
5. Select **EML** from the drop-down list.

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+ Desktop 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**.

The **Select a Macro** dialog box appears.

2. Select the **EML** macro you want to run, then click **OK**.

The macro runs.



Tips:

- To interrupt a macro while it's playing, select **Tools > Cancel Macro**.
- To assign runtime properties to your macros, select **Tools > Macro Properties**.

VBA macros

You can use Visual Basic for Applications (VBA) to create macros for automatic navigation. To do this, VBA uses Rumba+ automation application and session objects to access and handle multiple emulation sessions. VBA macros can perform emulation session related tasks, such as connecting, typing, screen text retrieving, and executing navigation sequences.

You can also use the VBA API to create and access Plus controls, modify their properties, and fire events. For further information, see the Rumba+ VBA API Help.

Connection Tools

Rumba+ Desktop Tools provides you with 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
- Rumba+ administrator override file
- Rumba+ AS/400 Communications
- Rumba+ Communication Monitor
- MSS Automated Sign-On



Note: For details of additional Rumba+ Tools, see the online Help.

The following summarizes the purpose and supported hosts for each of these administrator utilities:

- Rumba+ APPC Configuration
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
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.
- MSS Automated Sign-On
Configures automated sign-on for mainframe 3270 sessions.
Applies to mainframe.

Rumba+ APPC Configuration

The APPC Configuration tool helps you build 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 figure below shows the **Rumba+ APPC Configuration** window:

Because the Rumba+ APPC Configuration tool 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 tool 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.

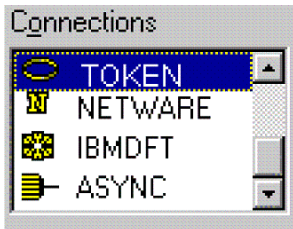
With the Rumba+ APPC Configuration Tool, you can set values for different local and remote environments. For CPI-C, you can add or modify side information entries. You also have the option of assigning user IDs and passwords to incoming remote transaction programs.

Connections

Use the Rumba+ APPC Configuration tool to name connections and add interfaces quickly and without error.

Working in the background, the tool creates a local LU name (logical unit name) for your connection from your network name and control point name. As a default, the tool inserts your computer name for the control point name.

The Rumba+ APPC Configuration tool allows you to easily add an interface connection. The **Connections** box allows you to identify the interface used for your connection.



International character set translation

The Rumba+ APPC Configuration tool works with the international character set specified in the regional settings for your workstation. This translation is accomplished using codepage tables stored in the `Program Files(x86)\Micro Focus\Rumba\System\Charsets` folder.

These tables work with Rumba+:

- The character set that Rumba+ uses is determined by the country that is specified in the > **Region and Language** window.
- EBCDIC to ANSI translation (or mapping) for your workstation is based on the codepage table currently in use.
- The `Program Files(x86)\Micro Focus\Rumba\System\Charsets` folder contains files called `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, use a text editor to modify the corresponding `chr2exxx.gen` file. 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 tool 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 tool simplifies the task of managing remote and local logical unit (LUs). A local LU represents the local workstation, while the a LU represents the local workstation conversation partner.

LU 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 tool to add or change logical units:

- For a computer client, the LU name default can be the computer name. The Rumba+ APPC Configuration tool 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

You can use the Rumba+ APPC Configuration tool to easily create and edit side information files. Previously, when using APPC/LU 6.2 verbs, 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 tool 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 (TP) by naming and associating it with an executable file or command line. The TP can then have:

- A path to the executable program.
- An auto-start option.
- Optimum security.
- Queueing.

- Timeouts.
- PreLaunch Counts.

The TPs you add with the Rumba+ APPC Configuration tool must 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 TP, use the Rumba+ APPC Configuration tool'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.

Description

An administrator override file is a configuration file you create, then place on the network. Your 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.

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 file on your network. The next time a user pulls up the configuration dialog in the AS/400 display or Rumba+ APPC Configuration tool, the new AS/400 appears, configured and ready to go.

 **Note:** The administrator override file does not contain the control point name that is used for the connection. This name comes from the user's machine.


Function

The Rumba+ SNA engine, AS/400 Display, and Rumba+ APPC Configuration tool can all look at two configuration files at the same time and extract pertinent information from each file.

When you configure connections to an SNA-type server, the Control Point Name (also known as the "PC Name" in the AS/400 environment) makes one workstation unique from another workstation. The administrator override file contains the link information. This includes, but is not limited to the:

- Network name.
- Destination address.
- Remote LU.
- Side information.

Use the local configuration file to store information applicable only to the workstation.

 **Note:** Mainframe connection definitions are not supported by the administrator override file. To understand why, consider the following:

- Connection definitions for the Rumba+ Router and AS/400 display are shareable 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 shareable 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.

Local connection configuration

You can still configure connections locally even if the administrator override file is set. You can enable administrator override in the Rumba+ APPC Configuration tool 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. See [Configuration files precedence](#).

Also, similar to document printing, the AS/400 display shows only connections configured in the administrator override file, and users can only connect to these administrator override configured connections.

File 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.

Creating a file

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 file on a shared network drive.

1. From any machine that already has Rumba+ installed, configure the necessary connections using the Rumba+ APPC Configuration tool.
2. Save the file as you normally would for a local machine (note the file name in the title bar).
3. Ensure 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. Ensure that each intended user has access to this drive.



Note: Change the file attributes to R/O, This prevents users from accidentally corrupting the file.

Editing existing files

If an administrator override configuration file already exists and is currently being accessed by users, we recommend the following as the safest editing method:

1. Copy the current administrator override configuration file and rename it.
2. Change the R/O attribute (if set) to R/W so that the changes you make can be saved.
3. From the Rumba+ APPC Configuration tool menu bar, select **Open**. Select the configuration file.
4. Make the changes you want.
5. Save the file.
6. Once you are confident that you have a working configuration file, replace the existing administrator override configuration file with the new one.



Note: Change the administration override configuration file's attributes to R/O. This will prevent a user from accidentally corrupting the file.

Setting user access

1. From the **Rumba+ Desktop Tools** program group, start the Rumba+ Desktop APPC Configuration tool.
2. Select **File > Advanced**.
3. Check **Use Administrator Override Configuration**.
4. Enter the full path of the configuration file that you want the users to access, or click **Browse** to select it.
5. Optional: To ensure that the next person gets the same configuration, select **Make configuration available to all users of this machine**. This uses the `\\HKEY\LOCAL_MACHINE\` registry instead of the `\\HKEY\CURRENT_USER` registry.
6. Select **File > Save**.
7. Close the APPC Configuration tool.

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 does not rely on a mapped drive letter that could be overwritten or corrupted.



Notes:

- If you use **Browse**, the path is returned in UNC format.
- If the Rumba+ APPC Configuration tool cannot find the file you specified in the path for the administrator override file, a warning is displayed. The situation might 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 so, and the path is correct, choose to continue and exit.
- From the operating system's ComputerName, the Rumba+ APPC Configuration tool extracts a default control point name for the local configuration file. If you want to, you can specify another control point name by simply changing the field. Any changes you make to the control point name are also made automatically to the local LU name.
- You must use a `.cfg` extension on the file name. If you do not specify an extension, the Rumba+ APPC Configuration tool appends `.cfg` to the file name. The Rumba+ APPC Configuration tool supports both 10.2 SP2 format file names and long file names.

Assessing workstation setup

The best tool to use to determine if you have set up a workstation correctly is the Rumba+ Communications Monitor.

If you selected the APPC Client package install, Rumba+ Communications Monitor appears in the **Rumba+ Desktop Tools** program group. If the APPC Client was not installed, Rumba+ Communications Monitor is still installed, but does not appear in the program group. In this case, you can run the program directly:


```
C:\Program Files(x86)\Micro Focus\Rumba\System\nof.exe
```

Rumba+ Communications Monitor starts the Rumba+ SNA engine and displays the information that the Rumba+ SNA engine interprets. Rumba+ Communications Monitor should display the connections that you configured in the administrator override file.

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 box allows users 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, the administrator override file still works.

Configuration-free workstations

With Rumba+ Tools, you can have self-configuring workstations. By using SCCM or a batch file, each new workstation looks for its:

- Administrator override file.
- Local configuration file.

Rumba+ AS/400 Communications

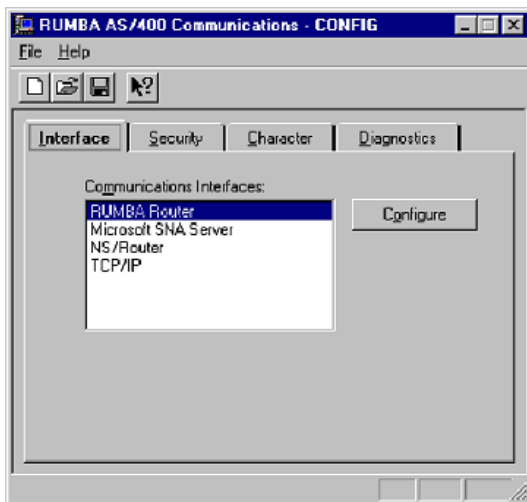
You can use Rumba+ AS/400 Communications to locate other AS/400 systems and to configure or change host connections and link interfaces for the:

- Application Program Interface (API).
- Submit Remote Command application and API.

Overview


The configuration settings for the AS/400 Communications tool are saved in a `.ehh` file. You can save multiple configurations under different `.ehh` files for later use. The configuration file can be especially useful when you are setting up data queues for users who do not have Rumba+ AS/400 display or printer capability.

The figure below shows the **AS/400 Communications - CONFIG** dialog box:



The table below maps the major tasks you perform with the AS/400 Communications tool to the appropriate page in the dialog box:

To ...	Go to ...
Configure communication interfaces	Interface page
Configure link interfaces	Interface page
Add AS/400 sign-ons	Security page
Select a foreign language's host character set	Character page
Display APPC error messages	Diagnostics page

 **Note:** If you use Rumba+ AS/400 Communications to configure EHNAPPC, all non-display and non-printer sessions 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.

Configuring links

To use Rumba+ AS/400 Communications to configure links:

1. Click the **Interface** tab.
2. Select a communications interface, then click **Configure**.
For specific information on configuring, refer to the online Help.
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. Select **File > Save**.

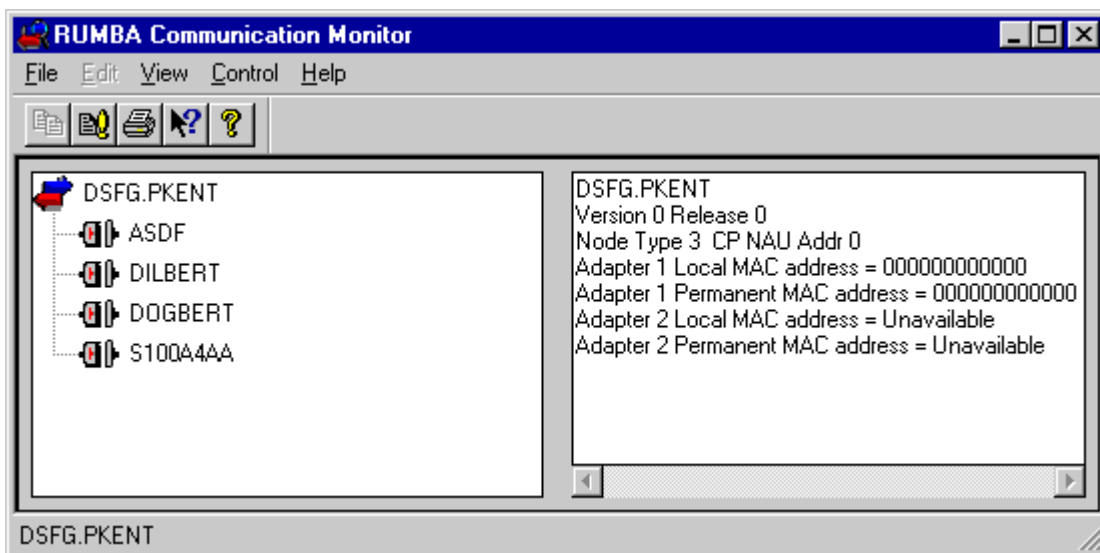
By default, the configuration is saved as an `.ehn` file.

Click the **Advanced** tab to specify an LU with a name different than your system's LU.

Rumba+ Communication Monitor

Rumba+ Communication Monitor helps you monitor, troubleshoot, and optimize your network. Also, you can view non-configured system links.

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: You can use Rumba+ Communication Monitor to start and stop links (either by the menu or by right-clicking your mouse). However, 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+ Desktop 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-AutoSignInProvider: 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 **Manage Sessions** in the navigation pane.
3. In the **Manage Sessions** 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 **Assign Access** 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+ Tools** program folder provide you with a set of powerful and cost-effective features that are designed to boost your productivity.

Rumba+ Trace

Rumba+ Trace combines several trace options in a single tool. When you troubleshoot a problem for a user, you can ask them to trace and record communication sessions with Rumba+ Trace. Later, you can selectively play back the binary data file to troubleshoot the problem. This section discusses the following aspects of working with Rumba+ Trace:

- API options
- Difficult troubleshooting tasks
- Display software code version
- Playback display and printer data stream trace

By default, all 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 Rocket Software.

API options

Rumba+ Trace provides 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)
- Rocket Software 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, ensure you select **Options > Display**. This is selected by default.
- Only 1000 lines of the most recent data is displayed.

- 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 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. Start Rumba+ Trace.
2. Select **Options > Configuration**.
The **Configure Tracing Parameters** dialog box appears.
3. Click **Select All**.
4. Click the **Output** tab, then check **Binary File**.
5. In the **Binary Output** field, 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.
8. Select **Trace Options > Configuration**.
9. On the **Output** page, uncheck **Binary File**.
10. Click the **API Selection** tab, choose the APIs you want to analyze.
11. Click the **Advanced** tab. Under **Binary File Replay**, type the name of the file you saved (or browse button to it).
12. Click **Replay Now** to view the API data flow.

Displaying software code version

Rumba+ Trace contains a useful feature that can give you an all-inclusive software code version report that covers `.cfg`, `.dll`, `.exe`, and `.ini` files:

Select **Options > Code Version Snapshot**.

If you use this feature, you might want to run it either before or after you complete your trace, because this report is extensive and takes time to run.

Playing back display and printer data stream trace

You can use Rumba+ Trace to record the AS/400 or mainframe display and printer data streams. You can then analyze and change the data, as well as add rules of behavior on key actions and change labels. This feature is especially useful for troubleshooting problems, or when you need to create a demonstration host for training or marketing purposes.

1. Start Rumba+ Trace.
2. Select **Options > Configuration**.
3. On the **API Selection** page, select a single display or printer data stream to record.



Note: The Demo Host can play back only one output file at a time, so ensure that you only choose one script at a time.

4. Click the **Output** tab, check **Formatted File**, then type 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+ feature you want to record.

7. Close Rumba+ Trace then start a Rumba+ display or printer session.
8. Select **Connection > Configure**.
9. Select **Demo Host** from the **Installed Interfaces** list.
10. Click the **Demo Host** tab.
11. In the **Script file** field, type in the path and name of the file you saved (or you can use the **Browse** button to navigate to it).
12. Click **Connect** to demonstrate the feature or problem you recorded.

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+ 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 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 extension 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.



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 Extension for Visual Studio

This extension 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 extension 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 extension also provides:

- Rumba+ project templates for different session types
- Integration with the Visual Studio Toolbox for adding 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.

Before installing the extension



Prerequisites:

- If you have Visual Studio 2012, 2013, 2015, or 2017 installed, you must install the Microsoft .NET Framework 4.7.1 Developer Pack.
- If you have Visual Studio 2019 installed, you must install Microsoft .NET Framework 4.7.2 Targeting Pack.
- If you have Visual Studio 2022 installed, you must install Microsoft .NET Framework 4.8 Targeting Pack.

Installing RDE

1. Close all open instances of Visual Studio.
2. Run `RumbaDevelopersEditionVisualStudioAddIn.msi` and follow the instructions in the Setup wizard.
3. Open each supported Visual Studio version you want to use the RDE from, then enable the Micro Focus RDE extension via Extensions Manager.
4. Close Visual Studio, then re-open it to start working with RDE.

The RDE extension is installed per user to each supported Visual Studio version detected on the machine.

You can remove the extension from the machine by choosing **Start > Control Panel > Programs and Features**.

To install RDE to all supported versions of Visual Studio, use the following command:

```
msiexec /i <full_path_to_RumbaDevelopersEditionVisualStudioAddIn.msi> /qn
```

Silent uninstall

To uninstall the extension, use the following command line:

```
Msiexec /x RumbaDevelopersEditionVisualStudioAddIn.msi /qn
```

Rumba+ Desktop 64-bit applications support

Rumba+ Desktop integrates 64-bit components allowing you to continue using your current RDE or HLLAPI applications in a Windows 64-bit environment.

For detailed information, see the *Rumba+ 64-Bit* guide.

Notices

Copyright

© 1996-2024 Rocket Software, Inc. or its affiliates. All Rights Reserved.

Trademarks

Rocket is a registered trademark of Rocket Software, Inc. For a list of Rocket registered trademarks go to: www.rocketsoftware.com/about/legal. All other products or services mentioned in this document may be covered by the trademarks, service marks, or product names of their respective owners.

Examples

This information might contain examples of data and reports. The examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

License agreement

This software and the associated documentation are proprietary and confidential to Rocket Software, Inc. or its affiliates, are furnished under license, and may be used and copied only in accordance with the terms of such license.

Note: This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when exporting this product.

Corporate information

Rocket Software, Inc. develops enterprise infrastructure products in four key areas: storage, networks, and compliance; database servers and tools; business information and analytics; and application development, integration, and modernization. Website: www.rocketsoftware.com

Contacting Technical Support

The Rocket Community is the primary method of obtaining support. If you have current support and maintenance agreements with Rocket Software, you can access the Rocket Community and report a problem, download an update, or read answers to FAQs. To log in to the Rocket Community or to request a Rocket Community account, go to www.rocketsoftware.com/support. In addition to using the Rocket Community to obtain support, you can use one of the telephone numbers that are listed above or send an email to support@rocketsoftware.com.

Rocket Global Headquarters
Rocket Global Headquarters
77 4th Avenue, Suite 100
Waltham, MA 02451-1468
USA

Country and Toll-free telephone number

To contact Rocket Software by telephone for any reason, including obtaining pre-sales information and technical support, use one of the following telephone numbers.

- United States: 1-855-577-4323
- Australia: 1-800-823-405
- Belgium: 0800-266-65
- Canada: 1-855-577-432

- China: 400-120-9242
- France: 08-05-08-05-62
- Germany: 0800-180-0882
- Italy: 800-878-
- Japan: 0800-170-5464
- Netherlands: 0-800-022-2961
- New Zealand: 0800-003210
- South Africa: 0-800-980-818
- United Kingdom: 0800-520-0439

ROCKET SOFTWARE END USER LICENSE AGREEMENT

IMPORTANT: ROCKET SOFTWARE, INC. OR ITS AFFILIATE DESIGNATED IN THE ENTITY TABLE LOCATED AT www.rocketsoftware.com/company/legal/AMC-assigned-entity ("LICENSOR") PROVIDES LICENSED SOFTWARE TO LICENSEE (HEREINAFTER "CUSTOMER") UNDER THIS END USER LICENSE AGREEMENT (THE "AGREEMENT"). THIS AGREEMENT GOVERNS CUSTOMER'S INSTALLATION AND USE OF THE VERSION OF THE LICENSED SOFTWARE IDENTIFIED IN THE APPLICABLE PRODUCT ORDER, OR IF NOT ACQUIRED VIA A PRODUCT ORDER, CUSTOMER'S DOWNLOAD AND INSTALLATION OR USE OF THE LICENSED SOFTWARE CONSTITUTES ACCEPTANCE OF THIS AGREEMENT. **PLEASE READ THIS AGREEMENT CAREFULLY, AS IT CONTAINS RESTRICTIONS ON CUSTOMER USE OF THE SOFTWARE. THIS AGREEMENT SUPERSEDES AND CONTROLS OVER ANY OTHER TERMS PROVIDED TO CUSTOMER REGARDING CUSTOMER'S USE OF THE LICENSED SOFTWARE, WHETHER WRITTEN OR ORAL, UNLESS A DIFFERENT WRITTEN AGREEMENT IS EXPRESSLY REFERENCED IN A PRODUCT ORDER OR EXECUTED BY LICENSOR AND CUSTOMER THAT SUPPLEMENTS OR SUPERSEDES ALL OR PORTIONS OF THIS AGREEMENT.** ENTERING INTO THIS AGREEMENT DOES NOT CONSTITUTE A SALES TRANSACTION.

1. DEFINITIONS. Capitalized terms in this Agreement are defined as follows:

"Additional License Authorization" or **"ALA"** means the additional specific software license terms that govern the use of a given software product, including (but not limited to) the provisions of the Non-Production Licensing Guide. The ALA(s) for Licensed Software are found at <https://www.rocketsoftware.com/company/trust/agreements> by product name, or can be provided by Rocket Software at Customer's request.

"Customer" or **"Licensee"** means the legal entity or individual that is identified in the applicable Product Order or who has rightfully received a license to the Licensed Product.

"Documentation" means the user documentation Rocket Software makes available for Licensed Software.

"Licensed Product" means Licensed Software and Documentation.

"Licensed Software" means the executable version of the software listed in the Product Order or otherwise provided to or rightfully acquired by Customer. This Agreement will govern the use of any update to the Licensed Software that Customer receives pursuant to a separate support and maintenance agreement as described in the Support and Maintenance section below, unless such update contains, comes with, or is otherwise specifically governed by a different end user license agreement.

"Open Source Software" means software or other material embedded in or made available with a Licensed Product where such software or other material is made available under an "open source license" as this term is commonly understood by members of the open source community, including, but not limited to, those licenses meeting all the criteria prescribed in the Open Source Definition provided by the Open Source Initiative (<https://opensource.org/osd>).

"Product Order" means the combination of a Rocket Software Quote and a purchase order or purchase order alternative agreed to by the parties for the sale of a license(s) of one or more specific items of Licensed Software to Customer.

"Rocket Software" or **"Licensor"** means the applicable Rocket Software entity listed on the Entity Table.

"Rocket Software Quote" is a quote issued by Rocket Software or its authorized billing agent which contains the one or more specific items of Licensed Software, quantity, price and other pertinent license details.

"Third Party Component" means any third party run time or other elements embedded in Licensed Software other than Open Source Software.

"Third Party Software" means additional or accompanying third party software (other than Open Source Software or Third Party Components) that is specified in the Documentation or in a file accompanying Licensed Software.

"Warranty Period" means the 90-day period starting from the date Licensed Software is delivered to Customer.

2. PRODUCT ORDERS. The licensing of software products takes place under Product Orders which (unless stated otherwise in the Product Order) incorporate the terms of this Agreement. Any terms that conflict or are in addition to the terms of this Agreement or any applicable ALA ("Inconsistent Terms") in a purchase order or other document issued by Customer are rejected by Rocket Software and have no force and effect. If there is any Inconsistent Term(s) in a Product Order, such Inconsistent Term(s) shall not apply except where such Inconsistent Term(s) is contained in a Rocket Software Quote or a Product Order that is signed by both parties. Open Text Corporation("OpenText") and its affiliates are temporarily providing quoting and invoicing service for certain Rocket Software products and services and is acting as a billing agent for Rocket Software until further notice. Rocket Software will provide relevant billing and payment information as and when the transition to Rocket Software invoicing and billing systems occurs. Rocket Software is not otherwise affiliated or associated with Micro Focus or OpenText. "Micro Focus" and "OpenText" are registered trademarks of OpenText or its affiliates.

3. LICENSING.

a. License. Except as specifically permitted in the applicable ALA, or as described in the Evaluation Licenses section below, Rocket Software and its affiliates, respectively deliver and license to Customer the Licensed Products under a non-transferable, non-sublicensable, non-exclusive license to use Licensed Software and its Documentation as set out in this Agreement and/or the applicable ALA and solely for Customer's internal business operations and not for further distribution or commercialization.

b. Evaluation Licenses. Except as specifically permitted in the applicable ALA, when Rocket Software and its affiliates, respectively deliver and license the Licensed Products solely for evaluation, Customer receives a non-transferable, non-sublicensable, non-exclusive license to use Licensed Products for internal evaluation and testing purposes only, and not for any development, production, distribution or commercial purpose ("**Evaluation License**"). The term of an Evaluation License will be 30 days starting from the date Licensed Product is delivered (i.e., made available for download or physically delivered) to Customer ("**Evaluation Term**"), unless Rocket Software authorizes a different period in writing. The Licensed Product is provided "as is" and there are no warranties or obligations for Rocket Software to provide support. The Evaluation License terminates at the end of the Evaluation Term, and Customer is required to return, or, if Rocket Software so directs, delete and destroy, all copies of such Licensed Product and provide Rocket Software with written confirmation of its compliance with this provision within 30 days of the end of the Evaluation Term. The Evaluation License for any prerelease or beta versions of Licensed Software ("**Pre-Release Software**") shall be for a term of 90 days unless Rocket Software authorizes a different period in writing. Customer agrees to promptly report to Rocket Software all problems (including errors, failures, nonconforming results, and unexpected performances) and any comments regarding the Pre-Release Software and to timely respond to all questionnaires submitted by Rocket Software regarding the results of Customer's testing of the Pre-Release Software. Rocket Software may choose not to release a final version of the Pre-Release Software or, even if released, to alter prices, features, specifications, capabilities, functions, release dates, general availability, or other characteristics of the Pre-Release Software.

4. USE RESTRICTIONS.

a. Except as specifically permitted in the applicable ALA or Documentation, Customer will not, directly or indirectly:

- a. Use Licensed Software for time-sharing, outsourcing, hosting, service bureau or like use, or permit access by, or use for the benefit of, any third party;
 - b. Modify or create derivative works of Licensed Software;
 - c. Reverse engineer, decrypt, disassemble or otherwise attempt to discover the source code of Licensed Software, except to the extent permitted by applicable law;
 - d. Unbundle component parts of Licensed Software for separate use, where Licensed Software with multiple components is provided to Customer as a single product;
 - e. Publish or disclose to third parties any evaluation or benchmarking of Licensed Software; or
 - f. Alter, destroy, or remove any proprietary notices or labels on or embedded in Licensed Software.
- b. Customer may make a reasonable number of archival copies of Licensed Software and Documentation, and will reproduce all copyright and other proprietary rights notices appearing in or on Licensed Products, including notices of all third party suppliers, on all permitted copies.
- 5. TERM.** This Agreement and the license term for the Licensed Software granted herein is perpetual, unless a subscription/term license has been purchased by Customer (in which case the license term shall be set forth in the Product Order or ALA), and is subject to earlier termination as provided in the Termination section below. If Customer has purchased a subscription/term license, such license shall automatically terminate upon expiry of such subscription/term, unless earlier terminated in accordance with this Agreement.
- 6. TERMINATION.** Except as set forth herein, either party may terminate this Agreement, and/or any license granted, on written notice if the other party materially breaches the terms of this Agreement, or any applicable ALA or Product Order and does not cure the breach within 30 days of the date such party is notified of such breach. Rocket Software may terminate this Agreement, along with any or all licenses then in effect with Customer, immediately by giving Customer written notice of termination in the event that (i) Customer becomes insolvent, has a receiver appointed, or files for or has filed against it, liquidation, bankruptcy or analogous proceedings; or (ii) Customer infringes or misappropriates the intellectual property rights of Rocket Software. Termination shall be without prejudice to any other rights or remedies a party may have. In the event of any termination, Customer's license(s) to install, access or use the Licensed Software will immediately terminate, and Customer shall destroy and erase all copies of such Licensed Software in its possession or control and provide written certification to Rocket Software that it has complied with this provision. Early termination of this Agreement or any Product Order will not entitle Customer to any credit or refund or reimbursement of any previously paid fees.
- 7. SUPPORT AND MAINTENANCE.** Customer is not entitled to any updates to Licensed Software, unless Customer purchases (or may receive under a subscription/term license) maintenance and support services pursuant to Rocket Software's then current applicable standard maintenance and support agreement (which is found at <https://www.rocketsoftware.com/company/trust/agreements> or can be provided by Rocket Software at Customer's request).
- 8. HARDWARE.** In the event that Rocket Software provides Licensed Software embedded in an appliance, or provides hardware to Customer for use in conjunction with Licensed Software, additional hardware terms and conditions will apply. In all other instances, Customer will obtain and install any necessary hardware for proper installation and implementation of Licensed Software.
- 9. PROFESSIONAL SERVICES.** If Customer has Rocket Software or a Rocket Software affiliate perform any services with respect to Licensed Software (e.g., installation, implementation, maintenance, consulting or training), Rocket Software or its affiliate will provide those services at its then current standard terms and conditions and rates unless otherwise agreed in writing by the parties. Customer may agree to enter into a professional services engagement with Rocket Software or its affiliate through a separate agreement or statement of work.
- 10. LIMITED WARRANTY.** Rocket Software warrants for the Warranty Period that the Licensed Software substantially conforms in all material respects to the Documentation, and any media on which Rocket Software provides Licensed Software is free from defects in materials and workmanship under normal use.

Customer's sole and exclusive remedy for breach of such warranty is (i) repair or replacement of the applicable Licensed Software or media free of charge so that it substantially conforms to Documentation or (ii) if Rocket Software reasonably determines such remedy is not economically or technically feasible, a refund of the license fee and any maintenance fee paid for the Licensed Software for the current year.

The license to use such Licensed Software will immediately terminate when Customer receives the refund.

The warranties described herein do not apply if the defects in Licensed Software or media result from: (i) failure to use Licensed Software in accordance with Documentation, this Agreement or the applicable ALA; (ii) malfunction of Customer's equipment or network; (iii) accident, neglect, or abuse; (iv) service by any unauthorized person; (v) other software used by Customer not provided by Rocket Software, or for which Licensed Software is not designed or licensed for such use; or (vi) any other cause occurring after initial delivery of Licensed Software or media to Customer, unless directly caused by Rocket Software. For purposes of this Agreement, Licensed Software is deemed delivered when it is first made available for download by, or physically delivered to, Customer.

Rocket Software has no responsibility for any claims made outside the Warranty Period.

The warranties described herein also apply to Third Party Components, but do not apply to (i) any free-of-charge Licensed Software or updates provided following the Warranty Period; or (ii) third party software that is not a Third Party Component.

11. DISCLAIMER OF WARRANTY. Except for the limited warranties herein, Licensed Products are provided "as is" without warranty of any kind. To the extent permitted by law, all implied or statutory terms, conditions, representations, and warranties (including without limitation all terms, conditions, representations and warranties regarding merchantability, quality or fitness for a particular purpose, title or non-infringement, or that may arise out of course of dealing, usage or trade practice) are expressly disclaimed and excluded to the extent allowed by applicable law. The limited warranties in this Agreement are provided on the basis that Customer is procuring Licensed Products for the purposes of a business and not for household or consumer use. Rocket Software does not warrant that operation of Licensed Software will be uninterrupted or error-free. Customer has sole responsibility to select Licensed Software that works with other software, applications or systems to achieve intended results.

12. LIMITATION OF LIABILITY.

- a. **Liability Cap.** In no event will any liability of one party to the other party arising out of or in connection with this Agreement exceed the total fees paid by Customer for the affected Licensed Software in the relevant Product Order(s) (including related maintenance and support). Nothing in this Section shall limit either party's liability for: willful misconduct or fraudulent misrepresentation; unauthorized use of intellectual property; breach of license; death or bodily injury caused by negligence; non-payment of amounts owed; or any liability which may not be excluded or limited by applicable law.
- b. **Mutual Disclaimer of Consequential Damages.** In no event will either party be liable for any indirect, special, incidental, consequential, punitive or similar damages; loss of profits, business, data or programs (including, but not limited to, the cost of recovery or replacement of such data or programs); loss, damage or any costs due to interruption, delay or inability to use Licensed Software, whether arising out of or in connection with this Agreement, even if informed of the possibility of such damages in advance.
- c. **Scope.** The limitations and disclaimers herein apply to all causes of action, including but not limited to, breach of contract, breach of warranty, negligence, strict liability, misrepresentation and other torts.
- d. **Exclusive Remedy.** The remedies in this Agreement are the parties' exclusive remedies and the limitations of the Liability Cap section and the Mutual Disclaimer of Consequential Damages section apply even if these remedies fail of their essential purpose. Customer may have other rights, including consumer rights, under the laws of the state or country where it is located.
- e. **Free-of-Charge Software.** If Rocket Software provides Customer with any Licensed Software free-of-charge or under an Evaluation License, to the extent permitted by law, Rocket Software and its affiliates will not be responsible for any loss or damage to Customer, its customers, or any third parties caused by that Licensed Software which is free-of-charge.

13. OWNERSHIP. Rocket Software, Inc. and its affiliates and their suppliers own or have the right to use or further distribute all intellectual property rights in Licensed Products, and authorize Rocket Software to distribute Licensed Products under this Agreement. Customer's only rights in Licensed Products are the express licenses referenced in this Agreement or the applicable ALA.

14. OPEN SOURCE SOFTWARE AND THIRD PARTY SOFTWARE. Open Source Software and Third Party Software are governed by the terms of their respective licenses and not the terms of this Agreement, notwithstanding anything to the contrary in any applicable ALA. Information about the Open Source Software may be found in a file accompanying the Licensed Software or in the Documentation or ALA.

15. LICENSE FEES AND PAYMENT TERMS. Customer will pay the license fees set forth in the applicable Product Order for Licensed Software within 30 days of the date of invoice. Software license fees are non-refundable, except as provided herein, and will be paid without deduction or tax withholding. Software License fees are exclusive of any transportation charges, sales, use, value added, and other applicable taxes and duties, and all such amounts will be paid or reimbursed by Customer. Customer will be liable for all outstanding past due amounts, which will accrue interest (at the rate of 1.5% per month compounded or the maximum rate permitted by law if lower) and any collection costs for recovery of past due amounts.

16. LICENSE VERIFICATION. Rocket Software has the right to verify Customer's compliance with licenses to the Licensed Software.

Customer agrees to keep records sufficient to show Customer's compliance, including serial numbers; license keys; logs; records identifying machines on which Licensed Software is installed, accessed, or from which Licensed Software can be accessed; the number of different users accessing or authorized to access Licensed Software (if applicable); and licensing metrics, reports, and copies of Licensed Software. Rocket Software may require Customer to provide information about its deployment of Licensed Software, which may be in the form of a questionnaire. Customer will have a reasonable time to complete the questionnaire (or other form of request) and provide it to Rocket Software with an authorized Customer signatory certifying the accuracy of the information furnished. Upon 10 days prior written notice, Rocket Software or its designated representative may inspect Customer's records, systems and facilities, during Customer's normal business hours, to verify compliance with licenses to the Licensed Software. Customer agrees to cooperate with such verification. Information obtained will be used solely for compliance purposes and will otherwise be subject to the confidentiality provisions of this Agreement. If Customer has engaged in unlicensed installation, use of, or access to Licensed Software or infringed or misappropriated intellectual property rights in Licensed Software, or has otherwise breached this Agreement or an ALA (a "Non-Compliance"), without prejudice to Rocket Software' other rights or remedies, including, without limitation, injunctive relief, within thirty (30) days' notice of such Non-Compliance, Customer agrees to purchase sufficient licenses and or subscriptions and associated support and maintenance (12 months forward and any applicable back support and maintenance) to cure the Non-Compliance, by paying Rocket Software's current (as of the date of such additional purchase) list license fees and support and maintenance fees to Rocket Software, plus interest (compounded at 1.5% monthly or the maximum rate permitted by applicable law if lower) for such additional licenses from the commencement of the Non-Compliance until payment of the aforementioned fees, with interest payable even if an invoice was not issued at the time the Non-Compliance occurred. If Customer's Non-Compliance results in an underpayment of license fees of 5% or greater, Customer shall also reimburse Rocket Software for the reasonable cost of such audit in addition to other amounts due.

17. PRIVACY. Customer is solely responsible for, and assumes all liability with respect to, its own collection, processing, storage, and transfer of any user data, including personally identifiable information, personal health and financial information, and other forms of personal information (collectively, "**Personal Information**"). Customer is solely responsible for notifying its users of proper use of such data. Customer is solely responsible for assessing the Licensed Product or any related product or service for compliance with any industry requirements applicable to Customer. Each party must comply with its respective obligations under all applicable laws, regulations, and industry standards regarding data collection and data privacy that apply to use of Licensed Software. Personal information or customer data that Customer provides to Rocket Software for the purpose of receiving products or services will be handled in accordance with the Rocket Software Global Data Protection and Privacy Notice (<https://www.rocketsoftware.com/company/legal/privacy-policy>) and the privacy and data security provisions in the applicable ALA.

18. USE OF CUSTOMER INFORMATION. To the extent permitted by law, Customer expressly consents to the collection and use of information about Customer's purchase, installation and use of Licensed

Software and the computer systems on which it is installed or accessed as necessary to fulfill the Product Order and provide the Licensed Product, for security and licensing purposes and to improve Rocket Software products and services.

19. CUSTOMER FEEDBACK. Customer agrees that, should it provide Rocket Software with any feedback or suggestions regarding Licensed Products ("**Feedback**"), Rocket Software and its affiliates will be free to use all such Feedback, including (but not limited to) all intellectual property rights in and to such Feedback, in their discretion and for any purpose, without obligation of any kind to Customer.

20. CONFIDENTIAL INFORMATION. Information exchanged under or in connection with this Agreement, including all information pertaining to Pre-Release Software, will be treated as confidential if identified as confidential at time of disclosure, or if the circumstances of disclosure reasonably indicate such information should be treated as confidential. Confidential information may only be used for the purpose of fulfilling obligations or exercising rights under or in connection with this Agreement, and shared with employees, affiliates, agents or contractors with a need to know such information to support that purpose, who are contractually bound to protect its confidentiality. Confidential information will be protected using a reasonable degree of care to prevent unauthorized use or disclosure for three years from date of receipt. These obligations do not cover information that (i) was known or becomes known to the receiving party without a duty of confidentiality; (ii) is independently developed by the receiving party; (iii) becomes publicly available without breach of this Agreement; (iv) is disclosed with the disclosing party's prior written consent; or (v) where disclosure is required by law, a court or government agency. Should the receiving party be required to reveal confidential information under subpoena, court order or other operation of law, the receiving party will provide reasonable prior notice to the disclosing party and request a protective order, if available.

21. INDEMNIFICATION FOR INTELLECTUAL PROPERTY RIGHTS INFRINGEMENT. Rocket Software will defend and/or settle any claims against Customer that Licensed Products furnished under this Agreement infringe a third party's patent, copyright, or trade secret rights. existing under the laws of a country where Customer is using the Licensed Product ("IP Infringement Claim"), provided: (i) Customer promptly notifies Rocket Software in writing of the IP Infringement Claim; (ii) Rocket Software has sole control of the defense and all related settlement negotiations; and (iii) Customer reasonably cooperates with Rocket Software in the defense of the IP Infringement Claim.

Rocket Software will pay all damages, costs, and expenses finally awarded (or agreed to by settlement) for any such IP Infringement Claim. Rocket Software will pay all reasonable out-of-pocket costs incurred by Customer for cooperation in the defense of the IP Infringement Claim. If, however, Customer wants separate legal representation, Customer will be responsible for the costs and fees of its separate counsel.

Should any Licensed Products provided under this Agreement become or, in Rocket Software's opinion, likely become, the subject of an IP Infringement Claim, Rocket Software may replace or modify affected Licensed Product so as to make it non-infringing and materially equivalent, or procure for Customer the right to continue using it. If neither alternative is reasonably available, once Customer has returned or destroyed all copies of the affected Licensed Product, Rocket Software will refund to Customer the full amount paid for affected Licensed Product less straight line depreciation on a five-year basis from date of delivery.

Rocket Software is not responsible for unauthorized use of Licensed Products, and has no obligations under this Section (Indemnification for Intellectual Property Rights Infringement), to the extent infringement results from (i) compliance with Customer's designs or instructions, (ii) a modification not authorized in writing by an authorized Rocket Software signatory, (iii) use or combination with software, equipment, or data not provided by Rocket Software, (iv) non-licensed use; or (v) Third Party Software or Open Source Software.

22. MISCELLANEOUS.

a. Assignment. Rocket Software may assign this Agreement and any Product Orders to an affiliate or a third party in connection with an asset sale or divestiture. Customer may not assign or transfer this Agreement (or any Product Orders) or any of its rights or duties hereunder, including (but not limited to) by operation of law, without Rocket Software's prior written consent, which will not be unreasonably withheld, and payment of any applicable assignment or transfer fee. Any transaction or series of related transactions resulting in a change in the ownership of more than 50% of the voting

equity of Customer will be deemed an assignment for purposes of this provision. Any attempted assignment of this Agreement (or a Product Order) not in accordance with this section (Assignment) will be null and void.

- b. Governing Law and Jurisdiction.** This Agreement and, subject to the Open Source Software and Third Party Software Section, licenses purchased under applicable Product Orders, as well as any claims or causes of action, whether in contract, tort or statute, based on, arising under or relating to this Agreement, will be governed and enforced as follows: If Customer's applicable place of business is located in North America, all matters arising out of or relating to this Agreement are governed by the laws of the state of Delaware, U.S.A. If Customer is located in the United Kingdom, Australia, Brazil, France, Germany, Italy, Japan, the Netherlands, New Zealand, Spain or Singapore, all matters arising out of or relating to this Agreement are governed by the laws of the country in which Licensee is located. In the rest of the world, the laws of England and Wales govern all matters arising out of or relating to this Agreement. The applicable law shall apply without regard to conflicts of law provisions, and without regard to the United Nations Convention on the International Sale of Goods. Any suit, action, or proceeding arising out of or relating to this Agreement shall be subject to the exclusive jurisdiction of the courts of the country determining the applicable law, except that the courts of the State of Delaware shall have exclusive jurisdiction in North America, and Rocket Software will be allowed to apply for injunctive relief in any jurisdiction. Each party agrees to be subject to the above-described jurisdiction and waives any right it may have to object to such venue, including objections based on personal jurisdiction or inconvenient forum. The prevailing party in any proceeding has the right to recover costs and reasonable attorneys' fees as awarded by the court or arbitrator.
- c. Export Control.** Rocket Software and Customer will comply with their respective responsibilities as exporters and importers under laws and regulations applicable to the export, import or other transfer of the software, services and technology provided by Rocket Software under this Agreement, including the export, import and sanctions laws of the United States and other applicable jurisdictions. If Customer exports, imports, or otherwise transfers any Licensed Products provided under or in connection with this Agreement (or any technical data related thereto), Customer is responsible for obtaining any required authorizations. Customer will not use Licensed Products for any purpose prohibited by applicable export laws, including nuclear, chemical, missile or biological weapons-related end uses. With respect to any Pre-Release Software provided to Customer, Customer represents and warrants that (i) it is a non-governmental entity, (ii) the Pre-Release Software will only be used for internal testing and evaluation, and will not be rented, leased, sold, sublicensed, assigned, or otherwise transferred, and that it will not transfer or export any product, process, or service that is the direct product of the Pre-Release Software, and (iii) it will only use the Pre-Release Software in the United States or the countries listed in Title 15, U.S. CFR Supplement No. 3 to Part 740 - License Exception ENC Favorable Treatment Countries. Customer agrees to indemnify and hold Rocket Software harmless from and against any damages, claims, losses, fines, settlements, attorneys' fees, legal fees and court costs and other expenses related to any such activities or any claims in connection with any breach of this Section.
- d. Survival.** Rights and obligations of the parties in the following Sections: Use Restrictions, Term, Termination, Disclaimer of Warranty, Limitation of Liability, Ownership, Open Source Software and Third Party Software, License Fees and Payment Terms, License Verification, Privacy, Use of Customer Information, Customer Feedback, Confidential Information, Indemnification for Intellectual Property Rights Infringement, and Miscellaneous will survive termination or expiration of this Agreement. Neither party will be liable for delays or non-performance beyond its reasonable control except for payment obligations.
- e. Notices.** All notices permitted or required under this Agreement will be in writing and delivered personally, by courier, telecopy, first class mail, electronic mail, or similar transmission to the other party. Notices to Rocket Software shall be addressed to: Legal Department, Rocket Software via email to legal@rocketsoftware.com. Notices to Customer may be sent via electronic mail to any email address set forth in the Product Order or to its address listed on the applicable Product Order, or at such other address as may be supplied in writing. The date of delivery or the date of mailing will be the date of notice.
- f. Resellers.** Rocket Software's obligations with respect to products or services supplied by Rocket Software and procured by Customer from an authorized Rocket Software reseller are limited to the

terms and conditions in this Agreement and the Documentation included with the Rocket Software supplied products and services. With purchases from a reseller, the applicable pricing and payment terms are as set out in the separate agreement between Customer and the reseller, and any terms in this Agreement related to Rocket Software pricing and payments shall not apply. Rocket Software is not responsible for the acts or omissions of the reseller, or for any other products or services that it supplies to Customer.

- g. Entire Agreement.** This Agreement and the applicable Product Order(s) and ALA(s) represent the entire understanding of the parties with respect to the subject matter of this Agreement and supersede any previous communications or agreements that may exist regarding the same subject matter.
- h. Order of Precedence.** Subject to the Product Orders section, any conflicting terms and conditions will be resolved according to the following order of precedence: the applicable Product Order, the applicable ALA, and this Agreement.
- i. Amendment.** No modification of this Agreement will be binding on the parties unless in writing signed by authorized representatives of Rocket Software and Customer.
- j. Waiver.** No waiver of any right under or in connection with this Agreement will be effective unless in writing, signed by authorized representatives of both parties. No waiver of any past or present right arising from any breach or failure to perform will be deemed to be a waiver of any future right arising under or in connection with this Agreement.
- k. Publicity.** For the term of this Agreement, Rocket Software may use Customer's name in press releases, product brochures, and other marketing materials to indicate they are Rocket's customer. Rocket will use Customer's trademarks, service marks, logos or branding, according to any branding guidelines which Customer makes available to Rocket Software. Licensee hereby agrees to serve as a reference for the Product(s) at times to be mutually agreed upon. Such references may, as mutually agreed upon, include activities such as (i) reference calls with or hosting of a site visit with mutually acceptable prospects; (ii) a published "News Release" announcing the successful partnership with Rocket no later than sixty (60) days from the date of this Agreement; (iii) mention of Licensee or Licensee logo on Rocket Software's website; (iv) the completion of a "Case Study;" or (v) sharing Licensee's "Success Story" which may include speaking at Rocket Software events. Both parties will agree to the content and timing of any announcement focused on their relationship prior to distribution.
- l. Severability.** If any provision in this Agreement or the applicable ALA(s) or Product Order(s) is held invalid or unenforceable, that provision will be construed, limited, modified or, if necessary, severed, to the extent necessary, to eliminate its invalidity or unenforceability, and the other provisions will remain unaffected.