Be User's Guide

Be User's Guide copyright © 1996 Be, Inc. All rights reserved.

Release 1.1d7 of the Be OS copyright © 1990–1996 Be, Inc. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted—in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise—without the prior written permission of Be, Inc.

The contents of this book are furnished for informational use only; they are subject to change without notice and should not be construed as a commitment by Be, Inc. Be has tried to make the information in this book as accurate and reliable as possible, but assumes no liability for errors or omissions.

Be, Inc. will from time to time revise the software described in this book and reserves the right to make such changes without notification. The software is furnished under license and may be used or copied only in accordance with the terms of the license.

Be, the Be logo, BeBox, Be OS, and GeekPort are trademarks of Be, Inc. Metrowerks is a registered trademark and CodeWarrior is a trademark of Metrowerks, Inc. Agfa and the Agfa Rhombus are registered trademarks of Agfa-Gevaert, AG. TrueType is a trademark of Apple Computer. Intellifont is a registered trademark of Miles Inc. Arial, Times New Roman, Monotype, and Monotype Baskerville are trademarks or Monotype Typography Ltd. registered in the U.S. Patent and Trademark Offices. The Type 1 processor resident in the UFST is licensed from Pipeline Associates, Inc. All other trademarks mentioned belong to their respective owners.

Be, Inc.
800 El Camino Real
Suite 300
Menlo Park, CA 94025
http://www.be.com

April, 1996

BUG-001.001-041596

Contents

Introduction 1

Chapter 1: Assembling the BeBox 3
Adding Internal Devices 4
Opening the BeBox 4
Working Inside 4
Adding Memory 6
Inserting ISA and PCI Cards 6
Adding a Display Adapter 7
Adding a Network Adapter 8
Adding an Internal Disk Drive 9
Closing the BeBox 12
Connecting External Devices 13
Connecting a SCSI CD-ROM Drive 15
Finishing Up 15

Chapter 2: Learning BeBox Basics 17
Turning the BeBox On and Off 17
Turning On the BeBox 17
Selecting a Boot Disk 18
Shutting Down the BeBox and Turning It Off 19
Resetting the BeBox 20
Interrupting the BeBox 21
Rebuilding the Database 21
Getting to Know the BeBox Workspace 22
Using the Mouse 23
Using the Keyboard 25

Working with Windows 27
Moving a Window 28
Resizing a Window 29
Scrolling the Items in a Window 29
Zooming a Window 31
Minimizing a Window 32
Selecting Items 33
Changing Views 34
Working in Icon and Mini Icon View 35
Working in List View 36
Closing a Window 37
Working with Menus 37
Navigating Menus by Dragging 38
Navigating Context-Sensitive Menus 38
Navigating Menus by Clicking 39
Navigating Menus from the Keyboard 40
Choosing Menu Items by Shortcut 40
Working with the Browser 41
Understanding the Hierarchical File System 41
Getting Information About an Item 42
Opening Folders 42
Working with Disks 43
Mounting a Disk 44
Unmounting a Disk 44
Renaming Items 44
Creating a Folder 45
Moving and Copying Items 46
Duplicating Items 47
Deleting Items 47
Working with the Dock 48
Adding Items to the Dock 49
Rearranging Items on the Dock 51
Removing Items from the Dock 51
Changing the Dock Display 52
Finding Items 53
Opening a Query 54

Chapter 3: Using Be Applications 55
Application Basics 55
Starting an Application 55
Getting Information About an Application 56
Switching Applications 56
Quitting an Application 56
Creating a New File 57
Saving a File 57
Opening Files 59
Working with Text 61
Selecting Text 61
Typing Text 62
Typing Special Characters 62
Copying and Moving Text 63
Moving Text with the Mouse 63
Deleting Text 64
Undoing 65
Using the Be Applications 65
BeBounce 65
BeLogo 65
CDPlayer 66
Playing CDs 66
Saving Audio Data 68
Clock 70
Edit 70
Changing the Font 70
Automatically Indenting Lines 71
Indenting and Unindenting Lines 71
Changing the Width of Tabs and Indents 72
Commenting and Uncommenting Lines 72
Finding Text 73
Replacing Text 74
Showing Line Numbers 75
Going to a Specific Line by Line Number 75
Creating and Using Markers 76
Deleting Markers 77
Creating a Script File 77

Flight 78
Preparing for Battle 78
Engaging Battle 78
FontChart 80
FontDemo 81
HeapWatch 82
HelloWorld 82
IconWorld 82
ImageViewer 83
Installer 83
Initializing or Formatting a Disk 83
Kaleidoscope 86
Keyboard 86
Magnify 88
Mandelbrot 89
MessageCenter 90
Midi 90
MiniPlayer 92
Orb 93
PlaySound 94
PoorMan 94
Pulse 95
ROMUpdater 95
Terminal 96
Learning About bash and the Programs in the /bin Folder 97
Workspaces 97

Using the Preferences Applications 99
Mouse 99
Network 100
Configuring Network Basics 101
Adding an Ethernet Network Interface 102
Adding a PPP Network Interface 103
Saving Network Settings and Restarting the Network Software 104
Screen 105
Setting the Screen Resolution, Colors, and Refresh Rate 105
Adjusting the Screen Size and Position 106
Setting a Desktop Color 106
Displaying an Image on the Desktop 107
ScrollBar 108
Sound 108
Time 110
Changing the Boot Disk 111
Adding Fonts 112
Installing Additional TrueType Fonts 112
Creating Screen Dumps 113
Capturing a Screen Dump 113
Converting a Screen Dump to Raw Format 113
Opening a Raw ImageViewer File in Photoshop 113
Transferring Files to and from the BeBox 115
Using ftp 115
Using suntar 118
Adding Drivers and Add-Ons 119

Chapter 4: Developing Be Applications 121
Setting CodeWarrior IDE Preferences 122
Target 123
Access Paths (Access Paths) 123
C/C++ Language (Language) 124
PPC Processor (Processor) 124
PPC Linker (Linker) 125
PPC PEF (PEF) 125
PPC Project (Project) 126
Developing Be Applications on a Macintosh 127
Using the CodeWarrior IDE 127
Transferring Your Application to the BeBox 129
Using the Cross-Debugger 129
Setting Up the Nub on the BeBox 129
Setting Up the Cross-Debugger on the Macintosh 130
Debugging Applications 131
Developing Applications on the BeBox 133
About the Development Environment on the BeBox . . 133
Using the BeIDE 134
Using the Tools 135
Preparing the Makefile 135
Building the Application 137
Creating Resources for a Be Application 138
Using IconWorld 139
Creating an Application-Information Resource . 139
Creating Icons 140
Drawing Icons 141
Importing Icons 143
Setting File Types 143
Adding Resources to Applications 144
Getting the Icons to Be Recognized 145
The Debugger 146
Entering the Debugger 146
Debugger Windows 147
Using the Debugger 147
Conventions 148
Instruction Display 149
The Commands 150

Appendix A: The BeBox Ports 153
Internal Ports 154
Internal BeBox Motherboard Ports 155
Front Panel Port (J1) 155
Speaker Port (J2) 155
Logic Analyzer Probe Port (J3) 156
Floppy Disk Drive Connector (J4) 157
SCSI-II Connector (J5) 158
IDE Connector (J6) 159
DRAM SIMM Slots (J7 through J14) 160
PCI Slots (J16,17,18) 160
ISA Slots (J19 through J23) 160
Power Connectors (J15) 161
Internal I/O Board Port 162
CD-ROM Drive Pass-Through Connector 162
External BeBox Ports 163
Parallel Port (J27) 164
Keyboard Connector (J25) 165
External SCSI-II Connector (J26) 166
Headphone Port 167
Microphone Port 167
Line-In Left Port 167
Line-In Right Port 168
Line-Out Left Port 168
Line-Out Right Port 168
MIDI-In Ports 169
MIDI-Out Ports 169
Serial Ports 170
GeekPort 171
Joystick Ports 172
IR Ports 172
Mouse Port 173

Introduction

This book is a preliminary draft of the *Be User's Guide*, written to accompany the preliminary version of the Be™ operating system: Release 1.1d7 of the Be OS.™ This book shows you how to set up a BeBox,™ it teaches you the basics of the Be OS, the Be user interface, and the Be applications, and it guides you through the process of developing Be applications using the Metrowerks® CodeWarrior™ 8 Integrated Development Environment on a Macintosh computer as well as using the preliminary development environment on the BeBox.

- Chapter 1, "Assembling the BeBox," shows you how to assemble the BeBox and connect a variety of internal and external devices to it.
- Chapter 2, "Learning BeBox Basics," teaches you the basics of working with the BeBox, including turning it on and off, using the keyboard and mouse, and managing your files.
- Chapter 3, "Using the Be Applications," first introduces you to tasks common to most Be applications, such as starting them up, creating and saving files, and switching among running applications. Then it provides brief instructions for using most of the applications that come with the Be software. Finally, it teaches you some advanced procedures, such as installing TrueType™ fonts.
- Chapter 4, "Developing Be Applications," describes the process of developing Be applications.
- Appendix A, "The BeBox Ports," provides diagrams, pin-out tables, and other technical information about the many BeBox ports, from the industry-standard disk drive connectors to the unique Be GeekPort.™

Future editions of this will include a complete glossary and an index. It will also be more complete and more thoroughly edited.

In addition to this book, you'll also want to refer to:

- *Installing or Upgrading to Release 1.1d7 of the Be Software*, the booklet that accompanies this user's guide, which shows you how to install release 1.1d7 of the Be OS on a BeBox and the development environment on a Macintosh.
- *The Be Book*, the reference to the Be API, now available in HTML format in the **/documentation/The Be Book** folder, and in PostScript and a variety of archive formats on the *Be OS* CD-ROM.
- The Be world-wide web site: **http://www.be.com**, where you'll find advice on and support for developing Be applications, additions and corrections to this book and *The Be Book*, and a great deal of other information about Be, Inc., and the Be hardware and software.

1 Assembling the BeBox

The BeBox comes in a number of configurations, but at a minimum, you need:

- A 101-key PC or Microsoft keyboard
- A serial or PS/2 mouse
- A high-quality “multisync” display
- A PC floppy disk drive
- An IDE or SCSI hard disk drive
- A display adapter card (for a list of compatible cards, see “Adding a Display Adapter” on page 7)
- 8 MB (megabytes) of RAM (Random Access Memory)
- For software installation, a SCSI CD-ROM drive
- An NE2000, 3Com 3C503, or compatible network adapter (if you want to connect the BeBox to an Ethernet network)

This chapter shows you how to assemble a BeBox with these and other devices. For details on the connectors and ports both inside the BeBox and on the back panel, see Appendix A, “The BeBox Ports.”

Adding Internal Devices

To add memory, a graphics adapter, a network adapter, or internal disk drives, you need to open the BeBox.

Opening the BeBox

Opening the BeBox is easy; you don't even need tools.

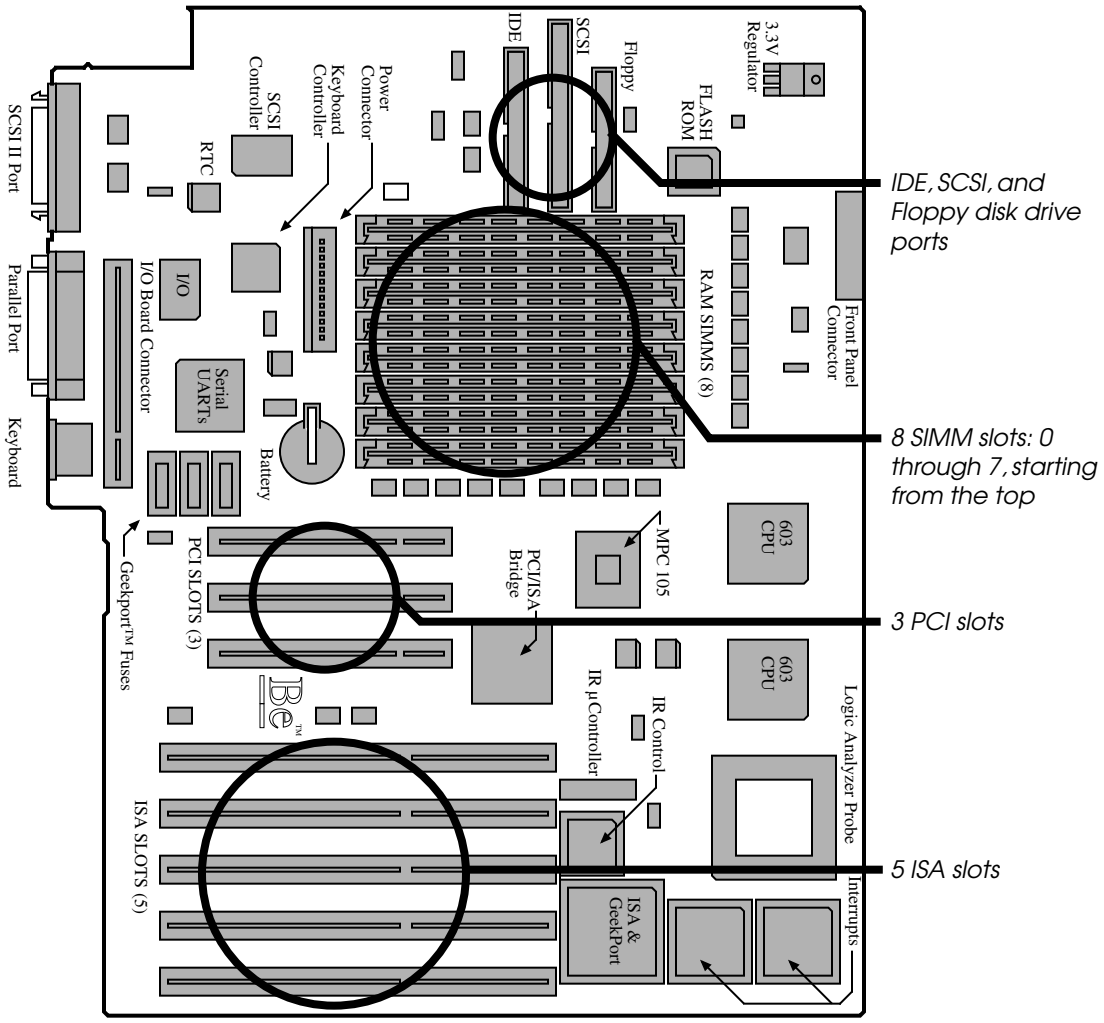
1. Make sure the BeBox isn't plugged in and you don't have any devices connected to the external BeBox ports.
2. Unscrew the two brass thumbscrews on either side of the back panel until they're loose (they're "captive" screws, so they just hang loosely in place when they're unscrewed—they don't come out all the way).
3. Move the outer cover forward about two inches and then lift it up and off the internal chassis.

Working Inside

Once the cover is off, it's easiest to add devices to the inside of the BeBox if you set it flat on its side, with the large opening facing up.

If the back panel is to your left, you can see the steel housing of the power supply on the upper left, next to the back panel. The drive bays are to the right of the power supply: One bay hangs from the top of the BeBox, the other is connected to the front. The *motherboard* is the large circuit board on the bottom, with the I/O board plugged into it perpendicularly, on the left side.

The motherboard has slots for 3 PCI cards, 5 ISA cards, IDE drives, SCSI drives, floppy disk drives, and 8 SIMMs (Single In-line Memory Modules). Take a moment to find these connectors on the motherboard, using the following diagram:



The BeBox Motherboard

Adding Memory

You can add up to 256 MB of RAM to the BeBox by inserting identical pairs of 2, 4, 8, 16, 32, 64, or 128 MB, 60-nanosecond or faster, 72-pin SIMMs in memory slots 0 and 1, 2 and 3, 4 and 5, or 6 and 7. Add SIMMs to the slots in order—start with slots 0 and 1 and don't leave a gap between the SIMM slots. The BeBox can use parity SIMMs, but it doesn't yet support parity-checking. Also keep in mind that while each pair of SIMMs must be made up of identical twins, no two sets of twins must be alike: For example, you could insert 32 MB SIMMs in slots 0 and 1 and 2 MB SIMMs in slots 2 and 3.

To add a SIMM module, start with the top empty SIMM slot (the one with the lowest number). With the notch on the SIMM's connector edge to the left (nearer the back panel), slide the SIMM into the slot at a 45-degree vertical angle—that is, with the connector edge of the SIMM closer to the top of the BeBox than the edge nearer to you. When the SIMM is all the way into the slot, lift up the edge nearer you so the SIMM snaps into place (it's held in the slot by a pair of metal clips).

To remove a SIMM module, start with the bottom SIMM (the one in the slot with the highest number). Release the SIMM by pushing the two metal clips away from the SIMM while gently pushing down on the edge of the SIMM. When it's free from the clips, lift it out.

Inserting ISA and PCI Cards

You can add up to 5 ISA and 3 PCI cards to the BeBox. The ISA and PCI slots are on the motherboard, as shown in the diagram on page 5. You can insert cards in any slot: Order and slot number make no difference

1. Unscrew the metal plate that covers an unused slot's opening through the back panel. Save the screw—you'll need it in a moment.

   Store the metal plate someplace safe, in case you ever remove the card from the slot. The plate helps keep dust out of the BeBox and reduces electromagnetic interference.

2. Insert the card in the slot.

   It's best to start by lining up the gold "fingers" on the connecting edge of the card with the slot, and the metal connector plate with the opening through the back

panel. Make sure the card is started into the slot correctly. Then firmly push the card all the way into the slot.

3 Replace the screw you removed in step 1.

The screw keeps the card from coming loose, particularly when you're connecting a cable to the card.

Adding a Display Adapter

Currently, the BeBox works with a number of PCI display adapters based on graphics chips manufactured by Cirrus Logic and S3. You can insert a display adapter into any one of the three PCI slots.

You can use any display adapter based on the Cirrus 5430, 5434, 5436, and 5440 family of chips. For example, the BeBox has been tested with the Orchid Kelvin 32 (based on the 5430), the Orchid Kelvin 64 (based on the 5434), the Diamond SpeedStar Pro SE (based on the 5430), and the STB Nitro 64 (based on the 5434). Cards that are based on the Cirrus 5434 or 5436 and have 2 MB of RAM can display the Be OS with 8 or 32 bits of color data per pixel. Cards that are based on the other two Cirrus chips or that only have 1 MB or RAM can display the Be OS with 8 bits of color data per pixel.

You can also use a display adapter based on the S3 Vision 864 (combined with the ATT21C498 RAMDAC chip) and S3 Vision 964 (combined with the TVP3025 RAMDAC chip). For example, the BeBox has been tested with the #9 GXE64 (based on the Vision 864) and the #9 GXE64 Pro (based on the Vision 964). With 2 MB of RAM, these cards display the Be OS with 8 or 32 bits of color data per pixel. With 1 MB of RAM, the #9 GXE can display the Be OS with 8 bits of color data per pixel.

You set the number of colors per pixel, the resolution (the number of pixels on the screen), and other settings for how the Be OS is displayed with the Screen preferences application. See "Screen" on page 105.

With release 1.1d7 of the Be OS, third-parties can start writing drivers, so you can expect to see support for many more display adapters in the near future. For more information, see *The Be Book*, the "ReadMe" file that accompanies the S3 sample project (**/develop/projects/s3/ReadMe**), and the Be web site (**http://www.be.com**). The "ReadMe" file also provides additional information about features supported by cards based on S3 chips.

Adding a Network Adapter

You can add up to five Ethernet network adapters to the BeBox, inserting them in one or more of the five BeBox ISA slots. With a network connection, you can use the **ftp** utility to transfer files between the BeBox and a Macintosh, or any other machine on the network that can run **ftp**. If the Ethernet network has a connection to the Internet, you can use the Orb application to surf the world-wide web. For more information on **ftp** and Orb, see Chapter 3, "Using Be Applications."

You use the Network preferences application to configure the BeBox to use the network adapters you add. You can also use Network to configure the BeBox to communicate with another computer using a modem or a direct serial connection. See "Network" on page 100.

Two kinds of Ethernet network adapters work with the BeBox: The NE2000 and 3Com 3C503. You can also use any of the many models of network adapters that are compatible with the NE2000.

Most network adapter cards come set to IRQ 3 and I/O port 300. The BeBox serial ports use IRQ 3, so you must change the network adapter's IRQ before you install it. The default value in the Be networking software is IRQ 5 and port 300, but you can use IRQ 5, 9, 10, 11, 13 or 15 and I/O port 220, 240, 280, 2A0, 2C0, 300, 320, 340, or 3C0. If you add more than one network adapter, you must set each to a unique IRQ and I/O port.

If you're adding a 3Com 3C503 card, make sure it's set to use memory address C8000. Because memory addresses must be unique and you can't change memory address settings using the Network application, it's difficult to install more than one 3Com card in the BeBox. (If this poses a serious problem for you, contact Be Developer Support by e-mail at **devsupport@be.com**.)

For information on changing network adapter settings, see the owner's guide that came with the adapter.

Adding an Internal Disk Drive

The BeBox has two internal *drive bays*—metal frames where you can mount disk drives. One drive bay hangs from the top of the BeBox's internal chassis, the other bay is connected to the front. You can mount up to two 3.5-inch drives in the top bay. You can mount a floppy disk drive in the bottom section of the front bay, plus either two half-height or one full-height 5.25-inch drives in the front bay. If you have the appropriate mounting brackets, you can also mount 3.5-inch drives in the 5.25-inch area of the front bay.

You add an internal disk drive by removing a bay, mounting the drive in the bay, replacing the bay, and then connecting the drive to the motherboard.

Removing a Drive Bay

1. Disconnect any cables connected to drives already mounted in the bay you want to remove.
2. Remove the screws that hold the bay in place.

   The top bay is held in place by two screws through the top of the BeBox's internal chassis.

   The front bay is held in place by 3 screws through the front of the BeBox's internal chassis.

   Support the bay with your hand as you remove the last screw, so it doesn't fall.
3. Remove the bay.

   Once the screws are removed, the top bay is held only by metal flanges hanging from the top of the BeBox. You can simply tilt the bay down a bit and slide it out.

   The front bay slides straight out of the front of the BeBox.

Mounting a Drive in the Top Drive Bay

Remove the drive bay from the chassis, as described above. Then slide the drive into either the left or right side of the bay, with the circuit-board face of the drive facing down. Align the screw holes on the circuit-board face of the drive with the mounting

holes on the bottom of the bay. Then screw the drive to the bay, using the screws that came with the drive.

Warning: Some 3.5-inch drives have components that stick so high above the circuit-board face of the drive that they will form an electrical connection that will likely destroy the drive if you mount it without insulating these components from the drive bay. If you're not sure how to insulate components, get advice from someone with experience assembling computers.

Mounting a Drive in the Front Drive Bay

Remove the drive bay from the chassis, as described above. Slide the drive into the bay. Align the screw holes on the side of the drive with one of the sets of mounting holes on the sides of the bay. Then screw the drive to the bay, using the screws that came with the drive. If you installed a floppy disk drive, CD-ROM drive, or another drive that uses removable media, remove the corresponding cover plate from the *bezel*, the plastic cover on the front of the outer cover of the BeBox. If the panel is attached to the cover with screws, remove the screws first. Then simply push the panel in until it's free.

Replacing a Drive Bay

Reverse the steps you followed to remove the drive bay. Be sure to replace the screws that secure the bay to the internal chassis.

Connecting a Drive to the BeBox Motherboard and Power Supply

After you mount a drive in a drive bay and replace the bay in the chassis, you can use the cables that come with the BeBox to connect the drive to the BeBox motherboard and power supply.

1. Use one of the provided ribbon cables to connect the drive to the appropriate drive connector on the motherboard.

   Ribbon cables must be connected with the correct orientation: Make sure pin 1 in the connectors on the motherboard and on the drive are aligned with pin 1 on the

cable. On most cables, the edge of the ribbon carrying the signal for pin 1 is marked with a line or stripes.

If you're connecting a SCSI device, use the connectors at both ends of the SCSI ribbon cable first, and only use the connectors in the middle of the cable to attach additional devices. Make sure the device at the end of the ribbon cable is terminated. Also make sure that each SCSI device has a unique SCSI ID, from ID 0 through 6 (ID 7 is reserved for the SCSI controller on the motherboard). For instructions on terminating a SCSI device and setting its SCSI ID, see its owner's guide.

2 Plug one of the plastic power supply connectors into the drive.

Connecting a CD-ROM Drive to the I/O Board

You can use a "pass-though cable" to connect an internal CD-ROM drive to the BeBox I/O board. With a pass-through cable, you can listen to audio CDs on the BeBox's internal speaker and through the line-out and headphone ports. You can also save sections of CD audio tracks as disk files. (For more information on playing CDs, see "CDPlayer" on page 66.)

Pass-through cables are available at larger electronics supply stores. (One source is TTS Multimedia Systems, 2045 San Elijo Ave., Cardiff, CA 92007, (800) 887-4968.) Make sure you buy a pass-through cable that has a connector for your brand of CD-ROM drive at one end, and a connector compatible with a MediaVision ProAudio Spectrum 16 card at the other (this is the end you connect to the I/O board).

Connect the pass-through cable's ProAudio Spectrum 16 connector to the connector on the I/O board: The connector is on the corner of the card farthest from the power supply and the motherboard. Then connect the other end of the pass-through cable to the CD-ROM drive.

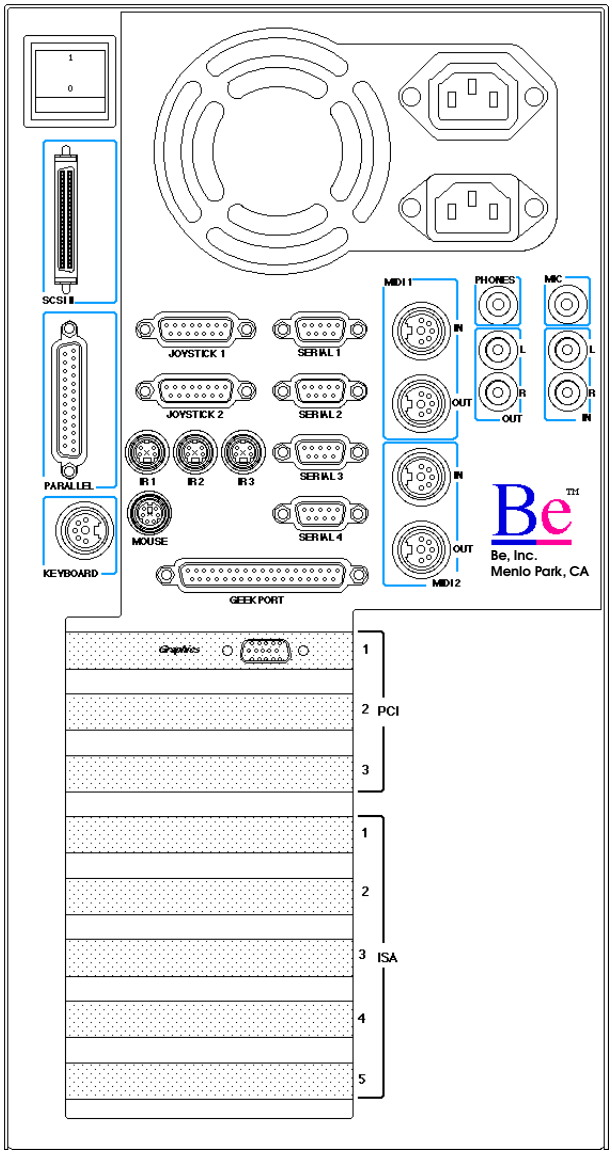
Closing the BeBox

When you're done adding internal devices, it's best to put the cover back on. This protects you and the internal components from each other, and reduces electromagnetic interference.

1. With the outer cover positioned over the internal chassis, but about two inches farther forward, lower the outer cover into place.
2. Make sure the floppy disk drive and the corresponding opening in the bezel are aligned. Then slide the cover back until it's closed.
3. Tighten the brass thumbscrews in the back panel.

Connecting External Devices

A plastic cover labels many of the ports on the back of the BeBox, but you may need to refer to the following diagram to identify the others:



1. Make sure the switch between the two power sockets on the back of the BeBox is set to the correct voltage.

   Warning: The voltage switch has two settings: 115 volts and 230 volts. The switch is initially set to 115 volts, which is the correct setting in North America. Outside of North America, make sure the switch matches the voltage in your area. If the voltage switch is set incorrectly when you turn on the BeBox, you risk destroying the BeBox.

2. Plug one end of the power cable into the back panel of the BeBox and the other end into a grounded, three-prong electrical outlet.

   Warning: You must electrically ground the BeBox using the power cable provided by Be. The power cable includes a three-prong plug—a plug with a grounding pin—that will only fit in a three-prong grounded outlet. The grounding pin is for the safety of the BeBox and—more important—for your safety. If you don't have a grounded outlet, hire a licensed electrician to install one. Do not circumvent the three prong plug by finding a way to plug it into an outlet that isn't grounded!

3. Connect the keyboard to the keyboard port on the back panel.

4. Connect the mouse to the appropriate port on the back panel.

   You can connect a serial mouse to serial port 2, 3, or 4.

   You connect a PS/2 mouse to the Mouse port.

   When you turn on the BeBox, it finds the mouse wherever you connected it.

5. Plug the display's power cord into a grounded outlet or—if you have the appropriate plug—into the socket on the back of the BeBox. Then connect the display to the port on the display adapter card in a PCI slot.

6. If you installed a network adapter card, use an Ethernet cable to connect it to your network.

   If you don't maintain the network at your location, be sure to check with your system administrator before connect the BeBox to the network. (Configuring the BeBox to communicate over the network is described in "Network" on page 100.)

Connecting a SCSI CD-ROM Drive

To install the Be operating system from the *Be Operating System* CD-ROM, you need to connect a SCSI CD-ROM drive to the BeBox's external or internal SCSI port. (IDE CD-ROM drives are not currently supported.) The external SCSI port is just below the On/Off switch on the back panel. Follow the directions that came with your CD-ROM drive to terminate the drive properly (as a rule, terminate the drive if it's the only SCSI device connected to the internal or external SCSI port on the BeBox, or if it's at the end of a chain of SCSI devices). You can set the CD-ROM drive to any SCSI ID, except SCSI ID 7, but remember that if you connect more than one SCSI device to the BeBox, each device must have a unique SCSI ID.

Finishing Up

Once you've assembled the BeBox, you can install the Be operating system (the Be OS) on its hard disk. Installing Be software is described in *Installing or Upgrading to Release 1.1d7 of the Be Software*, the booklet that accompanies this user's guide.

If the BeBox came with the Be OS installed, you can simply turn on the power switch and start using it. The next chapter starts with instructions for turning the BeBox on and off safely. Then it shows you the basics of using the Be OS.

2 Learning BeBox Basics

This chapter shows you the basics of working with the BeBox: Turning it on and off, working with the mouse and keyboard, and mastering the fundamentals of the user interface. The next chapter picks up where this one leaves off, showing you the basic techniques you use with Be applications, and then introducing the applications that come with the Be OS.

Turning the BeBox On and Off

Once you've set up the BeBox and installed the Be OS, you can turn on the BeBox and start using it.

Turning On the BeBox

1. Make sure there's no floppy disk or CD-ROM that contains Be system software inserted in a drive connected to the BeBox.

   When you turn on the BeBox, it tries to find system software on removable disks first. Then it looks for system software on a hard disk. For more information, see "Selecting a Boot Disk" on page 18.

2. Turn on any peripheral devices, such as the display or external disk drives.

3 Turn on the BeBox.

   The BeBox On/Off switch is a toggle-style switch on the top of the back panel. Pushing the toggle up turns on the BeBox; pushing it down turns it off.

   When you turn on the BeBox, the Be logo appears for a moment. Then the BeBox loads system software from the startup disk (this is often called "booting"), and performs some housekeeping tasks, which it may inform you about. Then the BeBox starts the Browser application. The Browser is where you manage the work you do on the BeBox: Arranging and opening files, starting applications, and so on (the Browser is described in detail in "Working with the Browser" on page 41).

When the BeBox is on, two columns of LEDs (Light-Emitting Diodes) become active on the front of the BeBox. The green LEDs at the bottoms of the columns act as power indicators: They glow whenever the BeBox is turned on. The upper LEDs show the relative load on the BeBox's two PowerPC processors: The more LED's are lit, the harder the PowerPCs are working. (The Pulse application provides a similar display, as described in "Pulse" on page 95.) The red LED at the bottom of the right column is a disk activity light: Whenever the Be OS uses an IDE or SCSI disk, the red LED glows.

Selecting a Boot Disk

When you turn on the BeBox, it looks in a fixed order for a disk to start up (or "boot") from. First, it looks for system software on a floppy disk. If it doesn't find a floppy disk with system software, it then looks for any other kind of removable disk, such as a CD-ROM. If it doesn't find a removable disk with system software, it checks the "boot preference"—a setting stored in a chip on the motherboard—to determine what disk you prefer to use as the boot disk. The BeBox then tries to boot from the disk specified in the boot preference. If for some reason the preferred disk isn't available or doesn't contain system software, the BeBox tries to boot from any other available disk with system software on it.

The boot preference is initially set to the master IDE disk. If you install the Be OS on another disk, the Installer asks if you want to change the boot preference to that disk. You can change the boot preference at any time with the **boot** command in a Terminal window. For more information, see "Changing the Boot Disk" on page 111.

Regardless of the boot preference and what disks are inserted or attached, you can select the disk you want to boot from by pressing any key right after you turn on the BeBox—as soon as you see the Be logo start to appear. The boot selection screen opens, complete with instructions for how to use it:

```
Select boot options and press Enter.
Or press Escape to use current boot preferences.

Use the up and down arrow keys to select options.
Use the Tab key to switch sets of options.

Select a boot disk:
   Rescan for bootable disks
>> /dev/ide_disk_master (Be disk 'My Boot Disk')
   /dev/scsi_disk_000 (Be disk 'Backup Boot Disk')

After booting from selected disk:
>> Mount other disks
   Don't mount other disks
```

Each bootable disk is listed by its device name as well as by the name it has in the Browser. Disks that don't have Be system software on them aren't included in the list.

For more information about SCSI IDs and IDE disks, see "Adding an Internal Disk Drive" on page 9 and "Connecting a SCSI CD-ROM Drive" on page 15.

Shutting Down the BeBox and Turning It Off

Before you turn off the BeBox, you must first shut it down. Shutting down ensures that any open files—including system files—are saved to disk correctly. If you turn off the BeBox without first shutting it down, you risk losing information.

1. Save any open files and quit any running applications.

   Saving files and quitting applications are described in "Application Basics" on page 55.

2. Choose Shut Down from the Browser's main menu.

   The main menu is just to the right of the application list, at the top-left corner of the screen. For more information about menus, see "Working with Menus" on page 37.

   If you just want to restart the BeBox instead of turning it off, choose Restart from the main menu instead.

3. When a panel tells you it's safe to turn off the BeBox, turn it off. You can also click the Restart button in the panel (or press the Enter key) to restart the BeBox right away, without turning it off.

4. After you've shut down and turned off the BeBox, turn off any peripheral devices that have their own power switches, such as the display or external disk drives.

Resetting the BeBox

If the BeBox crashes, or if the Shut Down and Restart commands don't work, you can restart the BeBox (without turning it off and on) by resetting it.

There are two ways to reset the BeBox. The nostalgic way is to press "Control-Alt-Delete." In other words, to press the following three keys at the same time: The key labeled "Ctrl" at the bottom left of the keyboard (not the one next to the arrow keys); either one of the two keys labeled "Alt" on either side of the space bar; and either the key labeled "Del" on the numeric keypad or the key labeled "Delete" above the arrow keys. (For more information about the keyboard, see "Using the Keyboard" on page 25.) The other way to reset the BeBox is to press the Reset and Interrupt buttons at the same time. The Reset and Interrupt buttons are built into the BeBox bezel, on the horizontal surface below the floppy disk drive, just above each LED array on the front of the BeBox.

Caution: If you restart the BeBox without choosing Shut Down first, you risk losing data. In particular, you'll lose any changes you've made to open files since you last saved them. You also risk corrupting the Be database. If you corrupt the Be database, you may need to rebuild it. For details, see "Rebuilding the Database" on page 21.

Interrupting the BeBox

Caution: In the prerelease version of the Be OS, the interaction between the Interrupt button and the Debugger is broken, so you shouldn't use the Interrupt button.

If you're using the Be Debugger, you can interrupt the BeBox and open a debugger window. (Using the Debugger is described in "The Debugger" on page 146.)

To interrupt the BeBox, press up on the Interrupt button. The Interrupt button is on the horizontal surface underneath the right side of the floppy disk drive, just above the right LED array.

If you press the Interrupt button accidentally, type **g** and press Enter to tell the BeBox to continue doing what it was doing when you interrupted it. Then close the Debugger window.

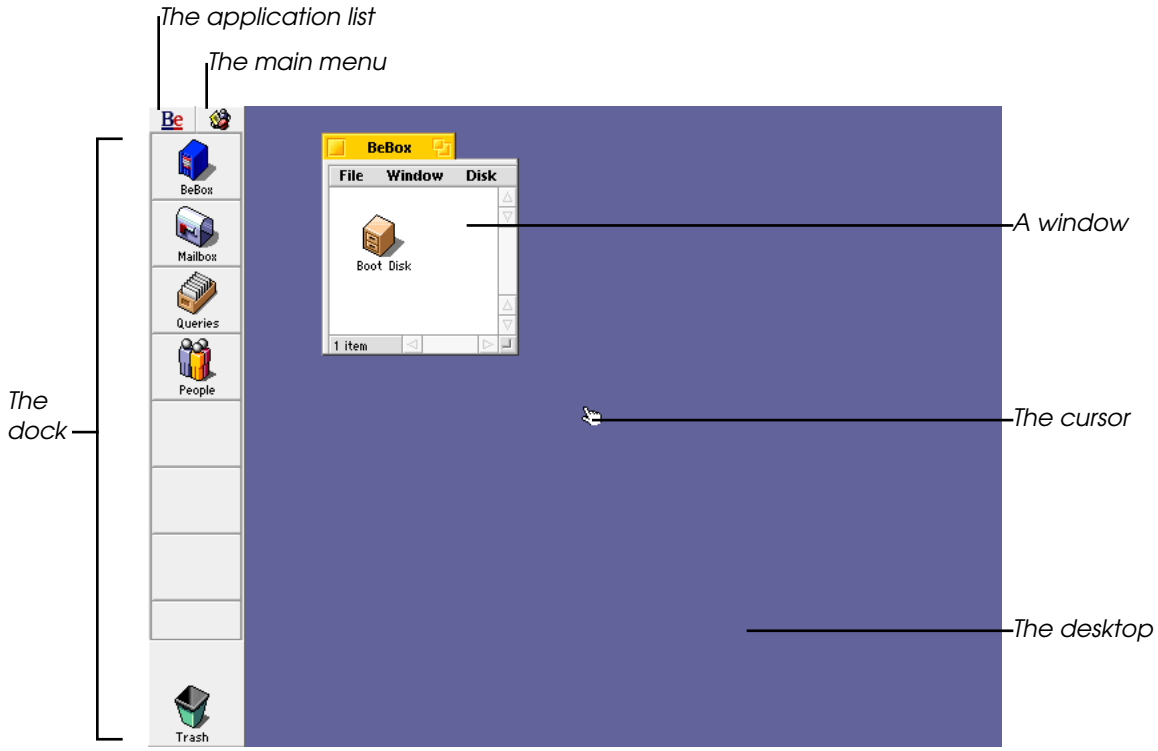
Rebuilding the Database

If the BeBox boots but the Browser never starts up, you may have a corrupted Be database. You can rebuild the database by holding down the left Control and Shift keys as soon as the Be logo disappears, shortly after you turn on the BeBox. Release the keys when the Browser starts up.

Caution: Rebuild the Be database only as a last resort. Rebuilding the database erases all information you've added to it, whether through working with the Be user interface or through your own programs. You lose your window layouts, your arrangement of icons on the dock, your database of audio CDs, and all other information you've stored in the Be database.

Getting to Know the BeBox Workspace

When you turn on the BeBox, the Browser starts up and the dock appears along the left side of the Be desktop. The desktop is the area—initially a solid blue—where windows open and you do your work.



Take a moment to identify the following features of the Be workspace:

- The *cursor* is the hand-shaped icon that moves when you move the mouse. Working with the mouse and using the cursor are described in "Using the Mouse" on page 23.
- The *application list* lists all running applications. You use it to switch among applications, as described in "Switching Applications" on page 56.
- The *main menu* contains menu items for the active application.

- *Windows* are where you view and work with files, text, pictures, and other information. See “Working with Windows” on page 27.
- The BeBox *icon,* at the top of the dock, opens a window that lists mounted disks. See “Working with Disks” on page 43.
- The Mailbox icon opens a window with demonstration messages. You fill the Mailbox with messages by double-clicking the MessageCenter application, as described in “MessageCenter” on page 90.
- The Queries icon opens a window with all the queries you’ve created, as described in “Opening a Query” on page 54.
- The People icon opens a window that lists “people” entities in the Be database. Initially, there’s only one person in the database, but you can add some by choosing Person from the Create submenu in the Browser’s main menu.

Though most of this user’s guide concentrates on showing you how to work in this initial workspace, you can create up to 32 workspaces and switch among them using the Workspaces application. For details, see “Workspaces” on page 97.

Using the Mouse

You use the mouse to control the cursor, to select items, manipulate the controls in windows and panels, choose menu items, and to perform many other tasks. When you move the mouse, a small icon called the cursor moves across the screen. The cursor changes shape depending on what you’re doing: A hand for selecting items and choosing from menus, an I-beam for selecting text, and so on.

You use the mouse buttons to manipulate the items on screen that are underneath the cursor. The BeBox can take advantage of one-, two-, and three-button mice. On a three-button mouse, the left mouse button is initially the *primary mouse button*, the button you click to perform the most common tasks; the right button is the *secondary mouse button*; and the middle button is the *tertiary mouse button*. You can change the role of the mouse buttons with the Mouse preferences application. For example, if you’re left-handed, you can make the right mouse button the primary mouse button. For details, see “Mouse” on page 99.

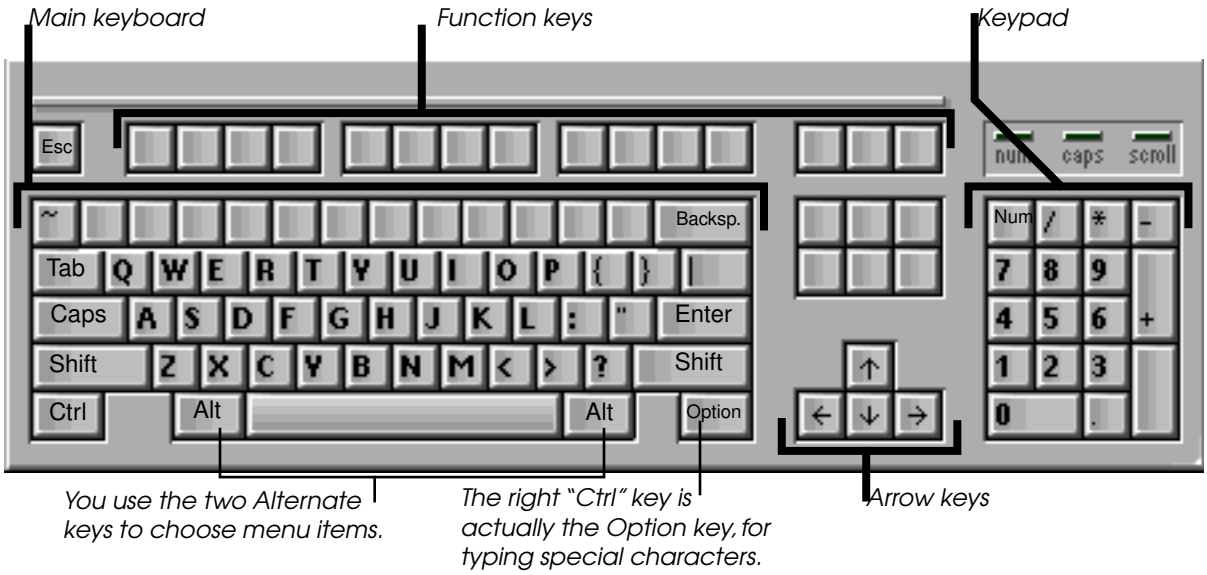
There are a few basic techniques (and terms that describe them) for using the mouse you should learn before continuing with this book:

Move	Move the mouse without pressing a mouse button
Click	Press and quickly release a mouse button without moving the mouse
Press	Press and hold down a mouse button without moving the mouse
Drag	Press and hold down a mouse button while you move the mouse
Double-click	Press and quickly release a mouse button twice in quick succession
Triple-click	Like double-clicking, by thrice
Double-click and drag	Like double-clicking, but hold down a mouse button instead of releasing it the second time, then move the mouse

You can adjust how the mouse interprets multiple clicks and how rapidly the cursor moves across the screen when you move the mouse (see “Mouse” on page 99).

Using the Keyboard

In addition to using the keyboard to type text, you also use it to modify the action of the mouse, to select items, to choose menu items, and to perform many other tasks. The keyboard has two main parts: The main keyboard and the keypad. Along the top of the keyboard are the Escape key and function keys. In between the main keyboard and the keypad are more function keys, plus the arrow keys.



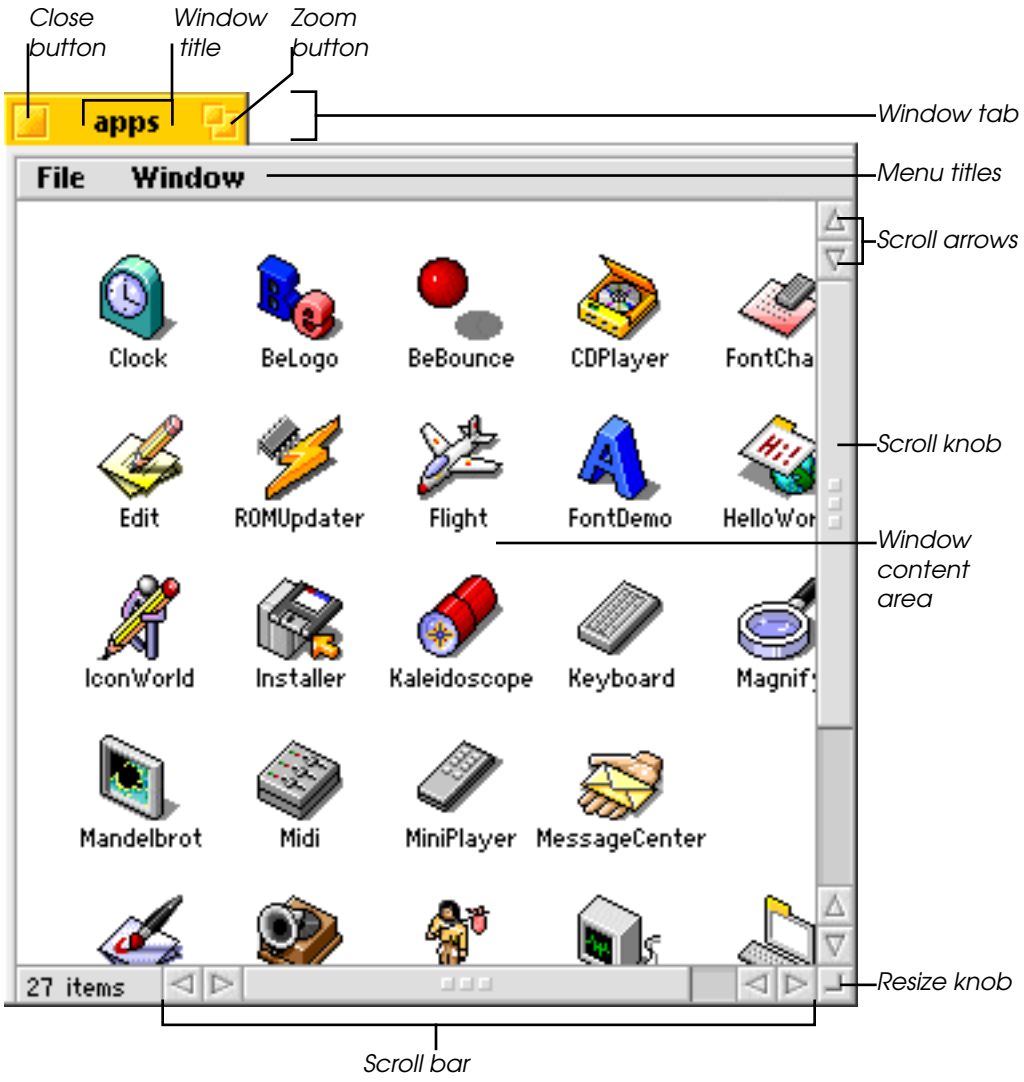
Most of the keyboard keys should be familiar to you from typing class. However, the BeBox uses some of the keys in a way you might not guess:

Key Label	Key Name	Function
Alt	Command	Used in conjunction with other keys as a shortcut to choosing menu items.
Ctrl (on left)	Control	Used to type special characters, particularly when you're working in the Terminal application.
Ctrl (on right)	Option	Used to type special characters, such as symbols and accented characters.
Enter (or arrow)	Enter	Used to end a paragraph when you're typing. Also used as a shortcut for clicking on-screen buttons, opening files, and other common tasks.
(arrows)	arrow keys	Used to move the cursor in text, to select items, and so on.

Using the keyboard is described throughout this book. In particular, see "Typing Special Characters" on page 62. For information about how the BeBox maps the characters you see on screen to the layout of the keys on the keyboard, see "Keyboard" on page 86 and "FontChart" on page 80.

Working with Windows

The Be OS displays information in windows, areas on the screen with some of all of the following features:



Most windows have a tab at the top, with the title of the window and one or more buttons. In the Browser, the title usually tells you the disk the folder is on and the name of the folder whose contents are shown in the window. (These subjects are described in detail later in this chapter. See "Working with the Browser" on page 41.)

You can have one or many windows open at a time, but only the *active window* responds to your actions. For example, typing on the keyboard or using the keyboard to choose items from a menu only affects the active window. Even when windows aren't overlapping, you can tell which is the active window by its tab: The active window has a yellow tab, all other open windows have gray tabs.

You make a window the active window by clicking anywhere in it. When you start up a new application or switch applications, the frontmost window in the new application becomes the active window.

Moving a Window

You move a window by dragging it by its border. The border includes the tab and the gray area on the edges of windows where there's no scroll bar. For most Browser windows, this means you can drag the window by its left and top borders, including the tab. You can drag other windows, panels, and palettes by any border.

When you drag a window by its tab, take care not to click the close or zoom button accidentally.

Resizing a Window

You can change the size and dimensions of a window by dragging its resize knob. The resize knob can be a box-shaped area or just a special part of the bottom-right corner of the window's border—depending on whether the window has scroll bars.

Drag either kind of resize knob to resize a window.

You can also click a window's zoom button (the icon the window's title bar) to make it as big as need be to show all that it contains—or at least as big as the screen. Clicking the zoom button again restores the window to its previous size.

In the Browser, choosing the Resize to Fit command from a window's Window menu has the same effect as clicking its zoom button.

Scrolling the Items in a Window

Sometimes a window contains more icons, more text, or a bigger picture than can fit in the window. You can move the contents of the window within the window to see the items or parts of the document that aren't visible. This is called *scrolling*.

When a window contains more than you can see in it, its scroll bars become active to show that you can scroll to see more. Initially, BeBox windows have *proportional scroll bars*, so called because their scroll knobs change size to give you a rough idea of what proportion of the contents of a window is visible: When you can see almost everything the window contains, the scroll knob is almost as long as the scroll bar; but when you can only see a tiny part of what the window contains, the scroll knob becomes small in relation to the whole scroll bar. You can turn off proportional scroll bars, change the look

of the scroll knob, and set some other options for how scroll bars work with the ScrollBar preferences application, as described in “ScrollBar” on page 108.

The simplest way to scroll is to click or press a scroll arrow. But there are lots of other ways to scroll:

Do This	To
Click a scroll arrow	move the contents of the window a little
Press a scroll arrow	keep moving the contents of a window a slowly
Drag a scroll knob	scroll the contents of the window quickly
Click in a scroll bar	scroll all the way in that direction

When you're working in the Browser, you can also:

Do This	To
Press Page Up	scroll up a windowful
Press Page Down	scroll down a windowful
Press Home	scroll all the way to the top
Press End	scroll all the way to the bottom

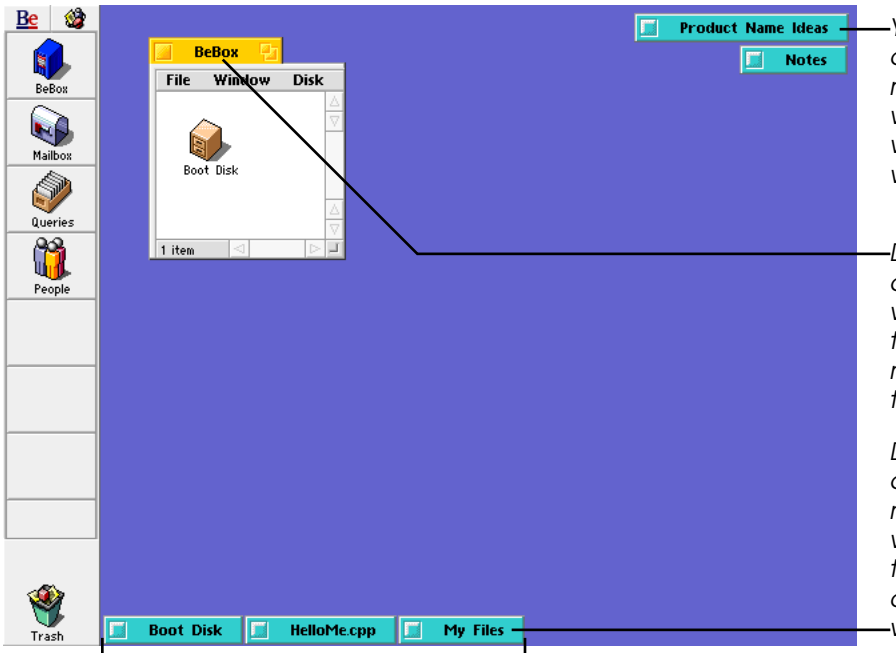
The content of a window also scrolls when you select an item that's not in view, for example, when you use the arrow keys to select items in a Browser window.

Zooming a Window

You can quickly increase the size of a window by clicking its zoom button. In the Browser, clicking the zoom button is the same as choosing the Resize to Fit command from a window's Window menu: The window becomes as large as necessary to display everything in it. Click the zoom button again to return the window to it's previous size and location. Some applications use the zoom button slightly differently. For example, when you click the zoom button in the Workspaces application's window, it moves the window to the lower-right corner of the screen.

Minimizing a Window

If you begin to have more windows open than is convenient, you can minimize one or more windows. Minimizing a window hides all of the window except its tab. To minimize a window, double-click it's tab (but be careful not to double-click the close or zoom button). Tabs of minimized windows are initially laid in rows across the bottom of the desktop. If you prefer a different organization of minimized windows, you can drag tabs to wherever you want on the desktop.



To restore a minimized window to its original size and location, double-click its tab again. Other ways to open minimized windows include:

Type of Window	Way to Open It
Folder	Double-click its icon in the dock or in the window that contains it.
Document	Same as for folder windows. Also, open the document with the Open command. If an application has only minimized windows, switching to that application by double-clicking its icon or choosing it in the application list opens one of its windows.

You can click the close button in the tab of a minimized window to close the window. If the window contains changes you haven't saved, you're asked if you want to save the changes before closing the window.

Selecting Items

To work with an item in a window, you first select it. Selected items usually change color to show that they're selected. For example, when you click an icon in a Browser window, it becomes darker (or *highlighted*).

unselected items **selected items**

Besides clicking, there are lots of techniques and shortcuts for selecting items. The following table summarizes ways you can select items in Browser windows, query windows, and some panels, such as the Open and Save panels.

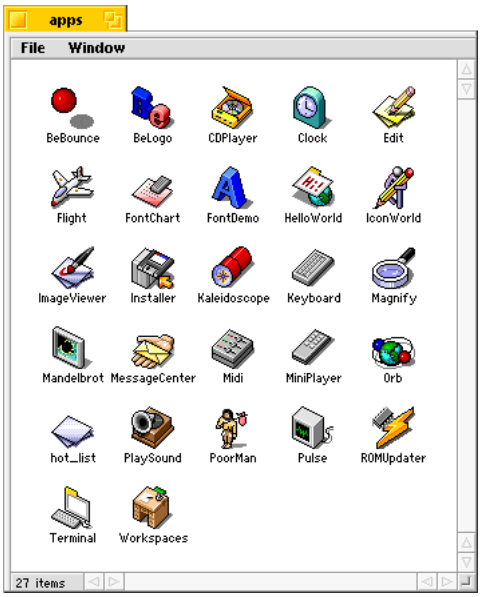
Do This	To
Click an item	Select it
Drag from an empty place in a window across one or more items	Select the items you drag across
Type one or more letters	Select the item that starts with that letter or letters
Press an arrow key	Select the next item, in the direction of the arrow
Press the Tab key	Select the next item, in alphabetical order
Hold down the Shift key while you press the Tab key	Select the next item, in reverse alphabetical order
Hold down the Shift key while you click, drag, or type	Select or deselect additional items
Choose Select All from the File menu	Select all the items in the window

Changing Views

Initially, the items in Browser windows are displayed as large icons with their names underneath them. If you prefer, you can view the items in the window as small icons with their names to the right. You can also view the items in the window in an ordered list view, which can tell you additional information about the items in the window. List views are also used in query windows, as well as some panels, such as the Open and Save panels.

You change views by choosing Icon View, Mini Icon View, or List View from a window's Window menu.

Working in Icon and Mini Icon View



Item's icons are easiest to recognize in icon view.

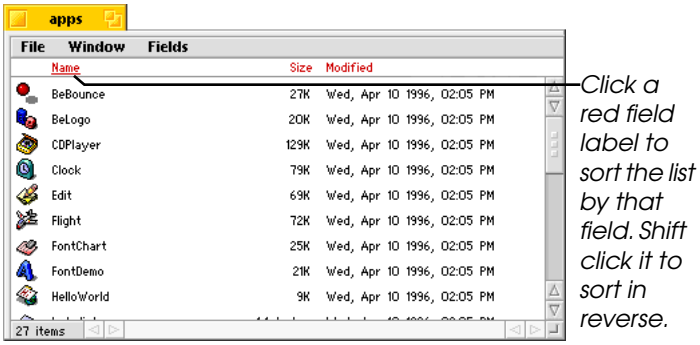
It's easiest to work with a lot of icons in mini icon view.

When you're working in icon view or mini icon view, you can move one or more items in a window to a new location in that window—perhaps to group related items in the same part of the window. Simply drag an item to the location you want. If more than one item is selected, they all move when you drag one of them. If you hold down the Alternate key while you drag icons, they snap to an invisible grid when you release the mouse button—this makes the arrangement of icons in the window more orderly.

Note: If you drag one or more items so that the cursor is over a folder icon when you release the mouse button, you copy or move the items to that folder. For more information, see "Moving and Copying Items" on page 46.

In icon view and mini icon view, you can use commands in the Window menu to straighten-up the arrangement of all the icons in the window. Choose Clean Up to align each icon in the window to the nearest location on the invisible grid. If you hold down the Shift key, the Clean Up command changes to Clean Up All. Choosing this command sorts the icons in the window in alphabetical order on the grid.

Working in List View



A lot more information about each item is visible in list view

When you're working in list view, you can change how the items are ordered in the list. Initially, items are sorted alphabetically by name, and the "Name" field title is underlined at the top of the list. If you click on another field title, the items in the list are sorted in a way that is appropriate for that field, and the field title you clicked becomes underlined. If you click a second time on a field title (or hold down the Shift key while you click on a field title the first time), the items are sorted in reverse order.

You can choose the fields you want to see in list view from the Fields menu. Fields shown in the window are checked in the menu.

Closing a Window

When you're done working with the contents of a window, you can close the window to make more room for other windows on the desktop. To close a window, click its close button. In many applications you can also choose the Close command, which is usually in a window's Window or File menu.

If you're working in an application where you need to save the changes you make, a panel asks if you want to save any unsaved changes when you close a window.

In many Be applications—such as the preferences applications—when you close an application's last (or only) window, the application quits.

Working with Menus

Be menus contain commands, lists of applications, settings, and other items.

You work with menus in three main places:

1. At the top-left corner of the screen.

   The application list and the main application menu, or "main menu," are at the top-left corner of the screen. The application list has the Be logo as its title. It's described in "Switching Applications" on page 56. The main menu contains menu items for the active application, that is, the application you're currently working in. The main menu's title is the logo of the active application.

2. In windows.

   Most of an application's menus are along the tops of its windows.

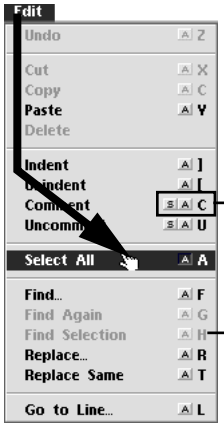
3. In panels.

   Pop-up lists and other controls in panels work very much like menus.

You can navigate menus and choose items from them in a number of ways, using the mouse, the keyboard, or a combination of the two.

Navigating Menus by Dragging

You can press a menu title, drag to the menu item you want, and release the mouse button to choose that menu item. If you release the mouse button without choosing a menu item, the menu closes.



To choose a menu item by dragging, press the menu title, drag to the item you want, and release the mouse button

If a menu item has a keyboard shortcut, it's indicated by icons for the keys you press.

You can't choose dimmed items, usually because they're not relevant to what's selected in the window.

When you drag through menus, you also see the *keyboard shortcuts* for the items in the menu. Keyboard shortcuts are combinations of keys you can press as a shortcut to choosing the menu items with the mouse. The icon represents the Command key (either of the keys labeled "Alt" on either side of the space bar on most PC keyboards). The icon represents the Shift key. Choosing menu items from the keyboard is described in "Navigating Menus from the Keyboard" on page 40.

Navigating Context-Sensitive Menus

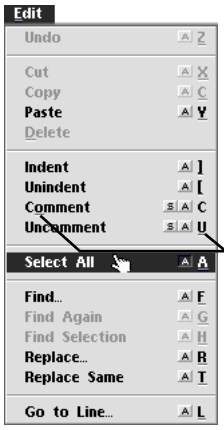
In some applications, such as the Browser, you can open a menu by selecting an item or positioning the cursor over it and then pressing a mouse button. For example in the Browser, if you position the cursor over an icon in the dock and press the secondary mouse button, a menu with items relevant to the item in the dock opens underneath the cursor. Context-sensitive menus are also available in the Open and Save panels. You can drag to a menu item you want and release the mouse button to choose it. If you don't want to choose one of the items in the menu, drag the cursor out of the menu before you release the mouse button.

Initially, the secondary mouse button is the right mouse button on a three-button mouse, but you can rearrange the buttons and their roles with the Mouse preferences application. See “Mouse” on page 99 for more information.

Navigating Menus by Clicking

You can click a menu’s title to open the menu and view its contents. Then you can click a menu item to choose it.

When you click a menu title, the menu stays open until you choose a menu item, click another menu title, or click elsewhere on the screen. When you move the cursor (without pressing a mouse button) over items in an open menu, menu items highlight underneath the cursor and submenus open. You can press the Enter key on the keyboard to choose a highlighted menu item.

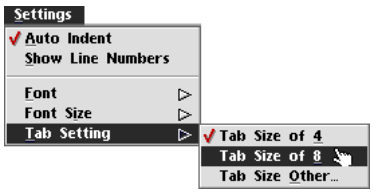


Click a menu title to open the menu, then click the menu item you want to choose. Or move the cursor over an item in an open menu and press the Enter key.

The underlined character in a menu item indicates its “trigger,” the character you can type to choose that menu item when the menu is open.

When you click a menu title, you see the menu’s *triggers*. Triggers are characters you can type when a menu is open to choose a menu item by typing instead of clicking. Triggers are indicated by underlined letters in menu titles and menu items. For more information, see “Navigating Menus from the Keyboard” on page 40.

An arrow next to a menu item indicates that it's a submenu. Move the cursor over a submenu title to open it. Then click the menu item you want to choose in the submenu.



Navigating Menus from the Keyboard

You can press and release the Command key (either of the keys labeled "Alt" on either side of the space bar) to display the triggers for the menu titles in the active window. Triggers are underlined letters in menus that indicate the character you can type as an alternative to clicking a menu title or menu item. Type the trigger letter for the menu you want to open. Then type the trigger letter for the menu item you want to choose.

Most menu items have a trigger. If you want to use the keyboard to choose a menu item that doesn't have a trigger, you can use the arrow keys to select the menu item and then press Enter.

When you open a menu with the Command key or by clicking the menu title, you can also use the arrow keys on the keyboard to navigate menus. Use the left and right arrow keys to open different menus and submenus. Use the up and down arrow keys to highlight menu items. When the menu item you want to choose is highlighted, press Enter.

Choosing Menu Items by Shortcut

Many menu items have shortcuts: Combinations of keys you can press to choose a menu item, as a shortcut to choosing it by dragging, clicking, or by navigating to the menu item from the keyboard. If a menu item has a shortcut, it's displayed to the right of the menu item in the menu. The A icon represents the Command key (either of the keys labeled "Alt" on either side of the space bar on most PC keyboards). The S icon represents the Shift key. Shortcuts are always displayed in menus as uppercase letters, though you only need to press the Shift key along with the Command and letter keys if the shortcut includes the S icon.

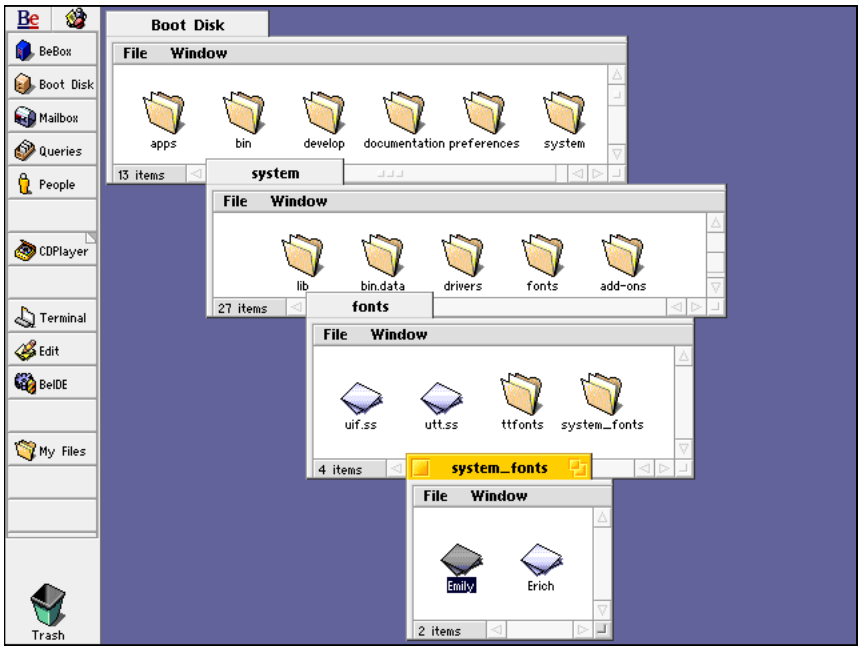
Working with the Browser

You use the Browser to organize your work: Files, applications, database items, and so on. The Browser is also the main tool you use to start applications.

Understanding the Hierarchical File System

Applications, documents, and other files in the Be file system are stored on disks. Because the number of files can become quite large, it's convenient to sort them into related groups. To do this, the Be OS—like many other operating systems—uses the metaphor of folders (also called "directories"), where you can keep together related files, and even related folders.

In the following illustration, the file **Emily** (selected at the bottom of the screen) is in the **system_fonts** folder, which is in the **fonts** folder, which is in the **system** folder, which is on the disk named **Boot Disk**. (In this book, **boldface** is used for the names of files and folders.)



There's another way of expressing the location of items in this hierarchy that you use, for example, when you're working in the Terminal application: You use an item's *path name*. A path name is a series of folder names separated by slashes. For example, the path name of the Emily font in the illustration is: **/system/fonts/system_fonts/Emily**. The path for the folder the fonts folder is in is often represented as **/system**. The leading slash indicates that the **system** folder is in the *root* of the boot disk: That is, on the disk you booted from but not in a folder.

This system gets more complicated when you're working with more than one disk. There are also shortcuts to represent items in the same folder and items in folders closer to the root of the hierarchy. For more information, see "Terminal" on page 96.

Getting Information About an Item

You can get information about an item in a Browser window or the dock by selecting it and choosing Get Info from the File menu.

The information you get depends on the item. For disks, you get information about their capacity and free space. For files, you get information about their size, when they were created and modified, and their location in the file system. For applications, you get this same information, plus a version number.

Opening Folders

You double-click a folder to open a window that shows the contents of the folder. The title of the folder window is the name of the folder.

A folder that contains another folder is often called the *parent* of the folder it contains. When you're viewing the contents of a folder in a window, you can open its parent folder by choosing Open Parent from the Window menu.

You can also open a folder's parents from the pop-up menu that opens when you press the lower-left edge of a folder window, the area that normally lists the number of items in the window.

Press the lower-left corner of a Browser window to open a menu that lists the parents of the folder whose contents are shown in the window.<<Maybe not>>

Working with Disks

In addition to the BeBox boot (or "startup") disk, you can have many disks and their contents available in the Browser and in other applications: Additional SCSI or IDE hard disks, CD-ROMs, floppy disks, and so on. To make a disk available in the Browser and other applications, you *mount* it.

All currently mounted disks are listed in the BeBox window, which you can open by double-clicking the BeBox icon in the dock.

Tip: If you've removed the BeBox icon from the dock, you use the Browser's Find command to open a query window, from which you can drag its icon back to the dock, or just open the icon. For more information, see the tip in "Opening a Query" on page 54.)

Before you can mount any disk the first time, you must use the Installer application to *initialize* the disk, that is, to prepare it to work with the Be OS. For information on initializing disks, see "Installer" on page 83. You can't mount an uninitialized disk—even if the disk is a hard disk that was connected to the BeBox and turned on when you turned on the BeBox, its icon doesn't appear in the BeBox window until you initialize it.

You work with mounted disks in much the same way as you work with folders: Double-click them to open them, drag files onto them to copy the files onto the disk, and so on.

Mounting a Disk

The easiest way to mount a SCSI or IDE disk is to have it connected to the BeBox—and if it's an external device, turned on—when you turn on the BeBox. Icons for mounted disks are in the BeBox window, which you open by double-clicking the BeBox icon in the dock.

To mount a floppy disk, insert it in the floppy disk drive and choose Mount Floppy Disk from the BeBox window's File menu. The disk's icon appears in the BeBox window, where you can work with it in the same way you work with a folder icon.

Note: The current release of the Be OS only supports high-density floppy disks. (These are usually labeled "HD".)

To mount all disks connected to the BeBox, including CD-ROMs, other kinds of removable disks, hard disks, as well as floppy disks, choose Mount All Disks from the BeBox window's Disk menu.

A disk is also always mounted after you initialize it with the Installer application.

Unmounting a Disk

To unmount a floppy disk, CD-ROM, or any other kind of disk, select it in the BeBox window and choose Unmount Disk from the Disk menu (or just drag the disk icon to the Trash icon). The disk's icon is removed from the BeBox window, and any windows for folders on that disk are closed. If the disk is a floppy disk or a removable disk of another kind (such as a CD-ROM), you can then eject it from its drive by pressing the drive's Eject button.

Renaming Items

You can rename any file, folder, or other item in a Browser window and in some panels, such as the Open and Save panels. To change an item's name, select the name by clicking it and type a new one. You can also select an item in a Browser window and choose Edit Name from the window's File menu to select the name.

You type and edit an item's name in the same way as you type and edit any other text in Be applications. For more information, see "Working with Text" on page 61.

You can name an item anything you want, as long as there's no other item in the same folder with that name already. You can use up to 58 letters, numbers, and symbols—anything you can type, except the asterisk (*****), slash (**/**), colon (**:**), and backslash (****) characters.

Tip: If you change your mind about renaming a file or folder while editing its name, just press the Escape key to restore the original name.

Creating a Folder

You can create a new folder to store related files and even other folders.

1. Open the folder where you want to create the new folder, so the folder's window is the active window.

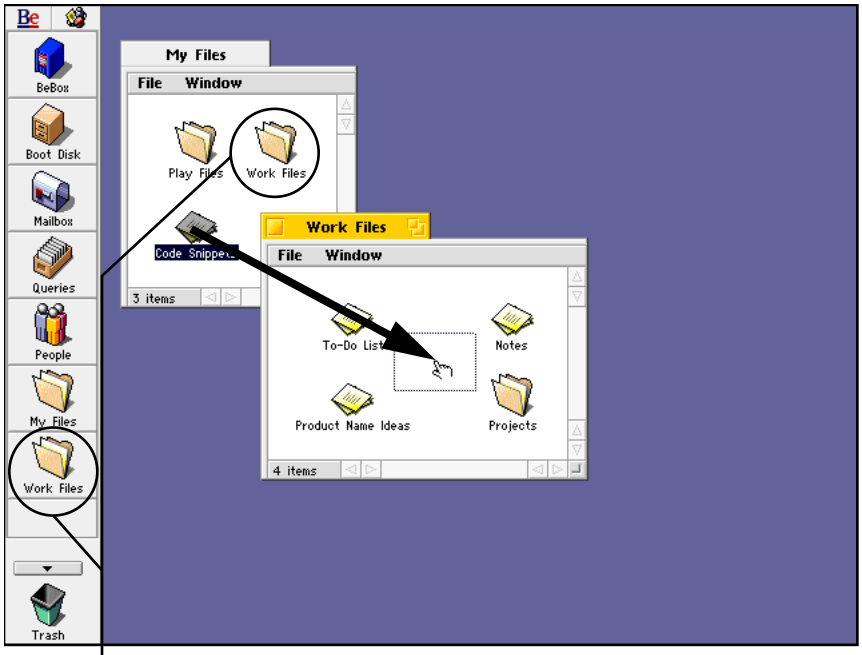
   For more information on windows, see "Working with Windows" on page 27.

2. Choose New Folder from the folder window's File menu.

   A folder named "New Folder" is created. The folder's name is selected, so you can simply type a name and press Enter to name the folder.

Moving and Copying Items

You can move items (such as a file or folder) from one folder to another. Simply drag the item into the window or onto the icon of the folder where you want to move it. You may have to arrange the windows on your desktop so both the item you want to move and the window or folder where you want to move the item are visible.



To move an item, drag it into the window where you want to move it. Or drag it onto a folder icon in a window or a folder icon on the dock.

In this example, you could also move the Code Snippets file into the Work Files icon in the same window as the Code Snippets file, or onto the Work Files icon on the dock. These both would have the same effect as dragging the file into the Work Files window.

If the folder you drag an item onto is on a different disk, the item is copied instead of moved. That is, a new copy of the item you drag is created where you dragged it, but the original copy remains in the folder where you dragged it from.

If you want to have copies of the item you're moving in both the source and destination folders on the same disk, hold down the Control key while you drag the item. This forces a copy whether the destination is on the same disk or not.

If you're copying or moving a lot of files, a status window opens with updates on the progress of the operation. If you click Cancel in the status window, the operation is stopped in midstream, but whatever has been moved or copied so far stays moved or copied—it's not undone.

Duplicating Items

You can create a copy of any item by selecting it and choosing Duplicate from its window's File menu. The duplicate is created in the same folder as the original, and has the same name, with "copy" appended. If you create more than one duplicate of the same item, the duplicates have "copy 2", "copy 3", and so on, appended to the original name.

Deleting Items

You can delete files, folders, and most other items you see in the Browser by dragging them to the Trash icon on the dock. You can also put items in the trash by selecting them in a Browser window and choosing the Move to Trash command from the window's File menu. When there are items in the trash, the Trash icon appears to have trash in it:

The trash is empty

There are items in the trash

Items you put in the trash aren't deleted permanently. You can double-click the Trash icon to see what's in the trash, and drag items back into a Browser folder window if you decide you'd rather not delete them after all.

When you're sure you want to remove items permanently, choose the Empty Trash command from the Browser's main menu.

If you make a habit of putting items in the trash and never emptying it, the trash eventually takes up a lot of disk space, so it's a good idea to empty the trash periodically.

Note: Dragging an item from the dock to the Trash icon doesn't put it in the trash: It just removes the item from the dock.

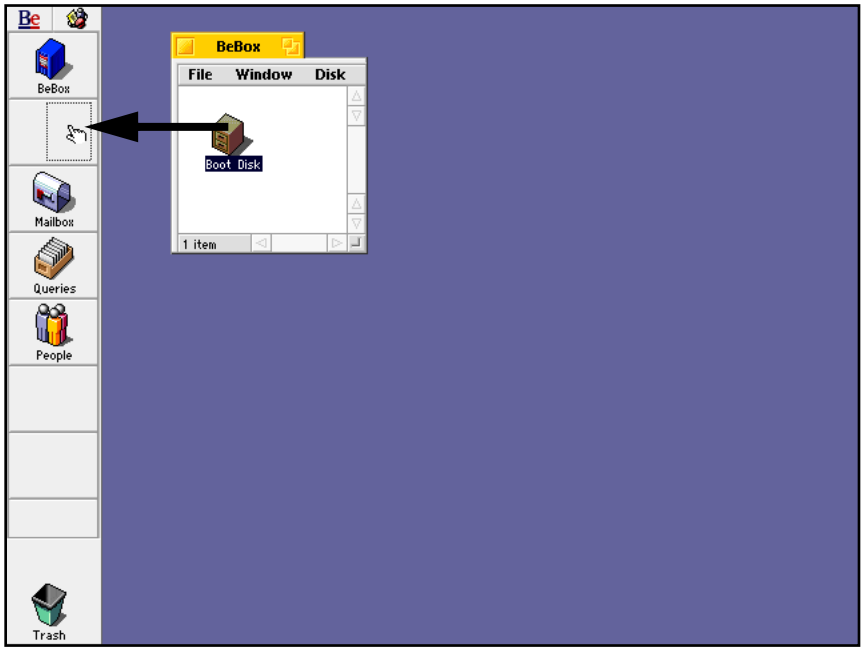
Working with the Dock

The Dock is a convenient place to store icons of the applications, files, folders, and other items you use most often. It's often easier to double-click an icon in the dock than to find the original item in its folder.

An icon in the dock isn't the real item it represents—the original remains in its folder. If you rename the original, the name changes in the dock. Similarly, if you move the original, the icon in the dock still represents the original, but in its new location. Conversely, you can't rename or destroy the icon in the dock, you can only open it, move it to a new location in the dock, or remove it from the dock.

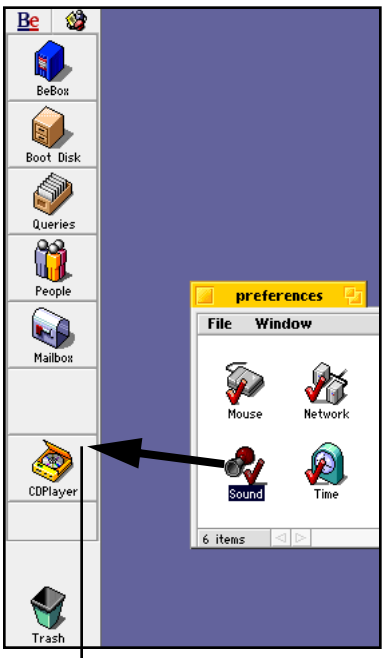
Adding Items to the Dock

To add an item to the dock, drag it onto one of the dock's slots.

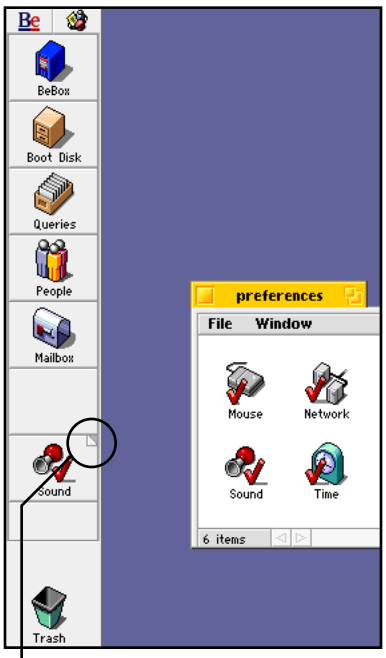


If you drag a file onto an application icon in the dock, the application tries to open the file. If you drag any item onto a disk or folder icon, the item is moved or copied to that disk or folder. If you drag any item into a space on the dock that already has an icon in it, but not directly onto the item, both items occupy that space in the dock, and the slot get's

a dog-eared corner to indicate there's more than one item in that slot. Click the dog-eared corner to see the other items in that slot.



Drag an icon to the upper-right corner of a slot in the dock (take care not to drag the icon onto the icon that's in the slot already).



The "dog-eared" corner indicates there's more than one icon in the slot. Click the dog-ear to make each icon visible in turn.

Rearranging Items on the Dock

To move an item on the dock to another slot on the dock, drag it to the slot you want.

Removing Items from the Dock

To remove an item from the dock, select it and choose “Remove from Dock” from the Browser’s main menu. Or drag the item to the trash (this doesn’t throw out the original item in the file system—it just removes the icon from the dock).

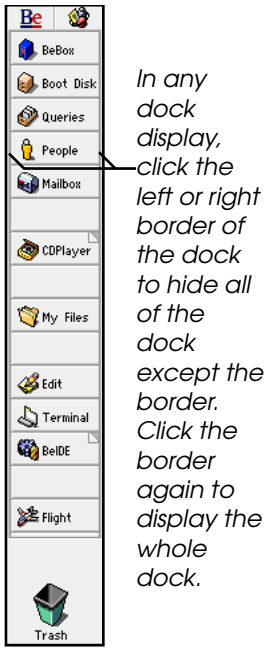
Changing the Dock Display

You can change how the dock displays its contents. You can also hide the dock from view.

Initially, items in the dock are displayed as large icons with their names underneath. You change how items are displayed by choosing options from the Dock Display submenu in the Browser's main menu.

Click a dock scroll button to see icons that don't fit in the current dock display

Icons dock display



In any dock display, click the left or right border of the dock to hide all of the dock except the border. Click the border again to display the whole dock.

Mini Icons with Names dock display

In Mini Icons dock display, the main menu moves under the application list.

Mini Icons dock display

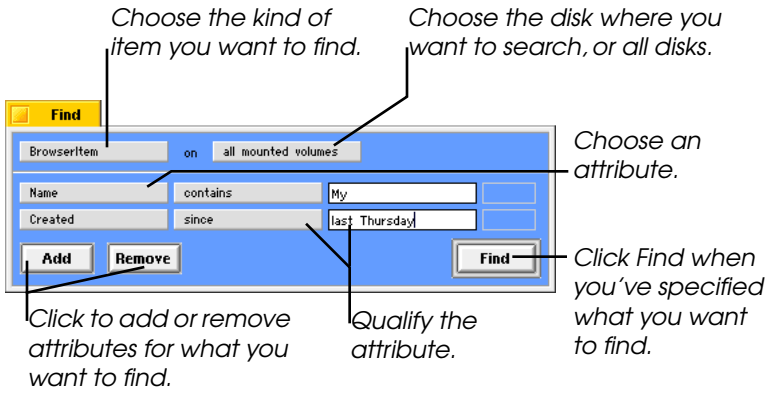
To hide all but a border of the dock from view, click the left or right border. Click the border again to display the whole dock.

Finding Items

You can use the Find command in the Browser's main menu to search for files, folders, and other items in the Be file system (that is, items stored as files on a disk). You can also search for entities in the Be database, such as the names of audio CD tracks you enter in the CDPlayer application. When you choose Find, the Find window opens, where you specify in greater or lesser detail the attributes of what you're looking for. These specifications, called *queries*, are saved in the Be database, so you can reuse them later.

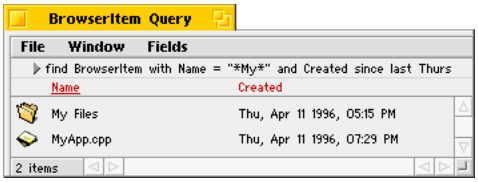
To find an item:

1. Choose Find from the Browser's main menu.
2. Specify the attributes of the item or items you want to find.



3 Click Find.

A query window opens, listing all the items that match the attributes you set in the Find window.



Query windows are very similar to other Browser windows, except they have a gray background and only work in list view. You can double-click items to open them, drag items to the dock and to other folders, and so on. (Think twice before dragging items from a query window to a location in the file system: Query windows usually contain items from many parts of the file system, and once you move them, you can't move them back as a set.)

Caution: Though you can see and edit a text version of the query in the upper part of a query window, resist the temptation to experiment. The query language is incomplete in the prerelease version of the software and you can easily create an irreparable query.

Opening a Query

The Be database saves every query you create—until you decide to delete it. You can double-click the Queries icon in the dock to open a window that lists all your queries. Double-click a query in the window to open another query window, which shows the current result of the saved query.

You can drag queries you don't want any more from the Query window to the trash.

Tip: The Queries icon is part of the Be database. Because it isn't a file or folder, you can't find it on a disk. If the Queries icon isn't in the dock, you can use the Find command to search for a BrowserItem named "Queries". Then you can double-click the Queries icon in the resulting query window or drag the icon back onto the dock. You can also use this technique to restore the BeBox, People, and Mailbox icons to the dock.

3 Using Be Applications

The Be OS comes with a number of applications, some that you use to create documents, some you use to configure how you prefer to work with the BeBox, and others that demonstrate the potential of the BeBox.

Most applications are in the **/apps** folder, but you can also find applications in the dock, in the **/preferences** folder, and in other locations.

Application Basics

You start applications, open files, type and edit text, save files, quit applications, and perform many other tasks the same way in most applications. This section introduces you to these basic techniques.

Starting an Application

You can start an application by double-clicking its icon. Other ways to start an application include selecting its icon in the Browser and choosing Open from the File menu, and selecting an application icon and pressing Enter.

You also start an application when you open a file—if the application you use with the file isn't already running.

When an application is running, its name is added to the application list—the menu at the top-left of the screen that lists all running applications.

Getting Information About an Application

You can get information about an application by choosing the About command from the application's main menu (the menu with the application's icon, to the right of the application list at the top-left corner of the screen). About commands usually appear in the main menu as "About" followed by the name of the application. For example, "About Edit" is in Edit's main menu.

Switching Applications

You can have many applications running at once on the BeBox. The Be OS and the applications themselves understand how to share processors, memory, and other resources, so many applications can do many things simultaneously.

While all running applications are "active," in the sense that they're all doing whatever you told them to do, only one application has its main menu to the right of the application list in the top-left corner of the screen—the main menu has the application's icon as its title. This is the application that responds to what you type on the keyboard, and if it has windows open, one of its windows is the active window.

There are a number of ways to switch among applications. The simplest is to click in a window belonging to the application you want to work in. Another simple way is to choose an application from the application list—the menu with the Be logo icon at the top-left corner of the screen. You also switch applications when you open a file.

Quitting an Application

When you're done working with an application, you can quit it. To quit an application, choose its Quit command. In most applications the Quit command is in the main menu.

In many applications, closing all an application's windows causes it to quit.

It's a good idea to quit an application if you're not going to be using it for a while. Quitting an application frees up resources for the applications you're still using.

Creating a New File

When you start some applications—such as Edit—they open a blank, unnamed file in a window. You can do work in this file and then save it (as described in "Saving a File" on page 57). In most applications, you can create and open another new file by choosing the New command. The New command is in different locations in different applications, but in most applications it's in the application's main menu or in the File menu in the application's windows.

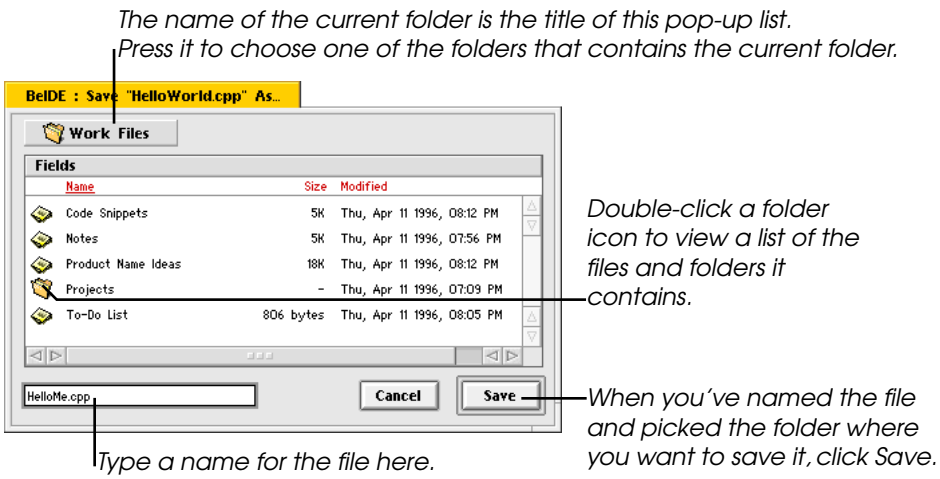
It's important to save a new file as soon as you create work in it that you don't want to lose.

Saving a File

When you work in a file, your changes aren't stored permanently until you save them. If you experience a power outage, if the BeBox crashes, or if you accidentally press Control-Alt-Delete, any changes you haven't saved are lost. Saving a file puts a copy of the file with your latest changes on a disk, where it's stored permanently—until you change the file and save it again.

If you make changes to a file that already has a name and location on a disk, you save changes by choosing the Save command. The Save command is in different menus in different applications, but it's most often each application window's File menu.

If you create a new file or make changes to a file that doesn't already have a name and location on a disk, choosing Save the first time opens the Save As panel, where you can give the new file a name and pick the folder and disk where you want to save it.



The central portion of the Save panel is a list view of Browser items. You can work in this list view in much the same way as you work in a folder or query window in list view in the Browser: You can select items, rename them, change how they're sorted in the list, and choose fields you want to view in the panel, and use the secondary mouse button to open context-sensitive menus (for more information on list views, see "Changing Views" on page 34).

To view the contents of a folder that contains the folder listed in the Save panel, choose a folder name from the pop-up list in the upper-left corner of the Save panel.

You can use the keyboard to navigate the controls in the Save panel: Press Tab to select the list view or the name field as the target for what you type (a gray rectangle surrounds the area your typing will affect). When the list view is selected, you can select an item in the list by typing the first few characters in its name, use the arrow keys to select items, and open a selected folder by pressing Enter (the Save button changes to Open when a folder is selected). You can also open the folder that contains the current folder by pressing Command–up arrow (the Command key is either of the keys labeled "Alt" on most PC keyboards). To close the panel without saving anything, press Escape or Command-w (this is the same as clicking the Cancel button).

When you name a file, select the folder where you want to save it, and click Save, the file is stored safely on the disk you selected and the title of the file in its window's tab shows the name you gave the file.

You can also use the Save As command to save a copy of a file with your latest changes under a new name or in a different folder or disk. When you do, the original file remains unchanged on disk, and the new file with the new name is displayed in the window you were working in.

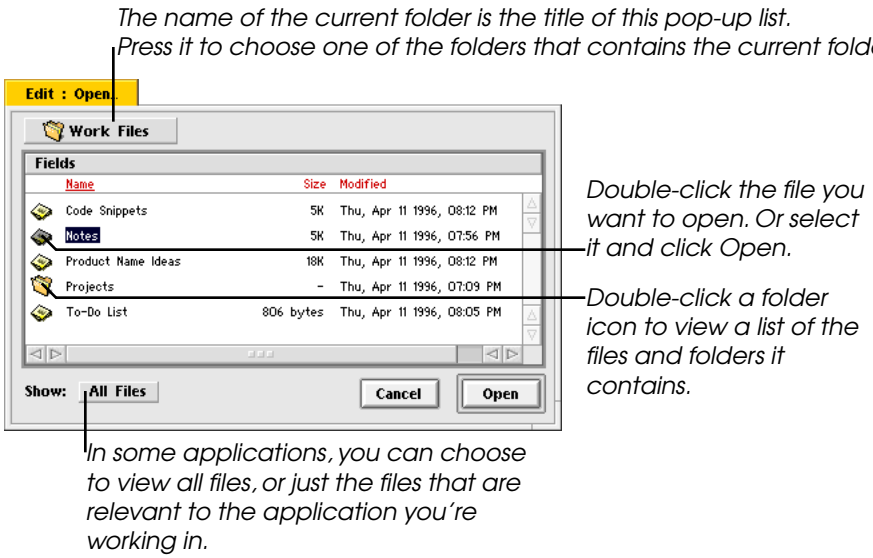
Opening Files

There are many ways to open files. The simplest is to double-click a file's icon. You can also select a file's icon in the Browser or the dock and choose Open from the File menu, or select the file and press Enter.

Most files have an application they're associated with, usually the application they were created in. When you double-click a file, it opens in this application. A file's icon usually resembles the icon of application it's associated with, so you can usually tell what application a file will open in by looking at its icon. If a file doesn't have an application associated with it, you're asked if you want to use Edit to open the file when you double-click it.

You can drag a file onto an application icon to open the file in that application—even if the file isn't normally associated with that application. This is a good way to open a file in an application it normally isn't associated with: For example, you can use this technique to open a script file you want to read or edit in the Edit application—if you just double-click a script file, the script runs, instead of opening in a document window. (Of course, the application has to know how to open the file, so you can't, for example, drag an ImageViewer file onto the PlaySound icon.)

If you're working in an application, you can open a file with the application's Open command. The Open command is in different menus in different applications, but it's most often in an application's main menu or File menu. When you choose Open, a panel opens where you can select the file you want to open.



The central portion of the Open panel is a list view of Browser items. You work in this list view in much the same way as you work in a folder or query window in list view in the Browser: You can select items, rename them, change how they're sorted in the list, choose fields you want to view in the panel, and use the secondary mouse button to open context-sensitive menus (for more information on list views, see "Changing Views" on page 34).

To view the contents of a folder that contains the folder listed in the Open panel, choose a folder name from the pop-up list in the upper-left corner.

You can use the keyboard to navigate the controls in the Open panel. You can select an item in the list by typing the first few characters in its name, use the arrow keys to select items, press Tab or Shift-Tab to select the next or previous item, and open a selected folder by pressing Enter. You can also open the folder that contains the current folder by pressing Command–up arrow (the Command key is either of the keys labeled "Alt" on

most PC keyboards). To close the panel without opening anything, press Escape or Command-w (this is the same as clicking the Cancel button).

When you double-click a file's icon in the panel (or select the file and click Open), a window opens that displays the contents of the file.

Working with Text

You use the keyboard and mouse to create and edit text in a lot of locations in the Be OS: You edit file names, text files, settings in panels, and so on.

You edit text by selecting the place you want to type and then typing. You can also copy text from one place and move it to another, or delete text.

The location on screen where what you type (or text you move) will be entered is indicated in two ways. The first is by a blinking vertical bar, called the *insertion point*. The second is by highlighted text you want to replace.

Selecting Text

There are a number of ways to select text.

Do This	To
Click	Select an insertion point
Double-click	Select a word
Triple-click	Select a paragraph
Drag across text	Select a character at a time
Double-click and drag	Select whole words at a time
Triple-click and drag	Select whole paragraphs at a time
Hold down the Shift key	Extend or reduce a range of selected text

In many applications, you can choose the Select All command to select all the text in a window or document; the Select All command is usually in an application's Edit menu.

Typing Text

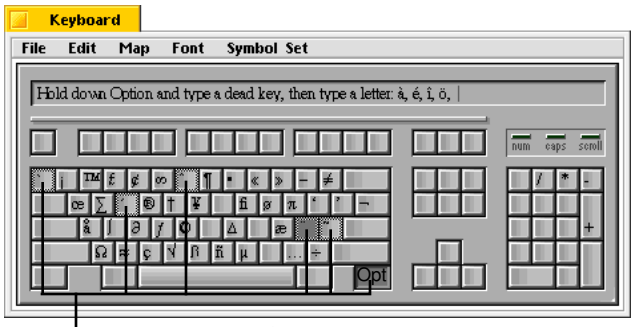
You type text in documents, when you name or rename files in the Browser or the Open and Save panels, and in many other places.

To type text, click to select an insertion point or select text you want to replace. Then type what you want. If you select text, what you type replaces the selection.

Typing Special Characters

You type accented characters, symbols, and other special characters by holding down the Option key (the key labeled "Ctrl" on the right side of most PC keyboards) while you type one or more other keys. You type most accented characters by typing a *dead key* with the accent you want, followed by the letter you want accented. Dead keys are keys that when you press them, don't place a character on the screen until you type another character. For example, to type *é*, you press Option-e and then you type **e**.

In the Keyboard application, dead keys are indicated by a checked border. The keyboard application is a good tool to use to find the accents and other special characters you want. For more information, see "Keyboard" on page 86.



Hold down the Option key while you type a dead key; then type a letter.

Copying and Moving Text

You can *copy* selected text and then *paste* (insert) it, someplace else. Select the text you want to copy and choose Copy from a window's Edit menu. Then select an insertion point or text you want to replace and choose Paste from the Edit menu. You can paste text in the same document you copied it from, or in any other application where you work with text, including the names of items in Browser windows and text fields in panels.

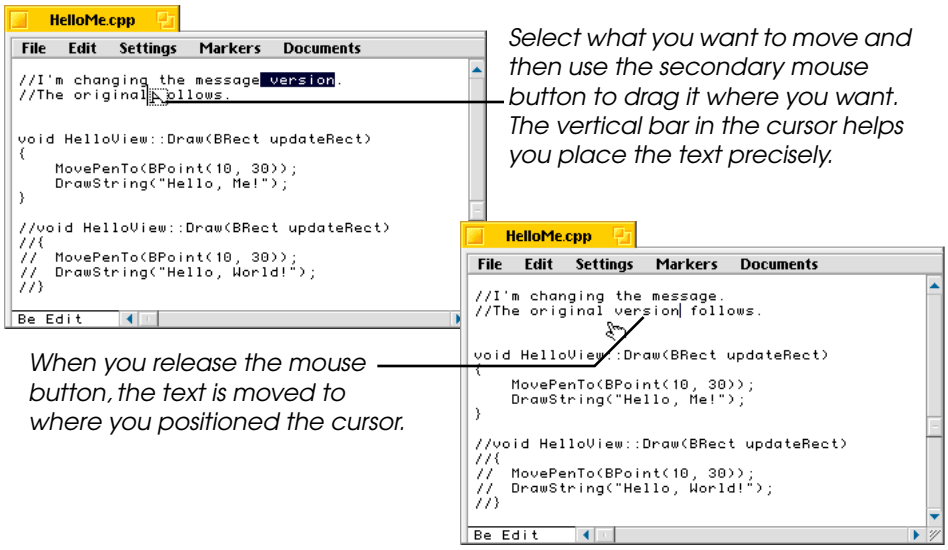
If you want to move text rather than copy it—that is, remove the text you select and paste it elsewhere—choose Cut from the Edit menu instead of Copy.

Tip: When copying or moving text, be mindful of the spaces before and after what you copy, and where you intend to paste it.

Text you cut or copy is stored in a special part of memory, called the *clipboard*. What you last cut or copied remains on the clipboard until you cut or copy something else, so you can keep pasting what you cut or copied repeatedly, without having to cut or copy the original again.

Moving Text with the Mouse

You can use the secondary mouse button to move selected text by dragging it. When you start to drag selected text, the cursor changes shape to help you position the text exactly where you want it.



Initially, the right mouse button is the secondary mouse button on a three-button mouse, but you can change which button is the secondary mouse button with the Mouse preferences application. For details, see “Mouse” on page 99.

Deleting Text

You can delete text in a number of ways. The simplest is to press the Delete key to erase the text to the left of the insertion point letter by letter. (Depending on the keyboard you’re using, the Delete key may be labeled “Delete,” “Backspace,” or just have a left-pointing arrow. See “Using the Keyboard” on page 25 for more information on identifying keys on the keyboard.)

If you select text, you can delete all of it at once by pressing the Delete key. Many applications also have a Delete command in their windows’ Edit menus. The Delete command acts on selected text just like pressing the Delete key.

Undoing

In many applications, you can undo your most recent action by choosing Undo from the Edit menu. For example, if you delete text in an Edit file accidentally, choose Undo to undelete it.

Using the Be Applications

Most of the applications and demos that come with the Be OS are in the **/apps** folder. This section briefly explains how to use each application in **/apps**, in alphabetical order.

BeBounce

BeBounce demonstrates how applications can communicate with each other in the Be OS. When you start BeBounce, a window with a bouncing ball opens. If you start a second copy of BeBounce, its window has no ball, but you can move the windows near each other to open a path between them. The ball can bounce from window to window through this path. You can't have more than two copies of BeBounce running.

For details on the code that makes BeBounce work, see the article "Opening the BeBox," in the January 1996 issue of *MacTech* magazine. A copy of the article will soon also be available on the Be web site (**http://www.be.com**). Commented source code for BeBounce is included with the Be OS, in **/develop/projects/BeBounce**.

BeLogo

When you start BeLogo, it opens a window with a 3D rendition of the Be logo, which rotates in three dimensions on a black background. The logo is a 3D model, which is mathematically rotated at 120 frames per second. BeLogo was designed to take advantage of all the CPU resources it can get, so while it's an interesting way to test the power of the BeBox, you may not want to leave it running when you're trying to get real work done.

CDPlayer

If you have a CD-ROM drive connected to the BeBox, you can use the CDPlayer application to play music and other audio CDs through the internal speaker and line-out ports. You can also use CDPlayer to copy audio data from an audio CD to a file on a disk.

Note: CDPlayer works best with Toshiba CD-ROM drives. Some of its features—such as fast forward, fast reverse, and saving audio data—may not be available if you're working with a drive from another manufacturer.

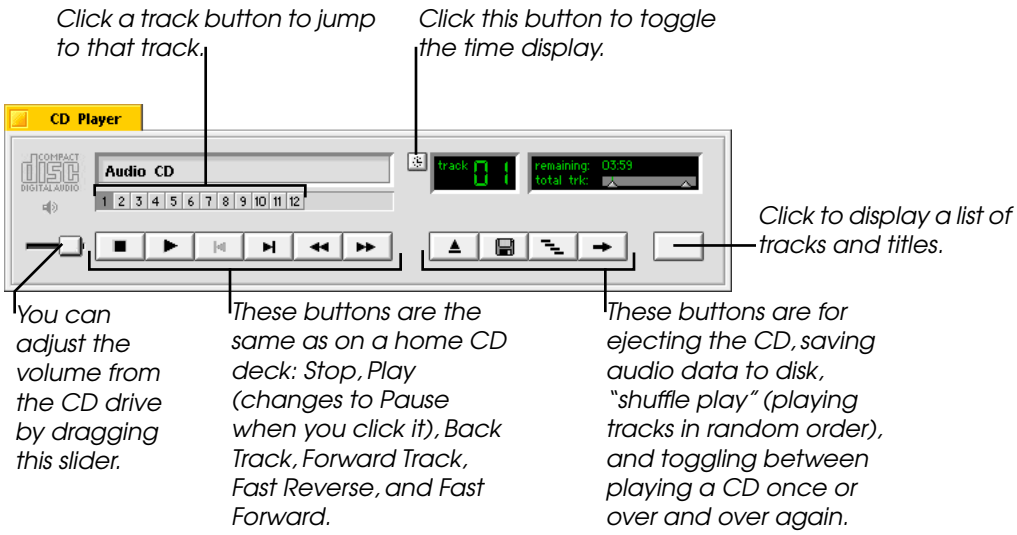
Playing CDs

When you start CDPlayer, it opens a window with controls very like those on a home CD deck. If you have a CD deck, you'll recognize the Stop, Play, Back Track, Forward Track, Fast Reverse, Fast Forward (or "scan"), Eject, and other buttons.

If you have more than one CD-ROM drive connected to the BeBox, choose the one with the CD you want to play from the Devices submenu in CDPlayer's main menu. You can also open a CD-Player window for each CD-ROM drive: Choose the CD-ROM drive you want to open a window for from the New Player submenu in CDPlayer's main menu. CD-ROM drives are listed in these two menus by their SCSI IDs.

The Fast Forward and Fast Reverse buttons scan the disk at double speed; you can hear what you're scanning. Holding down the secondary mouse button when you click Fast Forward or Reverse scans the disk at quadruple speed.

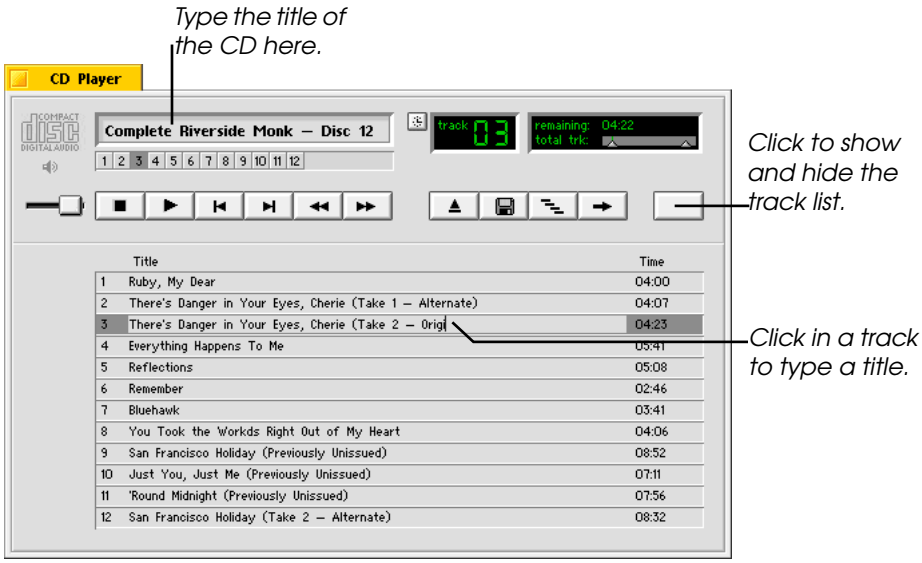
The volume slider adjusts the output of the CD-ROM drive. You use the Sound preferences application to control the input and output volumes for the BeBox, as described in "Sound" on page 109.



Note: In the prerelease version of the software, the shuffle button isn't hooked up.

If you click the rightmost button in the CDPlayer window, a lower part of the window opens with a list of the tracks on the CD. You can replace "Track 1," "Track 2," and so on with a description of each track—perhaps the title of each track from the play list. Click in the Title column and type the description you want. You can also replace "Audio CD" at the top of the window with the title of the CD. The information you type

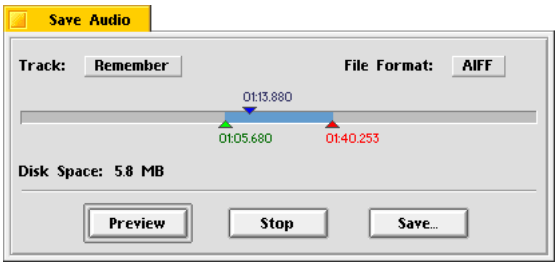
is stored in the Be database as “AudioCDs” data; the next time you use CDPlayer to play that CD, it recognizes the CD and includes the information you typed. You can also use the Browser’s Find command to search for AudioCDs data, such as a track or CD title you typed in a CD Player window (for more information on the Find command, see “Finding Items” on page 53).



Saving Audio Data

One button you won’t recognize from a CD deck is the Disk button. When you click the Disk button, the Save Audio window opens, where you can save some or all of a CD track to a file on a Be disk. Choose the audio format you want from the pop-up list: “Raw” and AIFF (Audio Interchange File Format) are supported now—WAVE (Waveform Audio File Format) files will be supported in the near future.

To save audio data, you need to connect the CD-ROM drive to the BeBox I/O board with a "pass-through cable." For details, see "Connecting a CD-ROM Drive to the I/O Board" on page 11.



Drag the green start marker and the red end marker to the starting and ending points in the track you want to save. The blue marker shows the current place in the track when you're playing it. The numbers next to the markers indicated the hours, minutes, and seconds from the beginning of the track. When the markers surround the section of the track you want to save, click the Save button to open a Save panel, where you name the audio file and select the folder where you want to save it (for more information on the Save panel, see "Saving a File" on page 57).

You can play the tracks you save with the PlayAudio application (in **/apps**). Simply drag the track you want to listen to onto the PlayAudio icon.

You can double-click the CDPlayer icon to start as many copies of CDPlayer as you want, so if you have more than one CD-ROM drive connected to the BeBox, you can listen to more than one CD at a time.

Clock

The Clock application is as simple as its name: When you start it up, it displays an analog clock. Clock has a number of faces. Click in the clock face to switch among them. If you prefer not to see a second hand, click in the face with the secondary mouse button—click again to restore the second hand.

Click to switch face styles. Click with the secondary mouse button to hide or show the second hand.

You set the date and time with the Time preferences application, as described in "Time" on page 110.

Edit

Edit is the Be text editor. You can use it to read, create, and edit text (ASCII) and script files.

Edit is not a word processor: It doesn't wrap long lines to fit in a window, you can't change font and paragraph settings for individual words or paragraphs, or do most of the other things you may be used to doing in a word-processing application. But Edit is great for editing programs and shell scripts.

You type text, copy and paste, type symbols and other special characters, just as you do anywhere you work with text in the Be OS. For details, see "Working with Text" on page 61.

If you have more than one Edit file open at once, you can switch from one to another by clicking in the window of the file you want to work in. You can also choose the name of the open file you want to work in from any Edit file window's Document menu.

Changing the Font

You can display an Edit file using any font. Choose the font, style, and size you want from the Font menu.

Note: Changing the font in an Edit file isn't the same as setting a font in a word processor—the font you choose is used for all the characters in the file.

For information on adding additional fonts to the BeBox, see "Adding Fonts" on page 112.

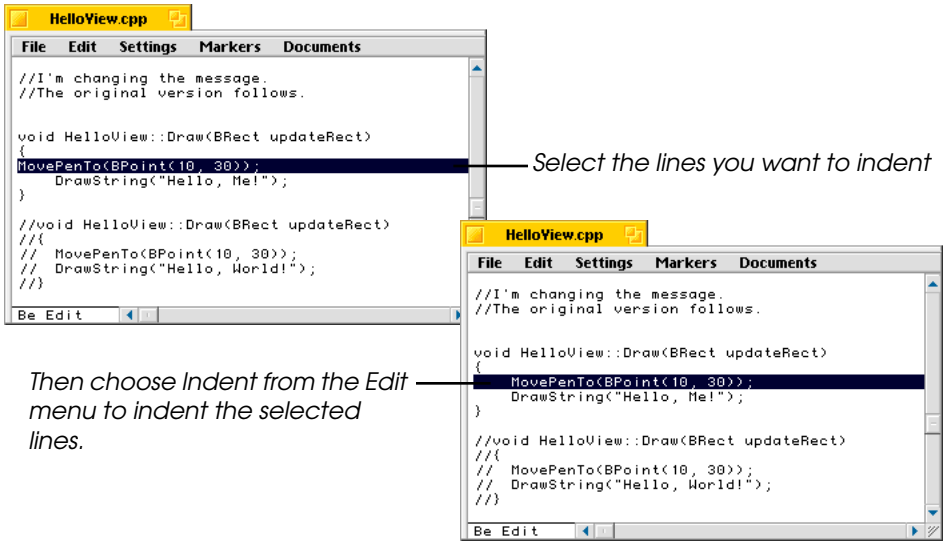
Automatically Indenting Lines

When you're writing code, it's often convenient to have lines that follow an indented line be indented automatically when you press Enter. Initially, Edit files are set to act this way. To end a series of automatically indented lines, press the Delete key at the beginning of the fist line you don't want indented.

You can turn off this behavior by choosing Auto Indent from the Settings menu (this unchecks Auto Indent in the menu).

Indenting and Unindenting Lines

You can indent selected lines by choosing Indent from the Edit menu.



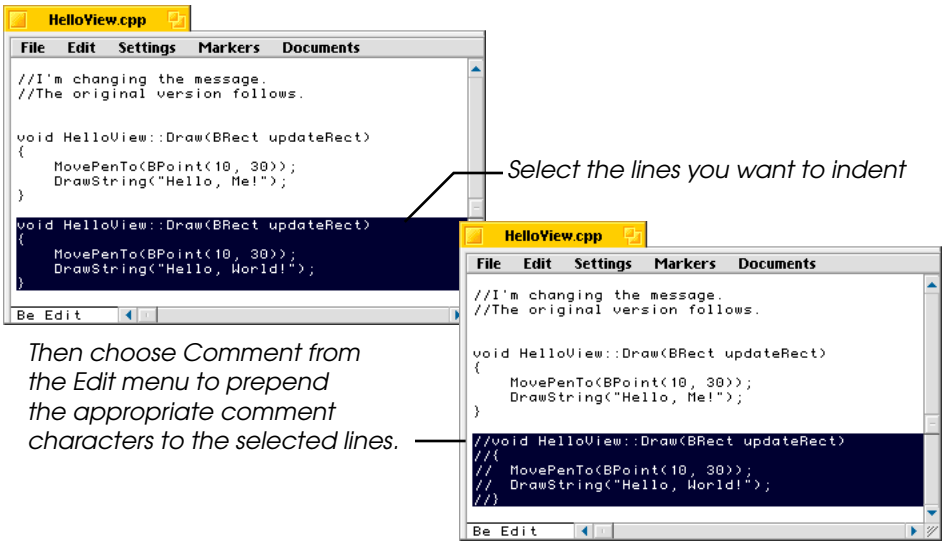
Unindent selected lines by choosing Unindent from the Edit menu.

Changing the Width of Tabs and Indents

Initially, when you press Tab or indent lines with the Indent command, each Tab character or indent moves lines four characters to the right. You can choose to indent eight characters instead of four by choosing “Tab Size of 8” from the Tab Setting submenu in the Settings menu. Or you can set the width you want by choosing Tab Size Other, entering a number for the width you want, and clicking OK. When you change the tab setting, all indented lines move to match the new setting.

Commenting and Uncommenting Lines

You can select lines in the programs you write and “comment them out,” that is, add the appropriate character or characters to the beginning of lines that you don’t want to get compiled, but rather to serve as comments.



The characters prepended to lines when you choose Comment depend on the kind of file you're editing, which Edit determines by the file's *extension*: the period and following characters at the end of the file's name.

File Extension	Comment Character(s)
.c, **.h**, or **.cpp**	**//**
.a	**;**
All others	**#**

You can remove the comment characters from the beginnings of selected lines by choosing Uncomment from the Edit menu.

Finding Text

You can find a string of characters in an Edit file.

1. Choose Find from the Edit menu.
2. Type the string of characters you want to find.

3. Check any Find options.

   Check Search Backwards to search for the characters in the file that are before the insertion point (in lines with lower line numbers), rather than after the insertion point.

   Check Wrap-Around Search to keep searching for the characters from the beginning of the file, after the search reaches the end of the file (this works the other direction if you have Search Backwards checked).

   Check Case-Sensitive to search for characters that match the capitalization of what you typed in the Find panel. Otherwise, searching for “Cornichon” finds both “Cornichon” and “cornichon”.

4. Click Find.

   The first occurrence (after the insertion point) of the characters you’re searching for is selected in the window. If the characters aren’t found, the insertion point doesn’t move.

   You can choose Find Again from the Edit menu to search for the next occurrence of the characters.

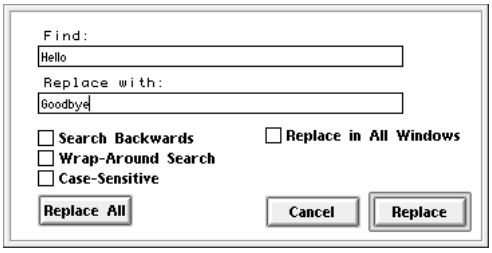
As a shortcut to using the Find command, you can select the characters you want to search for and choose Find Selection to find the next occurrence of those characters in the file.

Replacing Text

You use the Replace command to automate replacing one string of characters with another string.

1. Choose Replace from the Edit menu.

2. In the Replace panel, type the string of characters you want to replace and the string of characters you want to replace them with.



3. Check any Replace options.

   The Search Backwards, Wrap-Around Search, and Case-Sensitive options work the same as in the Find panel.

   Check Replace in All Windows if you want to replace the string in all open Edit documents when you click Replace All.

4. Click Replace to replace the next occurrence of the string. Or click Replace All to replace all occurrences of the string at once.

You can repeat the last replacement by choosing Replace Same from the Edit menu.

Showing Line Numbers

You can display a column that shows the number of each line in an Edit file by choosing Show Line Numbers from the window's Settings menu. Choose Show Line Numbers again to hide the line numbers.

Going to a Specific Line by Line Number

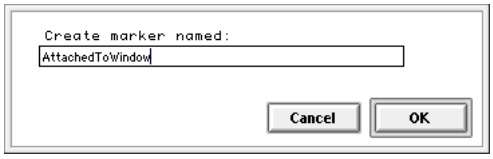
You can move the insertion point to the beginning of any line in an Edit file. Choose Go To Line from the Edit menu and in the panel that opens, type the line number you want to go to and click OK.

Creating and Using Markers

You can create and name markers in your Edit files that make it easy to jump directly to the location of the marker.

1. Select an insertion point where you want to create the marker. Or select text to use it as the marker name.
2. Choose Create Marker from the Markers menu.
3. Type a name for the marker.

   If you had text selected when you chose Create Marker, that text is inserted in the Create Marker panel. You can keep that text as the name of the marker or edit it.



4. Click OK.

   The marker (which is invisible) is inserted at the insertion point or beginning of the text selection, and the name of the marker is added to the Markers menu.

Once you've created a marker, you can go directly to its location by choosing the name of the marker from the Markers menu.

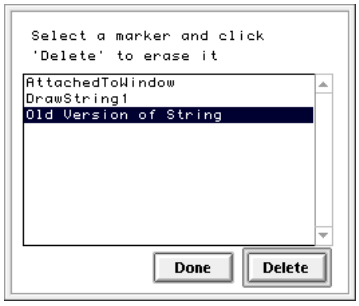
You can choose Alphabetize Markers from the Markers menu to list the names of makers alphabetically, rather than by their order in the file.

Deleting Markers

You can delete one or more markers when you're done with them.

1. Choose Delete Marker from the Markers menu.

   A panel opens with a list of the markers in the file.



2. Click a marker and click Delete.

   The name of the marker is removed from the list in the panel, and the marker is removed from the file.

3. Delete more markers if you want. Then click Done.

Creating a Script File

Script files are text files that take advantage of the Be shell, **bash**. You can write a script file in Edit and then save it as a script file by choosing the Convert to Script File command from Edit's File menu.

You run a script file by double-clicking it.

To edit an existing script file, drag its icon onto the Edit application (you can't double-click it to open it, because double-clicking it runs the script).

For more information about the Be shell, see "Terminal" on page 96.

Flight

Flight is a simple flying and shooting game. It's the only application currently bundled with the Be OS that shows off the two BeBox joystick ports.

When playing Flight, you fly across a barren landscape and shoot down unarmed helicopters (for 1 point) and other players (for 10 points), while trying to avoid crashing (losing 5 points), shooting down a team mate (granting the other team 10 points), or getting shot down yourself. Flight works best if you set the screen resolution to 800 × 600 or higher (see "Screen" on page 105). At lower resolutions, you can't see all of Flight's windows without overlapping. You can play Flight alone, with two players on the same BeBox, or as network game with up to eight players.

Preparing for Battle

When you start Flight, a darkened window opens: Dawn doesn't break until you start a game.

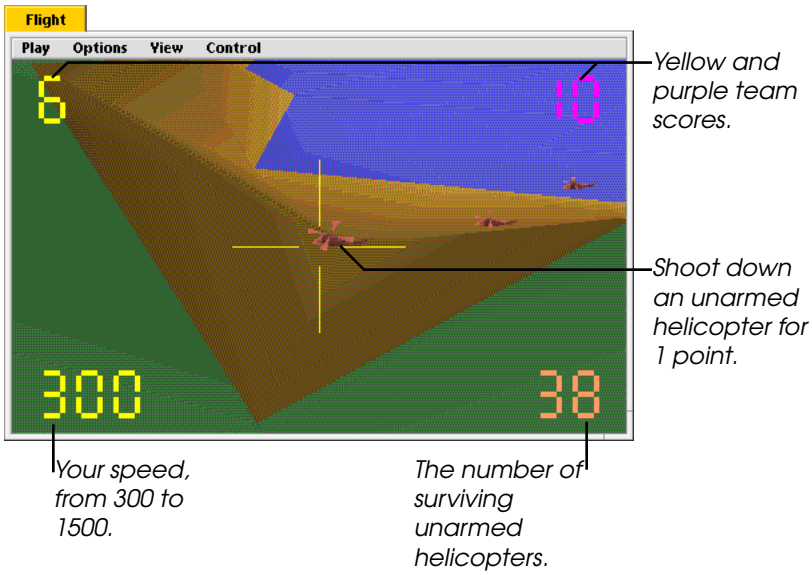
To set up a solo player (or a single entrant in a network game), choose the item from the Control menu that matches whether you'll control your flight using the keyboard, using a joystick connected to the upper joystick port on the back of the BeBox, or a joystick connected to the lower joystick port. Then ally yourself with the yellow or purple team by choosing the appropriate item from the Options menu.

To set up two players, choose Second Player from the Play menu and then set the control and team alliance for each player using the Control and Options menus in each window.

Engaging Battle

To start a game, choose Play Game from the first player's Play menu (only the first player can start and stop the game). To host a network game, choose Play NetGame. To join a network game hosted by another BeBox, choose Join NetGame.

When the game begins, Flight's main window provides a cockpit view of a plane flying through a landscape populated by unarmed helicopters. The crosshairs in the center of the window are for aiming your shooter.



If you're using a joystick, see the instructions that came with it to learn how to steer, change speed, and shoot.

If you're using the keyboard, use the arrows in the numeric keypad to steer, but think of a joystick as you use the up and down arrows: The up arrow points your nose down. Use the left and right arrow keys to the left of the numeric keypad to increase and decrease your speed. Shoot by pressing the right Shift key.

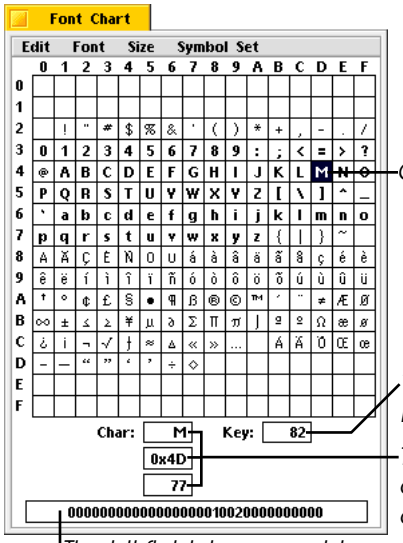
Each player can choose Map from the View menu to open a window that shows his position in the landscape. Yellow and purple team players are represented by crosshairs. Unarmed helicopters are represented by black dots. The other commands in the View menu control how wide a view is displayed in the window.

If you find it hard to fly, you can choose Crash on Ground from the Options menu at any time: When it's unchecked, you plow along the surface instead of crashing.

A game of Flight can't be said to have a real conclusion: Each player has an unlimited supply of ammunition and is reincarnated endlessly. However, the landscape does seem bleaker when all the unarmed helicopters are gone.

FontChart

FontChart opens a window with a chart of the characters in a font.



Click or type a character.

The Key field shows the raw keyboard key number in decimal.

These fields display the character you type (or click in the chart) in the current font, plus the character's code in hexadecimal and decimal.

The bit field shows raw binary data from the keyboard as you press and release keys

You can use the secondary mouse button to drag a character or the text from any of the fields in FontChart to another application. For example, you can drag a character from the FontChart window onto a key cap in the Keyboard application, or into an Edit document.

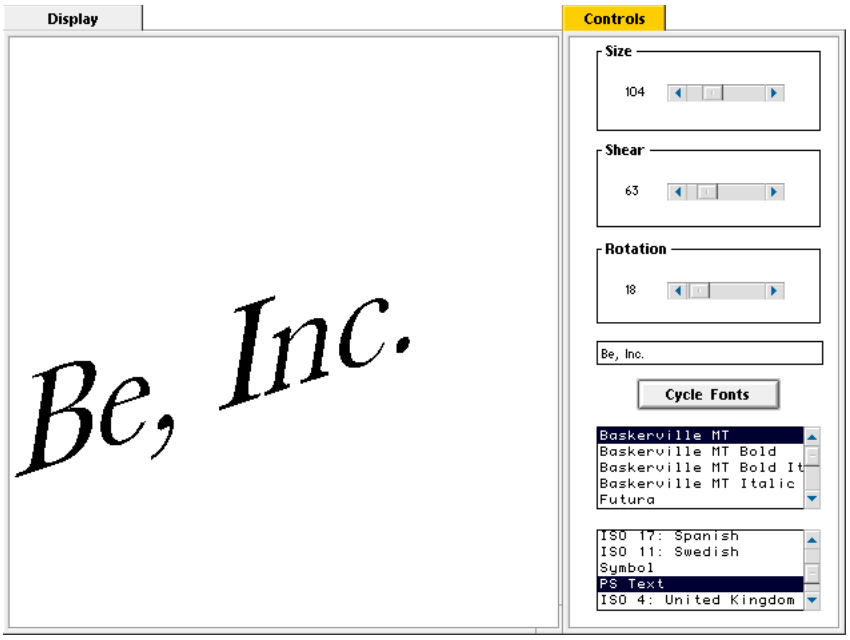
At the bottom of the window, the bit field shows the raw keyboard key number, in binary form, as you press and release keys singly or in combination. This information is useful for game developers and others who rely on raw keyboard data.

You can choose the Keyboard ID command from FontChart's Edit menu to see the ID number of the keyboard connected to the BeBox.

The prerelease version of the Be software uses the same symbol set as the Macintosh operating system. For more information on symbol sets, see “Keyboard” on page 86.

FontDemo

FontDemo shows off the BeBox’s speed and versatility with fonts. When you start up FontDemo, two windows open: A window that displays some sample text and a Controls window.



Use the Size scroll bar to change the size of the font in the Display window. Use the Shear scroll bar to shear (or slant) the text. Use the Rotation scroll bar to rotate the text.

Type the text you want to display in the field above the Cycle Fonts button.

Select the font your want to use for the text in the Display window in the upper scrolling list. Select a symbol set from the lower scrolling list.

Click Cycle Fonts to have FontDemo select the available fonts in order. You can use the other controls while FontDemo cycles through the fonts.

HeapWatch

HeapWatch, the application for inspecting an application's main memory heap, is on vacation in release 1.1d7. It's due back in time for the next release of the Be OS. For an alternative method of verify an application's heap, see Chapter 4, "Developing Be Applications."

You can verify an application's heap by running it from a Terminal window after setting the MALLOC_DEBUG environmental variable: Type **export MALLOC_DEBUG=true** in a Terminal window and then type the application's path name and press Enter. If the application fails with a memory error, information about the error is displayed in the Terminal window.

HelloWorld

HelloWorld is the Be version of the standard "Hello, World!" application, familiar to programming students everywhere. A compiled version of HelloWorld is no longer included in the **/apps** folder, but the sources, a BeIDE project file, and a makefile are included in the **/develop/projects/HelloWorld** folder in the Be OS. For more information, see Chapter 4, "Developing Be Applications."

IconWorld

IconWorld is an application for editing the icons of applications and their associated files. You also use it to assign application resources. Using IconWorld is explained in "Creating Resources for a Be Application" on page 138.

ImageViewer

You can use ImageViewer to open and view two kinds of image files: Color TIFF (Tagged Image File Format) files and files in the image file format used by the Be OS. The custom Be image file format is the file format of "screen dumps," the images of the screen you can create by pressing the Print Screen key. ImageViewer is also useful for converting image files to a format that's compatible with Adobe Photoshop on a Macintosh, and for showing off the Be drag-and-drop metaphor.

For information about creating screen dumps and ImageViewer's role in the process, see "Creating Screen Dumps" on page 113.

To show off the Be drag-and-drop metaphor, open an image file (a screen dump or one of the files in the **/optional/images** folder on the *Be OS* CD-ROM) and select an area you like by dragging across it. Then drag the selected area to a Browser window: A new image file is created, called "clip." You can double-click the new file to see that it contains just the area you selected.

Installer

The Installer application has two purposes: Preparing disks so you can use them with the BeBox and installing the Be OS. Installing the Be OS is described in *Installing or Upgrading to Release 1.1d7 of the Be Software*, the booklet that accompanies this user's guide.

Initializing or Formatting a Disk

You can use the Installer to prepare IDE, SCSI, and floppy disks for use with the BeBox by initializing them or—if necessary—by formatting them.

Note: The current release of the Be OS only supports high-density floppy disks. (These are usually labeled "HD".)

Initializing a disk prepares it to work with the Be file system. You must initialize a disk before you can mount it the first time and store Be files on it. It only takes a few seconds to initialize most disks.

You may need to format a disk if it has never been set up to work with a computer or if it has been damaged. Most disks come preformatted, so it's rare that you'll ever need to format one. It can take a very long time to format a disk: The better part of an hour for large disks. After the Installer formats a disk, it also initializes it.

If you're not sure whether you need to format a disk, try initializing it first. The Installer inspects the disk and lets you know if you need to format it.

Caution: Both initializing and formatting a disk erase everything on the disk (in fact, initializing a disk is a good way to erase it quickly), so make sure you've backed up any files you want to keep before you initialize or format a disk.

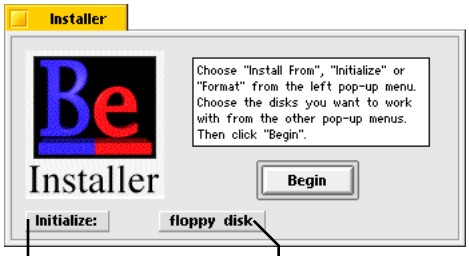
1. If you're initializing or formatting a SCSI or IDE disk, make sure it's connected to the BeBox.

   You can connect up to two IDE disk drives to the BeBox and up to seven SCSI disk drives. Depending on the number of disks you connect to the BeBox and how they're connected, you may need to configure each IDE disk as a "single," "master," or "slave" disk or configure each SCSI disk with a unique SCSI ID and terminate it correctly. For more information, see the owner's guides that came with the disks and Chapter 1, "Assembling the BeBox."

   Uninitialized disks don't appear in the Browser.

2 Double-click the Installer application.

3 Choose Initialize or Format from the left pop-up menu. Then choose the disk you want to initialize or format from the right pop-up menu.



Choose Initialize (or Format).

Then choose the disk you want to initialize (or format) and click Begin.

If a disk was previously initialized on a BeBox, it appears by name in the right pop-up menu. If a disk hasn't been initialized, it appears in the pop-up menu by its device name:

Device Name	Description
ide - master	The single IDE disk connected to the BeBox, or the master IDE disk if there are two connected.
ide - slave	The slave IDE disk, if there are two IDE disks connected to the BeBox.
scsi - id *n*	The SCSI drive set to SCSI ID number *n* (*n* can be from 0 through 6)
floppy disk	The disk in the floppy disk drive.

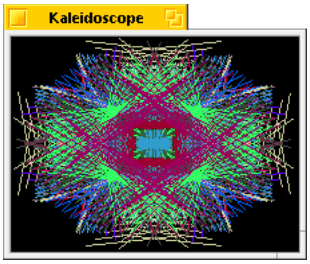
4 Click Begin. Then click Initialize (or Format) in the panel that warns you that initializing (or formatting) the disk will erase it.

When the disk is initialized, a message informs you that it's now available in the BeBox window. Floppy disks are named "fd" when you format them. Other disks are named "NewDisk."

You can now initialize another disk, install the Be OS, or quit the Installer.

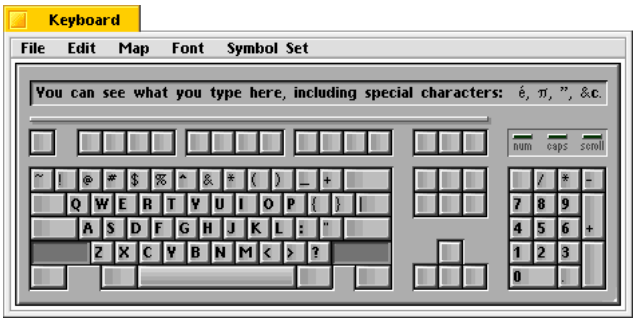
Kaleidoscope

Kaleidoscope is a simple line-drawing application: It opens a window where it draws patterns of colored lines as fast as it can. It's interesting to see the effect of multiple Kaleidoscope windows on the BeBox's performance.



Keyboard

The Keyboard application opens a window that shows you the current keymap, that is, the character you see on screen when you press each key. You can type on the keyboard or click keys in the Keyboard window to see the resulting character in the field at the top of the Keyboard window. If you press one or more modifier keys (such as Shift, Option, or Control) on the keyboard or click them in the Keyboard window, the keycaps change to show you what characters are mapped to the keys when the modifier keys are pressed.



Different fonts map characters to different keys. You can choose a font from the Font menu to see that font's mapping in the Keyboard window. This is also a useful way to look for a character you don't use very often, such as bullets, accents, or other special characters.

The prerelease version of the Be OS comes with two preset keymaps: "Qwerty," the standard keymap in the U.S.; and Dvorak, a popular alternative keymap. You can switch between those two keymaps by choosing one from the Map menu.

You can create a custom keymap by using the secondary mouse button to drag characters to the keys you want to map them to. You can also use the secondary mouse button to drag characters from other applications, such as FontChart.

Each keymap is made up of nine mapping tables: A table of what each key produces when you type it with no modifier key pressed, or with the Shift, Control, or Option key pressed, plus all the possible combinations of these keys.

If you use the tertiary mouse button to drag a character from one keycap to another, you actually drag all 9 characters associated with that key in the nine tables in the keymap.

You can save a keymap by choosing Save or Save As from the File menu. To use a keymap you've saved, drag the keymap file into the Keyboard window.

You can create a special keymap file, **/system/settings/key_map**, which is used as the default keymap each time you turn on the BeBox. Choose Save As System from Keyboard's File menu to save the keymap in the Keyboard window as that file.

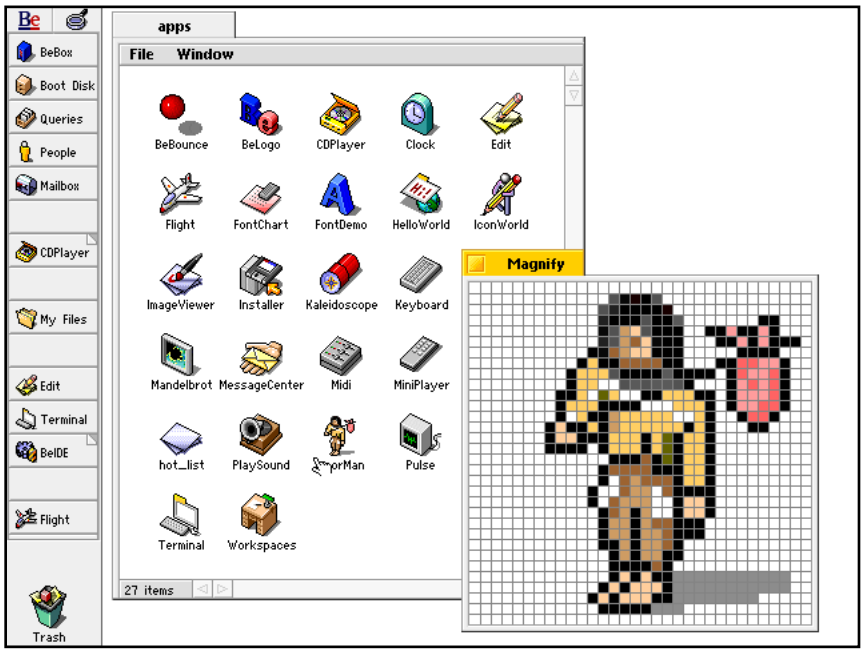
You can revert to the default keymap at any time by choosing System Startup from Keyboard's Map menu. If you delete the **/system/settings/key_map** file, you revert to the BeBox's idea of a default keymap the next time you restart the BeBox.

The "Interface Kit" chapter of *The Be Book*, the reference to the Be API, includes information about keymaps of interest to programmers.

You can choose a symbol set from the Symbol Set menu. Symbol sets are conventions for what characters are used in fonts and keymaps. Different computers and countries use different symbol sets. If you're having trouble reading a document that was created on another kind of computer, try switching to the symbol set used by that computer.

Magnify

The Magnify application displays a greatly enlarged image of the area of the screen next to the cursor. This is useful for inspecting the pixel-by-pixel details of icons and other objects on the screen.

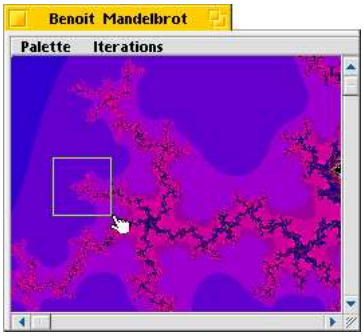


When you double-click Magnify, it opens a window that displays an area 32 pixels high by 32 pixels wide. If you want a different-sized window, start Magnify by typing Magnify's path name in a Terminal window, followed by a space and the number of pixels per side you want for Magnify's window. For example, type **/apps/Magnify 100** in a Terminal window to open a 100-pixel by 100-pixel window. The number you type must be a multiple of four.

Initially, Magnify uses a grid pattern to separate the pixels in its window. You can choose Show Grid from Magnify's main menu to hide the grid.

Mandelbrot

The Mandelbrot application demonstrates the power of the BeBox by drawing images based on the "Mandelbrot set," a set of mathematical formulas invented by Benoît Mandelbrot, the Polish mathematician. Mandelbrot takes advantage of the two PowerPC microprocessors in the BeBox by using one CPU to compute the odd lines in the image and the other to compute the even lines.



It's interesting to zoom in on areas of the image in the window. Drag across an area you like to recalculate the image so the area you dragged across fills the window. You can do this indefinitely. To zoom out, hold down the Shift key and click in the window.

You can choose a color palette from the Palette menu. You can choose the level of iteration you want from the Iterations menu. The higher the number you pick, the more precise—and time-consuming—are the calculations used to draw the image. Higher numbers are more important as you zoom further into an area of the image.

MessageCenter

You double-click MessageCenter to put mail in the Mailbox. The Mailbox, the icon in the dock, is a database entity that demonstrates the power of the Be database.

You can arrange to "get new mail" by dragging all the mail in the Mailbox window to the Trash, dragging Be movie, Be raw sound, and up to three text files into the **/system/msgs** folder, and then restarting MessageCenter. When you drag these kinds of files into the **/system/msgs** folder, Be movies must end in **.movie**, sounds must end in **.sound**, and the three text files must be named **1.text**, **2.text**, and **3.text**. A number of larger movies and sounds are included in the **/optional/movies** and **/optional/sounds** folders on the *Be OS* CD-ROM.

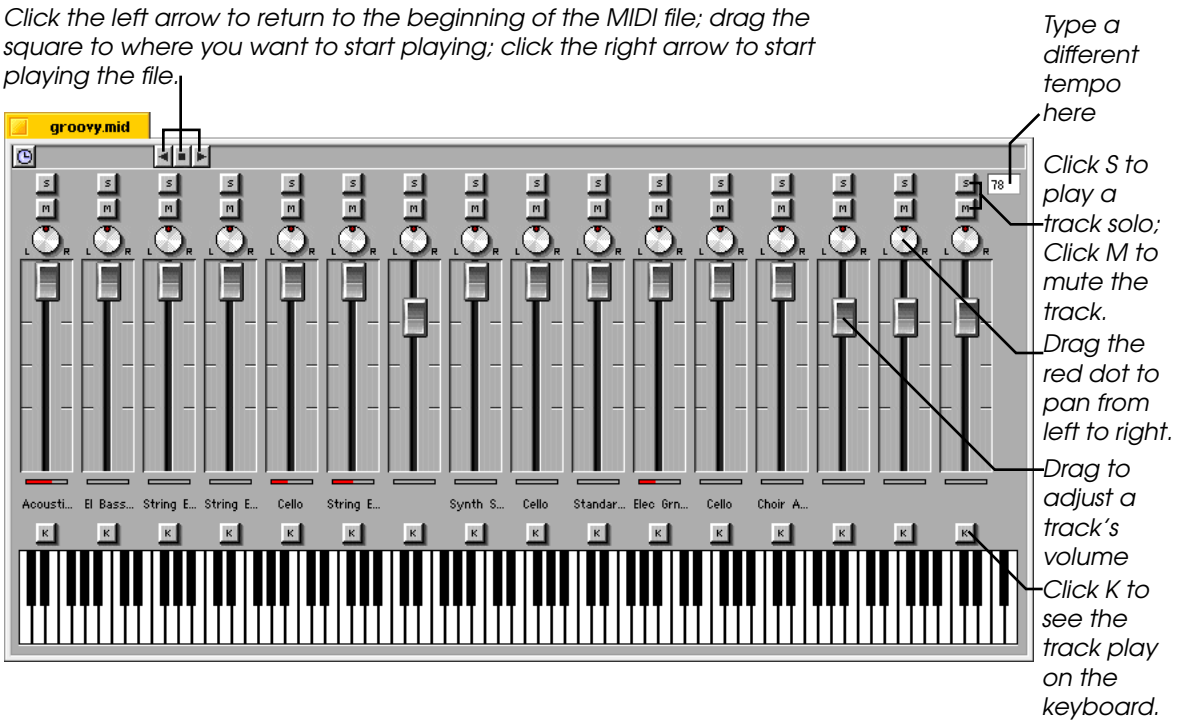
Midi

Midi is an application for playing MIDI (Musical Instrument Digital Interface) files on a MIDI synthesizer. (If you don't have a MIDI synthesizer connected to the MIDI-1 Out port on the BeBox, MIDI will play MIDI files, but you won't be able to hear them.) You open a MIDI file by dragging it onto the Midi application icon, which opens the MIDI window.

A number of MIDI files are included in the **/optional/midi** folder on the *Be OS* CD-ROM.

The full MIDI window is only visible if you're using a screen grid that's 800 pixels wide by 600 pixels tall. You set the screen grid with the Screen preferences application, as described in "Screen" on page 105.

Note: When you first copy a MIDI file onto the BeBox, it doesn't have a MIDI file icon and isn't associated with the Midi application until the first time you drag the file to the Midi application.



Each track in the MIDI file is represented by a vertical set of controls. You can play a track as a solo, mute the track, pan the track between the left and right audio channels, adjust its volume, and see it play on the keyboard, all by adjusting the controls for that track.

You can synchronize the beginning and end of the MIDI track with other application's files by dragging the clock icon to or from the Midi application's scroll bar.

MiniPlayer

MiniPlayer is a basic audio CD player—the antidote to CDPlayer's bells and whistles. If you just want to listen to a CD, MiniPlayer can handle it without taking up a lot of screen real estate.

MiniPlayer's controls are a subset of those in the CDPlayer application. For information about MiniPlayer's controls and menu items, see "CDPlayer" on page 66.

Orb

Orb is a very simple world-wide web browser, created as an experiment in only a few days. If you're connected to the Internet, you can use Orb to surf the web—but think of Orb as one of those long, wooden surfboards you'd see hanging over a bar on The Big Island. Orb understands only the most basic HTML formats (HTML, the Hyper Text Markup Language, is the formatting language used for world-wide web documents, or "pages".) For more information on HTML, point Orb at **http://www.w3.org/hypertext/WWW**. Orb tries to ignore HTML it doesn't understand, such as forms. Orb can display image files in GIF format, including image maps, but it can't display JPEG images.

The Hot List menu initially contains two URLs (Universal Resource Locators, the name for addresses of pages on the world-wide web), for the Be and Metrowerks home pages. You can add URLs to this menu by opening the **/apps/hot_list** file in Edit and adding one URL per line. When you start Orb, it tries to connect to the first URL in the **hot_list** file.

If you have an HTML file on disk, you can open it in Orb by typing **file:*/pathname*** (where *pathname* is the path of the file you want to open) in the field at the top of the Orb window. You can also open a local file with the Open command in Orb's File menu.

You can use the PoorMan application to host a web site on a BeBox, as described in "PoorMan" on page 94.

PlaySound

You can listen to an audio file by dragging it onto the PlaySound icon. PlaySound can play audio files in "raw," AIFF (Audio Interchange File Format), and WAVE (Waveform Audio File Format) format. Depending on the Sound preferences application settings and the devices connected to the BeBox, you hear the file on the internal BeBox speaker or on a device connected to the line-out ports. For more information, see "Sound" on page 109.

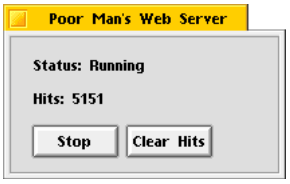
A number of sound files are included in the **/optional/sounds** folder on the *Be OS* CD-ROM.

When you double-click the PlaySound icon, it plays a "beep" on the speaker or whatever device you have turned up in the Sound application.

PoorMan

PoorMan is a simple but effective world-wide web server. A web server is an application that makes a web site available to a network: Either to a company network or, if the network is configured to allow it, to the whole Internet. PoorMan can host a web site made up of files in HTML, JPEG, and GIF formats. HTML (the Hypertext Formatting Language) is the formatting language used for world-wide documents, or "pages." JPEG and GIF are graphics file formats commonly used for illustrations in world-wide web pages.

When you start PoorMan, a window opens with a status area and two buttons. When you click Start, PoorMan looks for a "home page" (the starting point for a web site) named **index.html** in a folder named **/public_html** on the boot disk. If they don't exist, PoorMan creates them. The **index.html** file PoorMan creates is a very simple web page in HTML format.



When PoorMan is running, others on your network can use their web browsers to view the pages on your site. Depending on how your network is set up, they may be able to

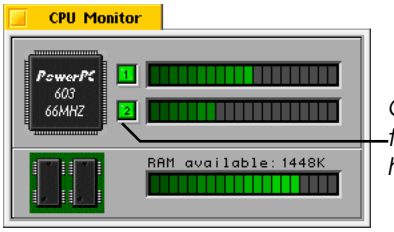
simply type your BeBox's hostname or IP address to open your site's home page (the **/public_html/index.html** file).

Each time someone browses a page or image on your web site, the "hit" is recorded in the total in PoorMan's window. Click "Clear Hits" to return the count to zero.

Unfortunately, you can't use Orb to browse the pages PoorMan is serving on the same BeBox (though you can use Orb to browse pages PoorMan is serving on other BeBoxes on the network). For more information about Orb, see "Orb" on page 93.

Pulse

The Pulse application shows you the relative load on the BeBox's two PowerPCs—an on-screen version of the information displayed by the two LED arrays on the front of the BeBox. Pulse also shows you how much of the BeBox RAM is in use.



Click a numbered button to "turn off" one of the CPUs. If you click both buttons, you'll have to restart the BeBox!

ROMUpdater

Part of the Be OS system software is actually on a programmable "flash ROM" chip on the BeBox motherboard. If Be sends you a file with updated software for the flash ROM, you update the ROM chip by dragging the update file onto the ROMUpdater application.

Nothing happens if you simply double-click ROMUpdater.

Terminal

The Terminal application opens a window where you can work with the Be shell. The Be shell is a UNIX-style, command-line environment based on **bash** (the Bourne-Again Shell, created by Brian Fox and Chet Ramey).

Most of the programs you work with in a Terminal window are in the **/bin** folder.

You can open as many Terminal windows as you want: Just keep double-clicking the Terminal icon or choose Start New from a Terminal window's Terminal menu. The Go to Next command in the Terminal menu makes another open Terminal window the active window, so you can cycle through them quickly from the keyboard by pressing Command-g.

You can use the commands in the Settings menu to customize how Terminal displays the text in its window. Choose traditional dimensions (in characters) for the current window from the Window Size menu. Choose a font and font size Terminal uses to display the text in its window from the Font submenus, but note that only monospaced fonts are included in the menu (Terminal relies on the constant character widths of monospaced fonts to align text and to display other text effects). Also note that Kate, the font Terminal uses initially, is only available in one size. Set the number of space characters Terminal substitutes for Tab characters by choosing a number from the Tab Width submenu.

You can use the keyboard to scroll the contents of the Terminal window:

Press	To scroll
Shift–up arrow	Up one line
Shift–down arrow	Down one line
Shift–page up	Up one windowful
Shift–page down	Down one windowful

The default shell environment is set by the **/system/profile** file. You can modify the environment by making a copy of that file, changing it, and saving it in the root folder of the boot disk with the name, **.profile**.

Learning About bash and the Programs in the /bin Folder

Explaining how to work on a UNIX-style command-line is beyond the scope of this user's guide. In fact, there are lots of good books on the subject available in any book store with a well-stocked section on computers.

A copy of the man pages for **bash** and for a great many of the programs in **/bin** are included in HTML format in the **/documentation/Shell Tools/HTML** folder. Man pages are the traditional, descriptive (if terse) documentation for UNIX-style programs. HTML is the formatting language used for pages on the world-wide web. You can read the man pages by transferring the **HTML** folder to a computer with a web browser application, such as Mosaic or Netscape on a Macintosh. (For help transferring files, see "Transferring Files to and from the BeBox" on page 115.) Start by opening the **index.html** file in a web browser. It contains an alphabetical list of the man pages, with "links" you can click to open the man page you want to read. Because the man pages make heavy use of the **<pre>** HTML code, you may also find them readable in a text editor, such as Be Edit.

When you're working in Terminal, you can also get a small amount of information about many of the programs in **/bin** (often enough to get by) by entering the name of the program followed by the **--help** flag (that's two hyphens followed by "help"). Enter **help** in Terminal to read about **bash** itself.

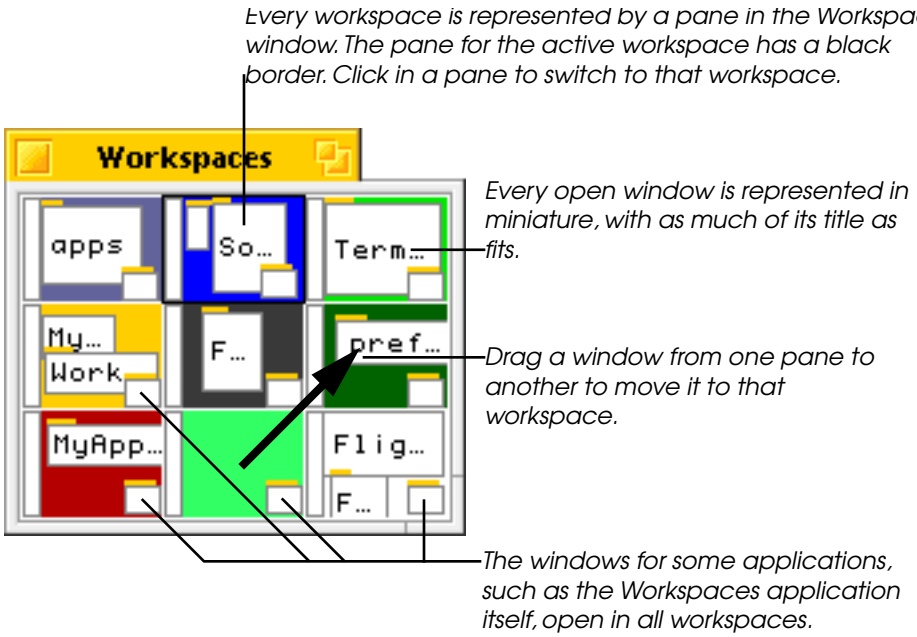
Workspaces

When you first start working with the BeBox, you have one workspace: A desktop and a set of windows that open on it, which you arrange to suit your projects. With the Workspaces application, you can arrange up to 32 such workspaces.

When you first start Workspaces, a window with nine panes opens. Each pane represents a workspace, with a miniature version of the dock and each window open in that workspace. Click in a pane to switch to that workspace. You can also switch workspaces by holding down the Command key while you press the function key that corresponds to that workspace (counting from left to right, top to bottom in the Workspaces window): Command-F1 to switch to the first workspace, Command-F2 to switch to the second, and so on.

The workspace you're currently working in is called the *active workspace*. When you open a window it opens in the active workspace. Many windows can be open in only one

workspace at a time. This is true for Edit document windows and the windows that open when you start most of the preferences applications. If a window is already open in a workspace and you open it again, you switch to the workspace where the window is already open. Other applications and their windows behave differently. For example, when you start the Clock application, the Clock window opens in all workspaces (this is also true for the Workspaces application itself). A third model is offered by folder windows in the Browser: They open in the active workspace when you double-click their folder icons, even if they're already open in another workspace.



If a window is open in more than one workspace, you can move it to a different location in each workspace. However, changing the size of a window, minimizing it, or closing it in one workspace does the same to all copies of the window in all workspaces.

You can drag an application's icon into a pane in the Workspaces window to start it in that workspace.

You can use the Screen application to give each workspace a different desktop color—this can help keep them distinct. The resolution (number of pixels on the screen) and color depth (bits per pixel) also only apply to the current workspace. See "Screen" on page 105 for more information.

If you want to work in more or fewer workspaces, choose the number you want from the Workspaces submenu in the Workspaces applications' main menu. If you increase or decrease the number of workspaces, panes are added or removed from the Workspaces window in order, starting from the bottom right. You can't reduce the number of workspaces if it would remove a workspace that has open windows: Close any open windows first. Avoid the higher prime numbers of workspaces—the Workspaces window can't lay them out very legibly.

Using the Preferences Applications

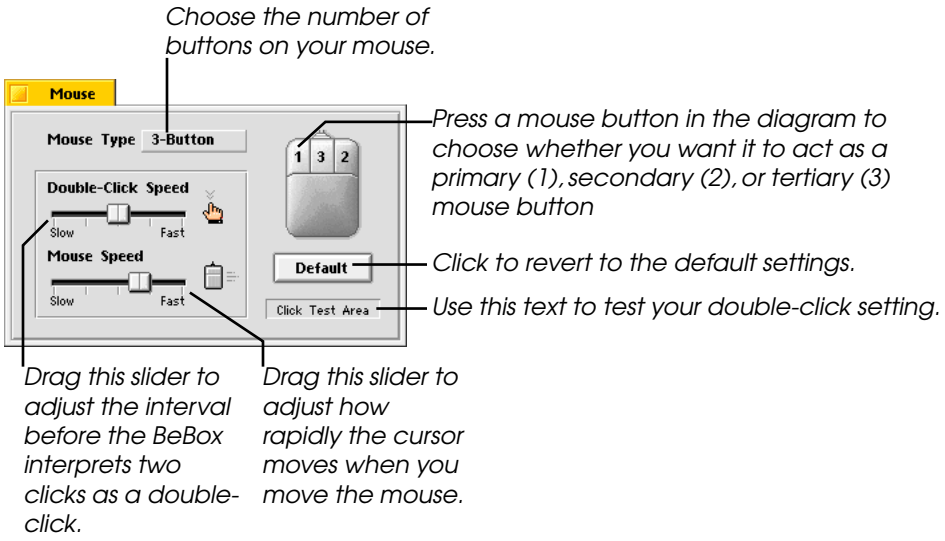
You can use the applications in the **/preferences** folder to configure many aspects of the BeBox to your liking. The preferences applications store your preferences in files in the **/system/settings** folder. You can revert to default preferences settings by deleting the appropriate settings file from the **settings** folder (some applications provide a way to revert to defaults using their commands or buttons).

Mouse

You can use the Mouse application to set your preference for a number of ways a mouse works with the BeBox. You can set whether you're using a one-, two-, or three-button mouse and which buttons you want to use as the primary, secondary, and tertiary mouse buttons (for more information on the role of the mouse and mouse buttons in the Be OS, see "Using the Mouse" on page 23). You can set how quickly you have to click the mouse button before the BeBox interprets two clicks as a double-click. You can also set how rapidly the cursor moves across the screen when you move the mouse.

Mouse stores your settings in the **Mouse_data** file in the **/system/settings** folder.

When you start up the Mouse application, the Mouse window opens.



The Mouse application stores your preferences in the **/system/settings/mouse_data** file.

Network

You use the Network application to configure the BeBox to communicate with other computers. You can configure the BeBox to communicate across an Ethernet network connected to an Ethernet network adapter in the BeBox. You can also configure the BeBox to communicate using PPP (the Point-to-Point Protocol) through one of its serial ports, via a modem or via a direct cable connection to the serial port on another computer.

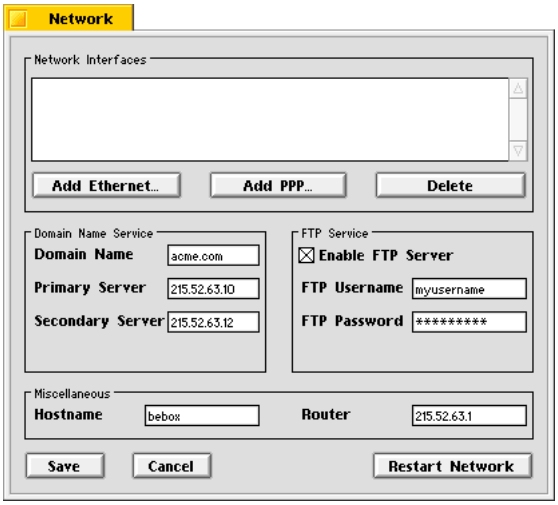
You can configure multiple network interfaces—for example, if you have more than one network adapter installed in the BeBox. However, it's up to the software you're using to know which interface to use.

Important: Before you start, be sure to let your network administrator or Internet service provider know you're connecting a new computer the network. Your network administrator or service provider will also provide you with information you'll need to enter in Network.

Configuring Network Basics

1. Start up the **Network** application.

2. Enter information about yourself, the BeBox, and the network in the Network window. You may need to ask your network administrator for some of the information requested by the Network application.



   - In the Domain Name field, type the name of your network domain.
   - In the Primary Nameserver field, type the IP address of your network's primary DNS server, if it has one.

     If your network doesn't have a DNS server, or if you or your network administrator decide you don't want to use it, you can create a copy of **/system/hosts-sample**, name it **/system/hosts**, and then edit the copy in the Edit application so it lists the names and IP addresses of the computers you want to connect to. Your network administrator may have a standard **hosts** file for all the machines on your network.
   - In the Secondary Nameserver field, type the IP address of your network's secondary (or backup) DNS server, if it has one.
   - In the Hostname field, type a name for your BeBox (most network administrators let you pick this name).

- If you want to allow users on other computers to use **ftp** to put and get files from your BeBox, check the Enable FTP Server option. Then in the FTP Username field, type the name you use when logging into other computers on your network. Keep in mind that entering a user name has two effects: First, it makes logging into other computers using **ftp** more convenient by presenting your user name as the default at the login prompt. Second, it makes it possible to run **ftp** on another computer and connect to the BeBox, using your user name to log in. However, in spite of the password field in the Network window, there is no password protection on the BeBox, so anyone who knows your user name can log into the BeBox. If this is a problem, don't check Enable FTP Server.

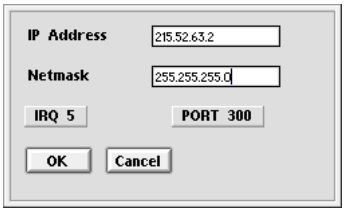
- In the Router field, type the name of the "smart" router on your network, if it has one.

Adding an Ethernet Network Interface

If you added an Ethernet network adapter in the BeBox (as described in"Adding a Network Adapter" on page 8), you need to use the Network application to configure the Be OS to use it.

1. Click Add Ethernet.

   A panel opens where you enter information about an Ethernet network adapter installed in the BeBox and about the network it's connected to. You may need to check with your network administrator for this information.



2. In the IP Address field, type the IP address of the network adapter card in your BeBox.

   This number uniquely identifies the BeBox computer on the network.

3. In the IP Netmask field, type the netmask for your class of network, if required.

4. Choose the IRQ and port that match the configuration of the network adapter.
5. Click OK.

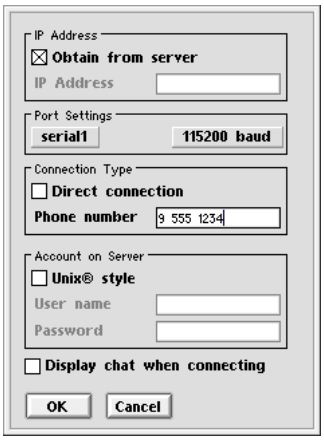
   The new network interface is added to the scrolling list at the top of the Network window.

Adding a PPP Network Interface

You can configure the BeBox to communicate with a network using PPP. PPP is the protocol many Internet service providers offer to subscribers for connecting to the Internet. If you want to communicate directly with a computer that can host a PPP connection, you can also configure the BeBox to communicate with that computer though their serial ports, using a cable called a "null modem cable."

If you're setting up the BeBox to work with an Internet service provider, review these instructions and check with your service provider to determine the correct settings.

1. Click Add PPP.



2. If your service provider establishes your IP address each time you connect (this is sometimes called "dynamic addressing"), make sure "Obtain from server" is checked. Otherwise, uncheck this option and type the IP address assigned to you (this is sometimes called "static addressing").
3. Choose the serial port the modem or null modem cable is connected to.

4. Choose the speed at which you want to communicate through the serial port.

5. If you're connecting the BeBox directly to another computer via a null modem cable connected to the two computers' serial ports, check "Direct connection." If you're using a modem to call a service provider, uncheck this option and type the phone number of the service provider.

   Type the number exactly as you would dial it: If you normally have to dial a number to get an outside line, type that number first, and don't forget to include a **1** and an area code if it's a long distance call. For more information, see the modem's owner's guide.

6. If your service provider uses PAP (the Password Authentication Protocol), leave the "UNIX style" option unchecked. If your service provider uses a more traditional "UNIX style" method to log you in, check that option and enter your user name and password.

7. Check the "Display chat when connecting" option if you want to observe the messages the BeBox and the host computer exchange as a connection is established.

8. Click OK.

   The new PPP interface is added to the list at the top of the Network window.

Saving Network Settings and Restarting the Network Software

Once you've configured the settings in the Network window and added the network interfaces you need—or if you make any changes—click the Save button. Your changes are saved in the network settings file (named **network** file in the **/system/settings** folder) and a panel asks if you want to restart the network. Click OK to restart the BeBox's networking software so the new settings take effect right away (otherwise, the settings don't take effect until the next time you restart the BeBox).

Screen

You use the Screen application to set how the BeBox, display adapter card, and display interact to display the Be OS. You also use Screen to set the color of the desktop.

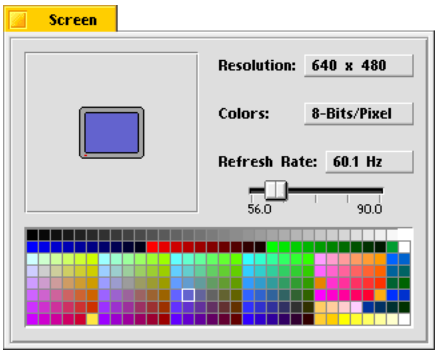
Setting the Screen Resolution, Colors, and Refresh Rate

Initially, the BeBox works with the display adapter and display to present a screen resolution of 640 pixels horizontally and 480 pixels vertically, using 8 bits of color data per pixel (sometimes referred to as the "screen depth"), with a refresh rate (the number of times per second the screen is redrawn) of 60.1 Hz. If you have a display adapter and display that can accommodate a higher resolution, more colors per pixel, or a faster refresh rate, you can change these settings with the Screen preferences application.

For information about what kinds of display adapters work with the BeBox and release 1.1d7 of the Be OS, see "Adding a Display Adapter" on page 7.

Warning: You risk damaging the display and even starting a fire if you select a resolution, number of colors per pixel, or refresh rate the display isn't designed to support. Read the owner's guides that come the display and display adapter to find out what combinations of settings are safe for the display.

When you start up the Screen application, the Screen window opens.



You change a setting by choosing from the pop-up lists, or by dragging the Refresh Rate slider, and then clicking OK in the panel that asks you to confirm the change. You can also adjust the refresh rate by pressing the left or right arrow key on the keyboard—this technique is useful to fine-tune the best-looking refresh rate.

If your display adapter or display can't support a setting you make, you may not be able to read anything on the screen—but if you wait a few seconds, the screen reverts to the previous setting if you don't click OK.

You can choose Confirm Changes from Screen's main menu to uncheck it in the menu, so you don't have to click OK to confirm each setting change. If you do this and then change a setting that makes the screen unreadable, you can use keyboard shortcuts for commands in Screen's main menu to restore the default settings: Command-d for Default Resolution and Command-r for Default Refresh Rate.

The changes you make to the resolution and number of colors only affect the active workspace, so if you're working with more than one workspace (as described in "Workspaces" on page 97), you can set a different resolution and number of colors for each workspace.

Adjusting the Screen Size and Position

When Screen is the active application, you can adjust the size of the screen image on the display by holding down the Shift key while you press the up or down arrow key. You can adjust the position of the screen image by holding down the Control key (the key labeled "Ctrl" in the lower-left corner of most PC keyboards) and pressing the up, down, left, or right arrow key. To restore the default screen size and position, choose Default Position from Screen's main menu.

Setting a Desktop Color

The desktop—the "background" of the screen—is initially solid blue, but you can pick a different color with the Screen application.

If you choose 8 bits of color information per pixel, the lower portion of the Screens window shows the 256 colors you can choose from for a desktop color (at 8 bits per pixel, you can display up to 2^8 (that is, 256) colors on the screen at one time). Click the color you want.

If you choose 32 bits of color information per pixel, the lower portion of the Screens widow offers three sliders, for the red, green, and blue components of the over 16 million colors you can display on the screen at one time (only 24 of the 32 bits are used

for color information, and 2^{24} equals 16,777,216). Drag each slider until you find a combination that produces the color you want.

If you're using the Workspaces application to work in more than one workspace (as described in "Workspaces" on page 97), the color you pick only affects the active workspace—the one where you're using Screen—so you can give each workspace a distinct desktop color.

Displaying an Image on the Desktop

You can display an image file on the desktop, instead of a solid color.

1. Use an application on another platform (such as Photoshop on a Macintosh) to save an image file in OS/2 BMP format.

   You can use any size image as a desktop image, as long as the width of the image is divisible by 4. However, if you use an image that's not the same size as the resolution you use on the BeBox—such as 640 × 480, 800 × 600, or 1024 × 768 pixels—the image will be cropped if it's larger, or repeated if it's smaller than the screen.

2. Use **ftp** or **tar** to place a copy of the image file on the BeBox.

   For information on **ftp** and **tar**, see "Transferring Files to and from the BeBox" on page 115.

3. Name the image file **DesktopImage** and place it in the **/system** folder.

4. Restart the BeBox.

Unlike desktop colors, which can be different in each workspace, the same desktop image appears in all workspaces.

Caution: Displaying an image file on the desktop takes up a lot of RAM: 1.6 MB for a 24-bit image. So if memory is tight, use a solid color instead.

Screen stores your settings in the **Screen_settings** file in the **/system/settings** folder.

ScrollBar

Initially, many windows in the Be OS use “proportional scroll bars,” as described in “Scrolling the Items in a Window” on page 29.

If you don’t like proportional scroll bars, you can use the ScrollBar application to set your preference to a fixed-size scroll knob and make some other choices about how scroll bars look and behave.

Click the arrow style, knob style, and knob type you prefer. Drag the green arrow to adjust the minimum scroll knob size. Click Defaults to return all the settings to the way they were when you first installed the Be OS. The settings you make in ScrollBar affect windows when you next open them.

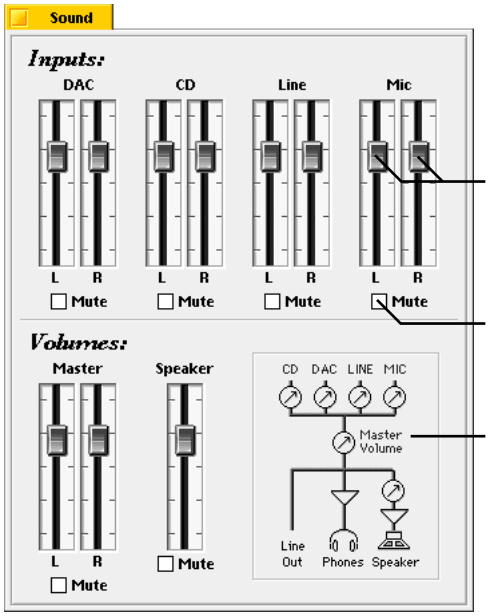
ScrollBar stores your settings in the **ScrollBar_settings** and **ScrollBar_data** files in the **/system/settings** folder.

Sound

You can use the Sound application to adjust the input and output volumes of a number of audio features on the BeBox.

Sound stores your settings in the **Sound_data** file in the **/system/settings** folder.

When you start up the Sound application, the Sound window opens.



Drag a slider up or down to increase or decrease the volume of a left or right channel.

Click a mute box to mute that feature.

The diagram shows the paths of the BeBox inputs and outputs (except that the microphone passes through the DAC).

Drag a slider to adjust the volume of an input or output feature, or check a Mute box to turn it off.

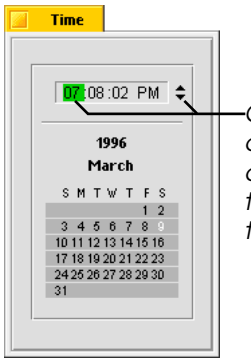
The DAC (Digital-Analog Converter) slider controls the circuitry the BeBox uses to convert analog data to digital data, and vice versa. PlaySound, the GeekPort, and the Microphone port all use the DAC.

The interaction between the sliders is diagrammed in the window. However, the diagram is inaccurate in one particular: The Microphone port signal actually passes through the DAC.

The Master volume slider controls all audio output from the BeBox. The Speaker slider only controls the BeBox's internal speaker.

Time

Clock circuitry on the BeBox motherboard keeps the time, even when the BeBox is turned off. You set the date and time on the BeBox with the Time application. When you start Time, a window opens with the BeBox's notion of the current date and time. Click the component of the date or time you want to change and then set the correct date or time using the controls for that component: Adjust the date or time using the small arrows; type an hour, minute, or second; or directly click a date on the calendar. Your changes take effect immediately.



Time stores the location of its window the last time you moved it in the **Time_settings** file in the **/system/settings** folder.

Changing the Boot Disk

When you turn on the BeBox, it looks in a fixed order for a disk to boot (or "start up") from. First, it looks for system software on a floppy disk. If it doesn't find a floppy disk with system software, it then looks for any other kind of removable disk, such as a CD-ROM. If it doesn't find a removable disk, it checks the "boot preference"—a setting stored in a chip on the motherboard—to determine what disk you prefer to use as the startup disk. The BeBox then tries to start up from the disk specified in the boot preference. If for some reason the preferred disk isn't available, the BeBox tries to start up from any other available disk. (For more information on selecting a boot disk when you turn on the BeBox, see "Selecting a Boot Disk" on page 18.)

The boot preference is initially set to the master IDE disk. If you install the Be OS on another disk, the Installer asks if you want to change the boot preference to that disk.

You can also change the boot preference by typing the appropriate **boot** command in a Terminal window:

Type	To set the boot preference to
`boot 0 0`	The single IDE disk connected to the BeBox, or the master IDE disk if there are two connected.
`boot 0 1`	The slave IDE disk, if there are two IDE disks connected to the BeBox.
`boot 1 n 0`	The SCSI disk set to SCSI ID number *n* (*n* can be from 0 through 6).

To find out the current boot preference, enter **boot** with no arguments in a Terminal window.

Adding Fonts

The BeBox includes a TrueType font renderer, which displays TrueType-format fonts on the screen at virtually any size. A small set of TrueType fonts is included with the Be OS. They're stored in the **/system/fonts/ttfonts** folder on the boot disk. Two bitmap fonts used by the Be software, Emily and Erich, are stored in the **/system/fonts/system_fonts** folder. (A third bitmap font, Kate, is built into the system software.)

Installing Additional TrueType Fonts

You can purchase additional TrueType fonts in PC (not Macintosh) format and install them on the BeBox, so they're available in applications that work with fonts, such as Edit and FontDemo. A few kinds of TrueType fonts don't work on the BeBox, but most do.

1. Copy the TrueType fonts you want to onto a Macintosh, UNIX, Windows, or DOS computer's disk.

   If you're using a recent version of the Macintosh operating system, you can just insert the disk the font came on in the floppy disk drive and copy the fonts you want onto your hard disk. For more information, see the Macintosh *User's Guide* or on-line help.

2. Use **ftp** to copy the fonts to the BeBox.

   If you don't have a network set up, you can also use **tar** to copy the fonts to the BeBox using a floppy disk. For more information about **ftp** and **tar**, see "Transferring Files to and from the BeBox" on page 115.

3. Move the fonts to the **/system/fonts/ttfonts** folder on the BeBox.

4. Make sure the fonts have the **.ttf** extension (uppercase letters don't work).

5. Restart the BeBox.

Creating Screen Dumps

You can capture images of the BeBox screen ("screen dumps") and open them in the ImageViewer application. You can also convert the screen dumps you create into a format that's understood by Adobe Photoshop on the Macintosh, so you can work with them in documents you create on the Macintosh.

Capturing a Screen Dump

To capture a screen dump, press the Print Screen key. This saves a Be ImageViewer file with an image of the screen in the root folder of the boot disk. The first dump you capture in this way is named **screen1**, the second **screen2**, and so on. You can double-click these files to view them in ImageViewer.

The size of the screen dump matches the resolution of the screen: 640 × 480, 800 × 600, and so on (for more information on setting the screen resolution, see "Screen" on page 105).

Converting a Screen Dump to Raw Format

If you want to work with a screen dump or other ImageViewer file on a Macintosh, you can save it in a "raw" format, which you can open in Adobe Photoshop and other image-editing applications.

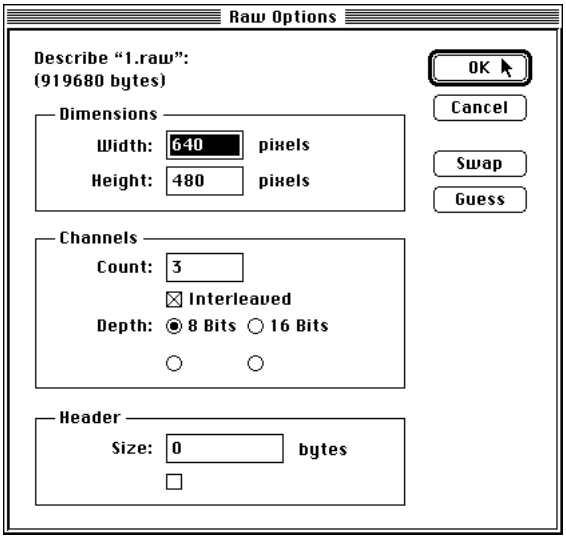
To convert an ImageViewer file, double-click it to open it in the ImageViewer application on the BeBox. Then choose the "Export to 24 Bits" command from the ImageViewer window's File menu. This saves a raw version of the file in the root folder of your boot disk, with the name **screen.raw**.

Opening a Raw ImageViewer File in Photoshop

Once you've converted an ImageViewer file to raw format, you can use **ftp** or **tar** to move it to a Macintosh (for more information, see "Transferring Files to and from the BeBox" on page 115).

When the raw file is on the Macintosh, start up Photoshop, choose Open from the File menu, and double-click the file you want to open.

In the dialog box that appears, enter the width and height in pixels that matches the resolution of the BeBox screen when you created the screen dump: 640 × 480, 800 × 600, and so on. Enter 3 in the Count box and make sure Interleaved is checked and the Depth is set to 8 Bits. Then click OK.



The Be file opens in Photoshop, where you can edit it, save it in other formats, and do any other work you want.

Transferring Files to and from the BeBox

There are two ways to transfer files to and from the BeBox and a Macintosh or other computer: Over a network or serial cable with **ftp** or with a floppy disk using **tar**.

Using ftp

You can use **ftp**, the "file transfer protocol" utility, to transfer files between a Macintosh (or any other computer that can run **ftp**) and a BeBox. To use **ftp** on a BeBox, you need to install an Ethernet adapter (as described in "Adding a Network Adapter" on page 8) or a null modem cable between the BeBox and another computer. You also need to use Network to configure the BeBox to communicate with the network or null modem cable (as described in "Network" on page 100).

ftp software is part of the Be OS, but you need to obtain **ftp** software for the Macintosh or other computer you want to exchange files with. There are a number of **ftp** packages available for the Macintosh. One good one, called Netpresenz (the successor to another good program, called FTPd), is available as "shareware" from a number of ftp and world-wide web sites on the Internet. (Shareware is software you can download and try out for free, but if you decide to keep it, you pay a small fee to its author.) Netpresenz makes it possible to get files from a Macintosh using **ftp** on another computer, such as the BeBox. Another good shareware application is Anarchie. Anarchie makes it possible to get files from another computer that can host an **ftp** session. To find an Internet site where you can get a copy of Netpresenz and Anarchie, visit **ftp.share.com/peterlewis** (an ftp site) or **http://www.share.com/peterlewis** (a web site).

You start **ftp** on the BeBox by double-clicking the Terminal application (in **/apps**) and in the window that opens, typing **ftp** followed by the IP address of the computer you want to connect to. If you have a DNS server on your network, or if you've set up a **/system/hosts** file, you can use the names and aliases in that file instead of the address:

```
ftp hostname
```

If **ftp** can find the other computer on the network and establish a connection, **ftp** displays a message such as:

```
Connected to hostname.node.com
220 Peter's Macintosh FTP daemon v2.2.0 awaits your command.
```

ftp then asks you to log in to the other computer:

```
Name (default hostname:default username):_
```

Supply your name as you're known on the other computer and press Enter. **ftp** then asks for a password, if required:

```
Password:_
```

The password you supply here is the one that you use on the other computer.

After the password is verified, you can transfer files between the two machines. To do this, you type **ftp** commands at the prompt.

The first command you should type is **help**:

```
ftp> help
```

This lists all the commands **ftp** recognizes. For more help on a specific command, type **help** *command*:

```
ftp> help cd
cd          change remote working directory
ftp>
```

When examining the commands, keep in mind that the *local* computer is the BeBox; the *remote* computer is the one on the other end of the Ethernet or serial cable.

Among the most commonly used commands are:

bin	To transfer files in binary mode (required for the Macintosh)
cd	To change directories on the remote machine
ls	To list the contents of remote directories
get	To retrieve a single file from the remote machine
mget	To retrieve more than one file
put	To transfer a file to the remote machine
mput	To transfer more than one file
prompt	To use **mget** or **mput** for many files, without approving each file
quit	To end the **ftp** session

For example, to transfer the files **MyApp** and **MyApp.xMAP** from the **Projects** folder on the Macintosh to the **MyAppProject** directory on the BeBox:

```
ftp> bin
200 Type set to I, binary transfer mode [macbinary disabled].
ftp> lcd /MyAppProject
Local directory now BeHardDisk:/MyAppProject
ftp> cd /MacHardDisk/Projects
250 "/MacHardDisk/Projects" cd successful.
ftp> prompt
Interactive mode off.
ftp> mget MyApp*
local: MyApp remote: MyApp
200 PORT command successful.
150 Binary transfer started (144k).
226 Transfer complete.
146693 bytes received in 1.63 seconds (89775 bytes/sec)
local: MyApp.xMAP remote: MyApp.xMAP
200 PORT command successful.
150 Binary transfer started (70k).
226 Transfer complete.
71282 bytes received in 1.28 seconds (55515 bytes/sec)
```

Using suntar

If you have a Macintosh and BeBox but they aren't connected to an Ethernet network or with a null modem cable, you can use the **tar** utility to transfer files via floppy disk. **tar** is included with the Be OS, but you need to obtain a copy of suntar—the free Macintosh **tar** utility. You can download suntar from **http://www.be.com** (the Be web site) or **ftp.be.com/pub/OtherUtilities** (on the Be ftp site).

In suntar, you can do everything you need to do using the commands in the Write menu:

1. Choose "Create new tar archive" from suntar's Write menu.

   A dialog box prompts you to insert a disk.

2. Choose the "Write data fork..." command (Macintosh resource forks are of no use on the BeBox). In the dialog box that appears, select the file you want to transfer and click Open. Or click Write Folder to write all the files in the current folder, or Write Selected Files to open a dialog box where you can select more than one file.

   You can choose "Write data fork..." as many times as necessary to put everything you want into the archive.

3. Choose the "End of writing" command and quit suntar.

   The disk is ejected.

4. On the BeBox, double-click Terminal (in **/apps**), insert the disk, and type:

   ```
   tar xv
   ```

   The files in the archive are copied onto the BeBox.

Adding Drivers and Add-Ons

You can add software to the BeBox that make it work with new kinds of devices or with familiar devices in new ways. This kind of software is called a driver. You can also add software that extends the functionality of existing applications. This kind of software is called an "add-on."

Adding a driver or add-on is as simple as copying it into the appropriate folder in the **/system** folder on the boot disk.

For the purposes of adding them, there are two kinds of drivers: Display adapter drivers (sometimes called "graphics drivers") and all other kinds of drivers.

To Add a Display Adapter Driver: Copy it into the **/system/add-ons/app_server** folder and restart the BeBox.

Caution: Don't replace, rename, or delete the supervga file in the **/system/add-ons/app_server** folder, or you won't be able to restart the BeBox.

Installing a display adapter card in the BeBox is described in Chapter 1, "Assembling the BeBox."

To Add Any Other Kind of Driver: Copy it into the **/system/drivers** folder and restart the BeBox.

To Add an Add-On: Create a folder in the **/system/add-ons** folder with the same name as the application the add-on adds onto (for example, the Be OS comes with an add-on for the Browser in the **/system/add-ons/Browser** folder). Then copy the add-on into that folder. The add-on is available immediately.

In most applications, the functionality the add-on provides is one or more commands in an Add-Ons menu. For example, the MakeArchive add-on adds the Make Archive command to the Add-Ons submenu in most Browser windows' File menus. You can use the Make Archive command to make a **tar** archive of selected files or folders.

4 Developing Be Applications

For the time being, you develop Be applications on another machine—specifically, on a Macintosh computer using the Metrowerks CodeWarrior 8 Integrated Development Environment—though you can also experiment with the preliminary development environment on the BeBox. When you've compiled your application on the Macintosh, you transfer it to the BeBox using a file-transfer utility or the Metrowerks cross-debugger (which you can also use to debug your application). Once your application is on the BeBox, you provide your application with icons, a unique identifying signature, and other information, in the form of *resources*. Finally, you can use the Be Debugger to track down problems in your application. This process is described in this chapter.

To set up a Macintosh so you can develop applications for the BeBox, you need to install Metrowerks CodeWarrior 8 for the PowerPC and the contents of the *Be Headers and Libraries* disk. On the BeBox, most of the development software is installed when you install release 1.1d7 of the Be OS, though if you want to use the cross-debugger, you need to copy the Metrowerks Nub from the **/optional** folder on the *Be OS* CD-ROM onto the BeBox hard disk.

Caution: The development environment won't work as described in this chapter unless you carefully follow the directions in *Installing or Upgrading to Release 1.1d7 of the Be Software*, the booklet that accompanies this user's guide.

Other applications you use on the BeBox when developing applications are:

- Terminal, the application you use to work with the Be shell, which is described in "Terminal" on page 96.
- Edit, the Be text editor, which is described in "Edit" on page 70.

- HeapWatch, an application for inspecting a running application's main memory heap.

  Note: HeapWatch is not included in release 1.1d7 of the Be OS—it's due back in time for the next release. But you can still verify an application's heap by running it from a Terminal window after setting the MALLOC_DEBUG environmental variable: Enter **export MALLOC_DEBUG=true** in a Terminal window and then Enter the application's path name. If the application fails with a memory error, information about the error is displayed in the Terminal window.

Setting CodeWarrior IDE Preferences

The CodeWarrior IDE on the Macintosh and the BeIDE on the BeBox both provide a large number of preferences you can use to customize the way the compiler works. You can set some of them any way you wish. However, there are a few that must be set (or unset) in order for compiled code to run successfully on the BeBox.

On the Macintosh, the **BeOS Application Project.**μ and **BeOS Driver Project.**μ project stationery files (in the **(Project Stationery)** folder in the **Metrowerks CodeWarrior** folder) have the preferences set correctly. On the BeBox, the project files (with the **.proj** extension) for the sample applications, drivers, and add-on in the **/develop/projects** folder have these the preferences set correctly for these kinds of projects on a BeBox.

On the Macintosh, you set preferences by choosing the Preferences command from CodeWarrior's Edit menu and clicking the settings you want to change. On the BeBox, you set preferences by choosing Settings from the BeIDE's main menu and clicking the button for the settings you want to change.

The following sections describe the preferences settings for the Macintosh and Be versions of the CodeWarrior IDE. In the title of each section, the name of the Macintosh preferences settings are listed first, followed by the equivalent Be Settings window button in parentheses.

For more information about the CodeWarrior preferences, see *Inside CodeWarrior 8*, the documentation for CodeWarrior available from Metrowerks.

Target

In the Macintosh version of the IDE, choose Macintosh PowerPC from the pop-up list at the top of the dialog box.

Because you only use the BeIDE to compile applications for the BeBox, there are no Target preferences.

Access Paths (Access Paths)

This preference tells the compiler where to look for header files and libraries.

You don't need to check the 'Treat #include <. . .> as #include ". . ." ' preference.

The Macintosh stationery and Be sample project files include the project folder in the upper part of the window. The small folders icon to the left of the folder name indicates that the compiler will search recursively through that folder (that is, the compiler will look for headers and libraries in the named folder, as well as in any folders contained in that folder). If your application defines header files that are in other folders, you need to add those folders to the list.

In the lower part of the Macintosh Access Paths preferences dialog box, the stationery is set so the compiler will look recursively in the **BeOS Support** folder for Be header files and libraries. If you didn't install the Be software exactly as described in *Installing or Upgrading to Release 1.1d7 of the Be Software*, you need to add the Be **Headers** and **Libraries** folders to this list.

In the lower part of the Settings window in the BeIDE, the sample applications set the compiler to look in different folders for headers and libraries—recursively in some and not recursively in others—depending on the nature of the application. Note that if your application uses the POSIX headers, make sure the compiler doesn't look recursively in the **/develop/headers/posix** folder on the BeBox.

C/C++ Language (Language)

In both the Macintosh and Be versions of the IDE:

- Check Activate C++ Compiler.
- Uncheck Enable C++ Exceptions.
- Check ANSI Keywords Only. Some symbols used in the kits clash with terms in the extended set of keywords.
- Check Enums are Always Int. The Be servers expect enumerated constants to be the size of an integer.
- Since the software kits don't deviate from C++ standards, the stationery and sample projects also set two other language preferences: Require Function Prototypes and ANSI Strict. These are recommended, but not necessary.

Note: RTTI doesn't work with classes derived from Be objects.

The stationery sets the Prefix File to **BeHeaders**—a precompiled, tokenized rendition of all Be header files. (This precompiled header file is built the first time you choose Make if the source file for the precompiled headers, **BeHeaders.pch**, is included in the project window). If you keep this setting, your source files won't need **#include** directives for any Be header files.

If you're writing a driver, set the preferences as described above for an application, but since drivers are written in C, not C++, uncheck Activate C++ Compiler, ANSI Strict, and ANSI Keywords Only.

PPC Processor (Processor)

In the Macintosh IDE:

- Choose PowerPC from the Struct Alignment pop-up list.

For efficiency, in both the Macintosh and Be versions of the IDE:

- Check Make Strings ReadOnly.
- Check Store Static Data in TOC.

- Check Use FMADD and FMSUB.

You can set any level of optimization you feel comfortable with.

PPC Linker (Linker)

In the Macintosh version of the IDE:

- Check Generate SYM File and Use Full Path Names. This produces the ".xSYM" file you can use with the cross-debugger.

In both the Macintosh and Be versions of the IDE:

- Check Dead-strip Unused Static Initialization Code (this option isn't available in release 1.4 of the CodeWarrior IDE).
- Check the Generate Link Map option so you generate an ".xMAP" file for debugging on the BeBox.
- Set the Initialization Entry Point to **_init_routine_**
- Set the Main Entry Point to **__start** (with two underbars).
- Set the Termination Entry Point to **_term_routine_**

If you're developing a driver, delete the settings from the three Entry Point fields.

PPC PEF (PEF)

In both the Macintosh and Be versions of the of the IDE:

- Choose None from the Export Symbols pop-up list if you're developing an application. If you're developing a driver, choose "Use #pragma" or 'Use ".exp" File", depending on the way you're writing the driver (for examples, see the S3 and skeleton driver examples in **/develop/projects**).

PPC Project (Project)

In both the Macintosh and Be versions of the IDE:

- Choose Application from the Project Type pop-up list, or choose Shared Library in the BeIDE if you're developing a driver.
- Type the name of your application in the File Name field (that is, the name of the executable file that you are building).
- Type the signature for your application in the Creator field (for more information about application signatures, see "Creating Resources for a Be Application" on page 138).
- Type **BAPP** in the Type field.

In the BeIDE:

- Choose the number of concurrent compiles you want the BeIDE to perform (this takes advantage of the built-in multithreading in the Be OS).

Developing Be Applications on a Macintosh

When you install CodeWarrior 8 and the *Be Headers and Libraries* disk on the Macintosh (as described in *Installing or Upgrading to Release 1.1d7 of the Be Software*, the booklet that accompanies this user's guide), you install the following folders and files you use with CodeWarrior to develop Be applications on the Macintosh:

- The **BeOS Support** folder (in the **Metrowerks CodeWarrior** folder) contains the Be headers and libraries, in the following folders:
    - **Headers** contains the same folders of header files as the **/develop/headers** folder on the BeBox (see "About the Development Environment on the BeBox" on page 133 for a detailed description), with one exception: **BeHeaders**, the precompiled, tokenized version of all the Be header files, isn't included. However, the source file for the precompiled headers is included. If you include this file (**BeHeaders.pch** in the **be** headers folder) in your project, **BeHeaders** is created the first time you build a project.
    - **Libraries** contains three libraries, **libbe.so**, the standard ANSI C library and the Be software kits; **libdll.a**, the Be runtime support library; and **libpos.so**, the POSIX library.
- The two "stationery" files, **BeOS Application Project.**µ, and **BeOS Driver Project.**µ. When you use the New Project command in the CodeWarrior IDE, choose the stationery file that matches the kind of project you're creating from the pop-up list in the New Project dialog box.
- **MW Cross-debugger for Be 1.4.1** (and its accompanying ReadMe file), which is in the **Metrowerks CodeWarrior** folder.

Using the CodeWarrior IDE

Using the CodeWarrior IDE is documented in *Inside CodeWarrior 8*, which is available from Metrowerks. However, stepping through the simple process of building the Be HelloWorld sample application is a good way to familiarize yourself with using CodeWarrior to build a Be application.

Create a folder on the Macintosh where you will keep your Be project files. Then transfer the six source files (the three files that end in ".cpp" and the three that end in

".h") from the **/develop/projects/HelloWorld** folder on the BeBox to the new folder on the Macintosh, using a file transfer utility, such as **ftp** or **tar** (see "Transferring Files to and from the BeBox" on page 115 for more information). Make sure you transfer the files as text files, or use a file-typing utility on the Macintosh (such as ResEdit or FileTyper) to set their file types as "TEXT", so CodeWarrior will be able to open them.

Double-click the CodeWarrior IDE to start it and choose New Project from the File menu. In the dialog box that prompts you to name your project, select "BeOS Application Project.μ" from the pop-up list, type a name for your project (typically the same name as your application, but with a ".π" extension—in this case, **HelloWorld.π**), and save the new project in the folder where you transferred the HelloWorld source files. A project window opens for your new project.

Choose Add Files from the Project menu and in the dialog box that appears, double-click each of the ".cpp" files and then click Done. This adds the HelloWorld sources to the project window. Note that the stationery includes the precompiled headers source file (**BeHeaders.pch**), so you will generate the precompiled header file (**BeHeaders**) when you build the project.

Choose Preferences from the Edit menu and in the dialog box that appears, click the PPC Project icon and replace "BeApp" in the File Name field with the name of your application, **HelloWorld** in this example (you can review "Setting CodeWarrior IDE Preferences" on page 122 to learn about the other preferences). Then click OK.

Now you can build your application: Choose Make from the Project menu. The IDE precompiles the Be headers first. Then it compiles and links HelloWorld.

Your compiled application and two symbol files are saved in the same folder as your project file and sources. You use the symbol file with the ".xMAP" extension to debug the application on the BeBox (as described in "The Debugger" on page 146), or the symbol file with the ".xSYM" extension to debug your application with the cross-debugger on the Macintosh (as described in "Using the Cross-Debugger" on page 129).

Transferring Your Application to the BeBox

Once you've built your application, you can transfer it to the BeBox using **ftp** or **tar** (as described in "Transferring Files to and from the BeBox" on page 115), or by using the Metrowerks cross-debugger (as described in the next section). If you transfer the files using **ftp** or **tar**, don't forget to transfer the symbol file for debugging (it has the ".xMAP" extension).

Using the Cross-Debugger

The Metrowerks symbolic debugger is not yet available on the BeBox. In the mean time, you can use the Macintosh version of the Metrowerks cross-debugger that's included on the *Be Headers and Libraries* disk. The Macintosh cross-debugger works with a "nub application" on the BeBox to transfer the application you developed on the Macintosh to the BeBox, either across an Ethernet network or across a direct serial connection. Then you can use the commands and other tools in the cross-debugger on the Macintosh to run, stop, and debug the application on the BeBox.

Caution: The connection between the cross-debugger and the Nub is fragile. Follow the instructions in this section carefully or you will likely crash both the Macintosh and the BeBox when you try to use the cross-debugger.

Tip: If you're planning to communicate between the Macintosh and the BeBox across an Ethernet network, test the connection with **ftp** or another utility to make sure you can transfer files successfully, before you try to set up the cross-debugger and the nub. (For more information, see "Using ftp" on page 115.)

Setting Up the Nub on the BeBox

1. Double-click the nub.

   The nub is in the **metrowerks_nub** folder in the **/develop** folder, where you copied it from the **/optional** folder on the *Be OS* CD-ROM, when you installed the Be OS on the BeBox, as described in *Installing or Upgrading to Release 1.1d7 of the Be Software*. It's named **Nub**.

2. Choose Settings from the MetroNub window's File menu.

3. Click the Misc. tab in the NubSettings window and click TCP/IP, if you're using an Ethernet network, or Serial, if you've connected the Macintosh and BeBox directly by their serial ports.

4. If you're using a serial connection, click the Serial tab and replace the "3" in "serial3" with the number of the serial port you're using on the BeBox to connect to the Macintosh.

5. If you're using an Ethernet network, it's best to leave the TCP/IP Port Number in the TCP/IP tab at its default setting: 4712. This matches the default setting on the Macintosh cross-debugger.

6. Leave the Files tab alone for the moment.

7. Close the NubSettings window.

Caution: If you quit the Nub application, you must restart the BeBox before you can start it again and establish a connection with the Macintosh cross-debugger.

The nub stores your settings in the Be database. If its settings seem to get scrambled, or if the nub stops working, you can remove all the nub settings from the Be database by starting it from the command line in a Terminal window: **cd** to the folder that contains the nub and enter **Nub -reset**

Setting Up the Cross-Debugger on the Macintosh

Once you've set up the nub on the BeBox, you can set up the cross-debugger on the Macintosh.

1. Double-click the cross-debugger.

   The cross-debugger is named **MW Cross-debugger for Be 1.4.1**. It's in the same folder as the IDE, if you followed the instructions in *Installing or Upgrading to Release 1.1d7 of the Be Software*.

2. In the dialog box that asks you to select a SYM file, click Cancel.

3. Choose Target Preferences from the Edit menu and uncheck the "Automatically launch applications when SYM file opened" and "QC-aware" options. Then click OK.

4. Choose Cross-PPC Preferences from the Edit menu. If you're using an Ethernet network, click TCP/IP and type the IP address of the BeBox in the Address field and leave the "Port no" field set to 4712—unless you had to change the port in the Be nub. If you're using a direct serial connection, click Serial and click the button that matches the port the serial cable is connected to: Modem Port or Printer Port. Then click OK.

The cross-debugger is set up. You can leave it running, or quit it and restart it later, when you're ready to debug an application.

Debugging Applications

Using the CodeWarrior cross-debugger is explained in *Inside CodeWarrior 8*, the documentation available from Metrowerks. But in outline, the process of debugging a Be application is fairly simple.

1. Make sure you've set up the cross-debugger on the Macintosh and the nub on the BeBox, as described above.

2. Create a folder for your application on the BeBox. Put a copy of the resource file for the application in the same folder (if you've already created one) and give the resource file the same name as the application, but add the ".rsrc" extension.

   Creating resources and resource files is described in "Creating Resources for a Be Application" on page 138.

3. Start the nub on the BeBox (if it's not already running), choose Settings from the MetroNub window, and click the Files tab in the NubSettings window. Then click Add and in the Add Program window that opens, type the name of your application and "Save" it in the folder you created in the previous step. Then close the NubSettings window.

   Note: You must click in the Save panel's file name field to save a file in an empty folder.

   Though no file is saved in the folder yet, "saving" in this way tells the nub where to save the file it receives from the cross-debugger on the Macintosh.

4. Compile your application using the CodeWarrior IDE on the Macintosh.

5. On the Macintosh, drag your application's symbol file onto the cross-debugger icon.

   The application's symbol file has the same name as the application, but with an ".xSYM" extension.

   The code for your application is displayed in the cross-debugger windows.

6. Choose Run.

   If there isn't a copy of your application in the folder on the BeBox you specified in the nub, the cross-debugger transfers a copy to the BeBox and starts to execute the program, stopping at "main". When the connection is established between the cross-debugger and the nub, the message in the MetroNub window on the BeBox changes from "Waiting" to "Connected."

7. Set breakpoints if you want them.

   Places in your code where you can set breakpoints are indicated by a small dash in the column to the left of the code. Click a dash to set a break point there—the dash turns to a red dot.

8. Choose Run again.

   The cross-debugger starts the application on the BeBox, and the message in the MetroNub window changes from "Connected" to "Debugging."

9. When you're done debugging your application, choose your application's Quit command on the BeBox to quit it.

   Caution: If you quit the nub while it's connected to the cross-debugger, you will crash the nub. Always quit the cross-debugger before you quit the nub.

Developing Applications on the BeBox

Release 1.1d7 of the Be OS comes with a preliminary version of the Metrowerks development environment for the BeBox. You can experiment with this version to develop applications in two ways: Using the BeIDE, the version of the CodeWarrior Integrated Development Environment for the BeBox, or using the tools (**mwcc** and **mwld**) and the **make** command in a Terminal window.

Caution: The preliminary development environment is not supported by Be or by Be Developer Support: You experiment with it at your own risk.

About the Development Environment on the BeBox

When you install release 1.1d7 on the BeBox (as described in *Installing or Upgrading to Release 1.1d7 of the Be Software*, the booklet that accompanies this user's guide), the following folders for the development environment are copied to your hard disk:

- **/develop** contains the BeIDE and folders for the headers, libraries, tools, as well as some sample projects (these folders are described below).
- **/develop/headers** contains three folders.
    - The **be** folder groups the Be header files by kit (or similar software categorization). For example, the **app** folder contains the header files for the Application Kit, **storage** files define the Storage Kit, and so on. The **be** folder also contains **BeHeaders**, a precompiled, tokenized version of all the Be header files.

      Among the contents of **be**, you'll notice one folder—**net**—for a software kit that hasn't been documented in *The Be Book* yet. This kit is not fully implemented and its programming interface is not certain. Ignore it for now. As soon as it's ready, we'll provide you with new headers and complete documentation.
    - The **gnu** folder contains GNU headers.
    - The **posix** folder contains headers for creating POSIX-style applications plus the ANSI C headers.

- **/develop/libraries** contains **libdll.a**, the Be run-time support library. It also contains:
  - **libfl.a**, the GNU flex library (flex is a lexical analyzer)
  - **librx.a**, the GNU regular-expression parsing library
  - **libtermcap.a**, the GNU terminal-handling library
  - **ralloc.a**, a malloc debugging library
- **/develop/tools** contains the compiler and linker. These are called **mwcc** and **mwld**, respectively. There are a number of text files with documentation and release notes for the tools in **/documentation/Metrowerks/tools**. Other development tools are found elsewhere on the disk. For example, **make** is in **/bin**.
- **/develop/projects** contains folders of sample applications, including HelloWorld, BeBounce, an S3 display adapter driver (complete with a ReadMe and lavishly commented sources), a skeleton driver project, and the sources for the MakeArchive Browser add-on. Each folder contains sources, a makefile, and a BeIDE project file.
- **/system/lib** contains two additional Be libraries: **libbe.so**, the shared library that includes the Be kits and the standard C library, and **libpos.so**, the POSIX library.

You'll also find an HTML edition of *The Be Book*, the documentation for the Be API, in **/documentation/The Be Book/HTML**.

Using the BeIDE

The BeIDE is based on the CodeWarrior IDE for the Macintosh; they work in much the same way. The CodeWarrior IDE is documented in *Inside CodeWarrior 8*, available from Metrowerks.

To create your own application or other project using the BeIDE, it's easiest to copy the sample project folder (in **/develop/projects**) that most closely matches the kind of project you want to create. Rename the project file (it has a ".proj" extension) to match the name of your project, and replace the existing source files with your own.

Double-click the project file you just renamed to start the BeIDE and to open a project window. In the project window, select the source files for the project you copied and choose Remove Files from the project window's Project menu. Then use the Project menu's Add Files command to add the source files for your project.

Choose Settings from the BeIDE's main menu to open the Settings window. Click the Project button and replace the file name of the project you copied with the file name of your project, and type your application's signature in the Creator field (for more information on application signatures, see "Creating an Application-Information Resource" on page 139). Click the Access Paths button and make sure the Project and System paths include all the headers and libraries you rely on in your code. Refer to "Setting CodeWarrior IDE Preferences" on page 122 to learn more about these and other settings. When you're done, click the Save button and then close the Settings window.

Now you're ready to build your application: Choose Make from the project window's Project menu. The BeIDE compiles and links your project. The compiled application is saved in the project folder, along with an ".xMAP" file for debugging (see "The Debugger" on page 146).

If you're creating an application, you'll probably want to give it an icon and other resources. To learn how, see "Creating Resources for a Be Application" on page 138.

Using the Tools

You can use the Metrowerks tools, **mwcc** and **mwld**, to build applications on the BeBox from the command line in the Terminal application. (The **/documentation/Metrowerks/tools** folder provides documentation for the tools from Metrowerks.) This approach is steeped in tradition: To build an application, you edit "make" files and invoke command-line driven tools.

Preparing the Makefile

The easiest way to build an application is to copy the sample project (in **/develop/projects**) that most closely matches the application you want to build, remove the example sources, and add your own.

Edit the application-building instructions, which are contained in the **makefile** file in the project folder. The essential steps are these (using the **makefile** from HelloWorld as an example):

- Replace the line

  ```
  app = HelloWorld
  ```

  with

  ```
  app = MyApplication
  ```

  where "MyApplication" is, of course, the name of your application. (Notwithstanding the example set by HelloWorld, the name of the folder in which your code lives needn't match the name of your application.)

- In the CPPOBJS section, replace the HelloWorld object files with your own:

  ```
  CPPOBJS = \
     $(OBJ)/HelloView.o \
     $(OBJ)/HelloWindow.o \
     $(OBJ)/HelloWorld.o
  ```

  Pay particular attention to the backslash characters. Neighboring lines must be separated by a backslash that immediately precedes the carriage return; the last line mustn't have a backslash:

  ```
  CPPOBJS = \
     $(OBJ)/MyApp.o \
     $(OBJ)/MyWindow.o \
     $(OBJ)/MyView.o \
     $(OBJ)/MyOtherView.o \
     $(OBJ)/MyMisc.o
  ```

- Near the bottom of the file is a list of compiler instruction entries that take the form

  ```
  $(OBJ)/HelloView.o: HelloView.cpp HelloView.h
     $(CC) $(INCLUDES) $(CFLAGS) -o $@ -c HelloView.cpp
  ```

  In general, for each file that you placed in the CPPOBJS section, you create an entry of this form and substitute your file names for HelloWorld's. The first line lists the file's "dependencies": The file to the left of the colon depends on the files that are listed (on that same line) to the right of the colon. For example, if

HelloView.h changes, **HelloView.o** will be automatically regenerated the next time you build the application.

You can list any number of dependencies. They needn't match the .o file's name. For example, let's say you have a header file (we'll call it **misc.h**) that all of the source files in your application look at. You would add the file to the list of dependencies for each of the source file entries:

```
$(OBJ)/MyApp.o: MyApp.cpp MyApp.h misc.h
    $(CC) $(INCLUDES) $(CFLAGS) -o $@ -c MyApp.cpp
```

The second line in the entry is (usually) a list of compiler instructions.

After you've edited the **makefile**, you're ready to build your application.

Building the Application

To build an application, you start the Terminal application, **cd** to the folder where your application lives, and type "**make**":

```
cd MyAppDir
make
```

The **make** program reads the **makefile**, then invokes the compiler (**mwcc**) and linker (**mwld**) according to the instructions that it finds there.

To start from scratch, type:

```
make clean
```

This removes all previously compiled object files as well as the application's executable file itself.

If you're creating an application, you'll probably want to give it an icon and other resources. Each time you compile your application, you need to add the resources to the application file. To learn how, see "Creating Resources for a Be Application" on page 138.

Creating Resources for a Be Application

Once you have your application on the BeBox, you need to provide it with:

- Icons that represent the application and its documents
- A unique signature that identifies the application
- Other information that will inform the way the application is launched (or "starts") and the way it communicates with other applications

All this information is stored at the end of the executable application file, or as a standalone file you can copy into your application file. These resources must be set up on the BeBox; it can't be done on the Macintosh. (Be resources are not the same as a Macintosh resource fork.)

However, you don't have to recreate the resources each time you recompile your application. You can set them up once in a standalone file and leave them on the BeBox. When you recompile your application, you copy them into new version of the recompiled application file (as described in "Adding Resources to Applications" on page 144).

Release 1.1d7 of the Be OS introduces a new way of storing resources. In earlier releases, every application used a companion file, with a "._rsc" extension, to store the resources associated with the application. In release 1.1d7, the ._rsc files are gone. Instead, an application's resources are stored at the end of the executable portion of the application file. You can also store resources as separate files—with no required extension, though it's convenient to give them ".rsrc" extensions.

You create and edit resources in either format—as parts of application files or as separate resource files—with the IconWorld application. You also use three of the tools in **/bin** to list the resources in a file, remove the resources from a file, or copy the resources from one file to another.

Using IconWorld

You use the IconWorld application, in the **/apps** folder, to create and edit a number of application resources, not just icons. You create or open an application or resource file in IconWorld in a number of ways:

- Drag the icon for an application or a resource file onto the IconWorld icon.
- First start IconWorld, then drag the icon for an application or resource file into the IconWorld window.
- Start IconWorld and use the Open command in the IconWorld window's File menu. (Note that the Open command opens the resource portion of a file, not the data portion. Therefore, it won't give you access to an image stored in a raw data file. To import a raw image into IconWorld, drag the file into the IconWorld window.)
- Start IconWorld and create the resources from scratch, or by copying in raw image data.

If you open an application file in IconWorld, the icons and other resources you add or change are added to the end of the application file when you choose Save, so the resources are assigned to that application. If you open a resource file in IconWorld, the resources you add or change are simply stored in the resource file when you choose Save: They aren't assigned to an application until you add them to one. You also create a resource file whenever you use the Save As command: The resources are saved in the resource file you name in the Save panel, even if you started by opening an application.

IconWorld can deal with only one application or resource file at a time; each time it opens an application or resource file, it closes the one it previously had open.

Creating an Application-Information Resource

IconWorld has an App Info menu, which you can use to set up an application-information resource for an application. This resource is of type 'APPI' (for "application information"), hence the name of the first item in the menu—Create APPI. To create or edit resources for a new or existing application or resource file, make sure this item is checked in the menu (by choosing it if it isn't checked). None of the other items in the App Info menu have any effect unless Create APPI is checked.

After Create APPI, the menu has three sections:

- Choose a launch behavior for your application—Single Launch, Multiple Launch, or Exclusive Launch.
- Turn on the Background App flag if your application doesn't display a user interface, and Argv Only if doesn't respond to messages.
- Set the application's identifying signature. IconWorld expects the signature to be a constant formed from up to four concatenated characters, such as 'abcd' or 'SLiM'. If you type numerals, such as "7890", the numbers will form the multicharacter constant '7890', not the integer 7,890.

  To make sure the signature is unique, contact the Be Developer Support group (**DevSupport@be.com**). You will soon be able to register your resources on the Be world-wide web site: **http://www.be.com**.

For documentation on launch behaviors and the other flags, refer to "Application Information" in the "Application Kit" chapter of *The Be Book*.

Creating Icons

There are two ways to create icon resources using IconWorld. You can draw the icon in IconWorld, either from scratch or by modifying another icon, or you can import a raw bitmap file that was created elsewhere. The raw bitmap must be fully specified 24-bit color data.

Every icon must have both a large 32 pixel × 32 pixel version and a small 16 pixel × 16 pixel variant. The Browser lets users choose whether to view the larger icon or the smaller "mini-icon." It displays the larger one by default, but always uses the smaller one in list view.

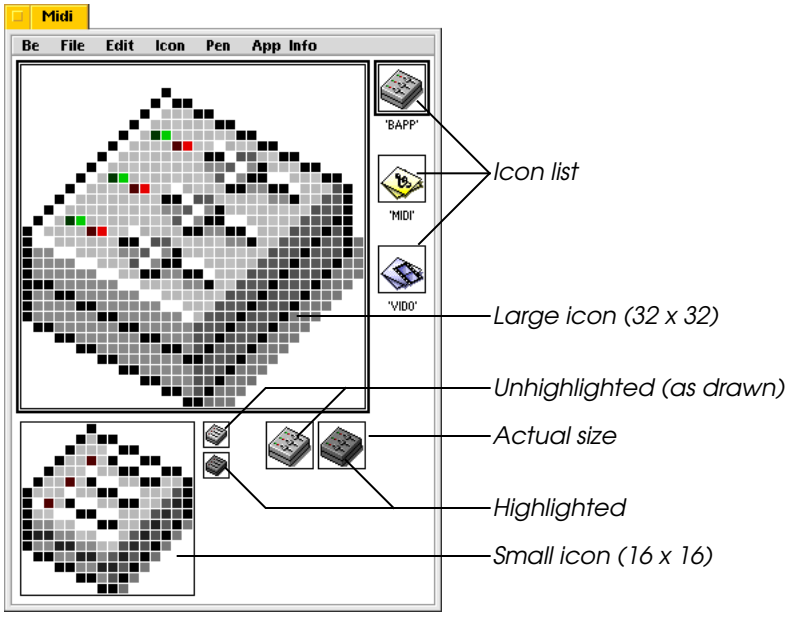
IconWorld assumes the drawing model discussed in "The Interface Kit" chapter of *The Be Book*. It would help to know something about that model—including front and background colors, patterns, and transparent pixels—before proceeding.

Drawing Icons

The main IconWorld window has areas where you can draw both a large icon and its smaller counterpart. The icon pair is treated as a unit and share a single file type, as discussed below.

Along its right edge, the IconWorld window lists the icons and their file types. Only the larger icon of the pair is shown. When an icon in this list is selected, its large and small variants are displayed in the drawing areas. The list marks the current item by highlighting its border. (To start, before an application or resource file has any resources, an empty square is marked.)

Towards the bottom, the window displays the current icons, both large and small, in their actual sizes. It shows the icons as they are drawn and also as they will be highlighted by the Browser. Highlighting is accomplished by systematically darkening selected colors.



IconWorld displays three palettes you can use for drawing icons—one of colors, one of tools, and one of patterns.

- Patterns are combinations of the front color (called the "fore" color by IconWorld) and a background color (called "back") as explained in "The Interface Kit" chapter in *The Be Book*. The default pattern is the solid front color, but you can pick another from the Patterns palette.
- The Colors palette lets you choose the front and background colors for whatever pattern you pick. The primary mouse button sets the front color and the secondary mouse button sets the background color (initially, these are the left and right mouse buttons, but you can change that as described in "Mouse" on page 99).
- The Tools palette has tools for drawing the icons and for selecting portions of what is drawn. They should be fairly self-explanatory. Try them out to see how they work.

IconWorld gives you a transparent background to draw on. Transparency doesn't have a hue, so it's displayed on screen as white (you can think of it as a clear sheet of plastic laid over a white surface). In the Colors palette, the transparent color is the white square located at the right bottom corner.

As you draw an icon, keep an eye on how it will be highlighted. You'll notice that pixels that are truly white (opaquely white) become darker when highlighted, just like other colors, but transparent pixels remain transparent (they stay white).

In general, you'll want the pixels around the outside boundary of the image to be transparent, and pixels inside—including those that are white—to be opaque (to have a true color). In this way, only the image will appear to be highlighted, not the background. If white portions inside the image don't darken when the image is highlighted, they'll appear to be transparent and not part of the image at all.

You can think of icons as being imaged in **OP_OVER** mode and placed over a white background, so that transparency always lets that white color show through.

Importing Icons

You can also import images into IconWorld as 24-bit color bitmaps. Drag a raw data file into IconWorld's main window.

An easy way to create a raw 24-bit color bitmap is to use ResEdit or your favorite paint application on the Macintosh. After the icon is drawn, you need to paste it into a 32 × 32 or a 16 × 16 document in Adobe Photoshop, or any other program that can export 24-bit raw bitmap data.

When creating a new file in Photoshop, set both the width and height to 32 pixels for the larger icon or 16 pixels for the smaller, set the resolution to 72 pixels/inch, and select RGB Color mode.

Once the icon is in a correctly sized Photoshop document, choose Save As from the File menu, name your icon, choose Raw from the pop-up list, and click Save. Then in the dialog box that appears, type **0** in the Header box, click the Interleaved Order option, and click Save (the other options don't matter).

Repeat this process so you have one raw file for the 32 × 32 bitmap and another for the 16 × 16 bitmap.

After you transport the files to the BeBox (using **ftp** or **tar**, as described "Transferring Files to and from the BeBox" on page 115), simply drag the files into the IconWorld window.

Although their colors will be adapted to the Be color map, IconWorld should display the images as you drew them. However, it interprets all white pixels as being transparent. Look at the way the image will be highlighted; everything that's white stays white. White areas inside the image will appear to be part of the background, as holes you can see through. If there are truly white areas inside the image, you'll need to repaint them with a shade of opaque white chosen from the Colors palette.

Setting File Types

When you save an application or resource file, IconWorld installs both the large and small icons in resources (of type 'ICON' for the large icon and 'MICN' for the mini-icon).

Each saved icon needs to be associated with a file type (not to be confused with the 'ICON' and 'MICN' resource types). The file type is a multicharacter constant that,

along with the application's signature, uniquely identifies files belonging to the application. The constant is formed from four concatenated characters; you can type these characters in the space beneath the icon in the list along the right side of main window.

If the icon represents the application itself (if it should be displayed for the executable file), its type should be 'BAPP'. If the icon represents a kind of document or other file that the application creates, its type should be identical to the type the application assigns to those files. Types and application signatures are assigned to files by calling BFile's **SetTypeCreator()** function. See the BFile class in "The Storage Kit" chapter of *The Be Book* for details.

For example, the document type for the Edit application is 'TEXT'. The Midi application associates its files with the 'MIDI' type. These types are local to the application—they have meaning only within the scope of the application's signature. Only the signature needs to be unique between applications.

Adding Resources to Applications

You add resources to an application in one of three ways:

- Open the application itself in IconWorld and create or edit its resources. Then choose Save. The resources are added to the end of the application file, replacing any resources that were there to begin with.
- Create or edit resources in IconWorld and then use the Save As command to name and save the resources in a standalone resource file. Then in a Terminal window, type:

  ```
  copyres resourcefile application
  ```

  to copy the resources in *resourcefile* into the *application* file (replacing any existing resources).

- Have the BeIDE or the Metrowerks cross-debugger add the resources to the application for you: Use IconWorld to create a resource file, by creating or editing resources, and use the Save As command to save a resource file with the same name as the application, but with the ".rsrc" extension. Place the file in the folder on the BeBox where you save your application with the BeIDE or where the cross-debugger transfers the application you developed on the Macintosh. Then each time you compile and link your application with the BeIDE or use the cross-debugger, the resources are copied from the resource file to the application.

You can use **copyres** to copy resources from any application or resource file into any other application or resource file. Simply type **copyres** *file1 file2*, where *file1* contains the resources you want to copy and *file2* is the file to which you want to add the resources. If *file2* already has resources, they're replaced. If *file2* doesn't exist, **copyres** creates a resource file with that name.

Two other applications in **/bin** are useful when working with resources: **listres** and **stripres**. In a Terminal window, type **listres** *filename* to view a table of the resources in *filename*. Type **stripres** *filename* to remove all resources from *filename*.

Getting the Icons to Be Recognized

When you add resources to an application, the Browser should immediately display your application with its new icons. However, a bug may prevent it from doing this. To force the Browser to display the new icon, open a Terminal window, **cd** to the folder that contains the application, and enter **setfile** followed by the name of the application.

The Debugger

The Be debugger lets you examine and alter the state of a running (or crashed) application. Currently, the debugger only displays assembly language instructions; a native source code debugger is anticipated.

In order to be debugged, an application must be accompanied by a symbol file. Symbol files, which are distinguished by their ".xMAP" extensions, are created automatically when you compile your applications (they're created by the **mwld** program). When you copy your application into a folder on the BeBox, be sure you copy the application's symbol file into the same folder.

Entering the Debugger

There are two ways to enter the debugger: You can fall into it, or force your way in.

- To fall into the debugger, your application must crash. Or, put more intelligibly, when an application crashes, it automatically enters the debugger. You can declare exceptional cases in your own code by using the **ASSERT()** and **DEBUGGER()** functions defined by the Support Kit. See the Support Kit documentation in *The Be Book* for details.

- To force your way into the debugger, you launch the **db** program. Normally, you launch **db** from the command line (you can also launch **db** by double-clicking its icon in a Browser window—it lives in the **/bin** folder). Launched from the command line, **db** accepts an argument that names the application that you want to debug. For example, to debug the MyApp application, you would launch **db** thus:

  ```
  $ db MyApp
  ```

  If MyApp is already running (and is a single-launch-only application), the debugger will "attach to" the running application. Otherwise, **db** causes a new image of the application to be launched and then takes over.

  The argument to **db** can also be a thread id number. For example, let's say MyApp is running and the id of its main thread is 87 (information that we got from running **ps**). To attach to the thread, we invoke **db** thus:

  ```
  $ db 87
  ```

Debugger Windows

When you enter the debugger, a new Terminal-like window appears. The title of the window will look something like this:

Team 40 Debug 2

Team 40 signifies that the team—or group of threads that belong to the same application—that's being debugged is team 40 (see the Thread documentation in the Operating System Kit chapter of *The Be Book* if you need a more detailed description of teams). **Debug 2** is simply an identifier for the window.

The team designation implies that the scope of the debugging process is team-wide. And so it is: Even though you may have invoked the debugger by telling it to attack a specific thread, the debug process that's running in a particular window has access to all the threads that belong to the designated team. You can switch between the threads in the team through debugger commands, as explained in a later section.

Each team that you debug is presented in its own window, and in only one window. If you invoke **db** on two threads (in separate invocations) that belong to the same team, the second invocation will "appear" in the window that was produced by the first invocation.

Using the Debugger

A debugger window displays, initially, the assembly code instruction to which the debugger is currently "pointing." The pointed-to instruction will not yet have been executed. (The format of the instruction display is described in the section "Instruction Display" on page 149.)

Immediately below the instruction display is the debugger prompt. The prompt displays the name of the thread that you're debugging, followed by a colon. For example, if you're debugging MyApp's main thread, the prompt would look like this:

```
MyApp:
```

To this prompt you type commands that let you examine and manipulate your application. The section "The Commands" on page 150 lists and describes all the debugger commands that are currently implemented.

Conventions

Some of the debugger commands take arguments; typically, these arguments are address or data values. The following rules apply to the interpretation of the values that you pass to the debugger (throughout the following, the debugger prompt is shown as "`db:`"):

- Numeric values are taken as hexadecimal. For decimal interpretations, prepend '#'. For example, "#16" (decimal 16) is the same as "10" (hexadecimal 10).
- Numeric values can be arithmetic expressions. **db** knows how to add, subtract, divide, and multiply, and can group expressions.
- An expression without a command displays the value in hexadecimal, decimal, and as a string.
- Register identifiers (r0...r31, pc, lr, and so on) can be used in expressions. Also, register values can be set directly; for example,

  ```
  db: pc = 800003b0
  ```

  sets the program counter to the instruction located at the absolute address 0x80003b0.
- Symbols can be used in expressions. **db** performs auto-completion on unambiguous partial symbol names.
- Symbol names that look like hexadecimal numbers can be distinguished with a backquote; for example,

  ```
  db: `abc
  ```

  finds the symbol "abc", not the number 0xabc.

- The most recently entered address is represented by a dot (.). For example,

```
db: il 80000248 #4
...
db: il . #8
...
db: il . #16
...
```

  displays, successively, four, eight, and sixteen instructions starting at address 0x80000248.

- The "hat" character (^) following an address performs indirection.

Instruction Display

The output of commands that display instructions follows this format:

```
function:
+offset:hexDumpoparg1, arg2, ..., argN
```

The ingredients are:

- *function* is the name of the function that the thread (the thread that the debugger is examining) is currently executing.
- *offset* is the location of the current instruction, given as a hexadecimal offset into the function.
- *hexDump* is the instruction itself, encoded as a hexadecimal machine instruction.
- *op* and *arg1, arg2, ... argN* is the instruction presented in a more readable form. *op* is the instruction's assembly language operator (such as **lwi**, **addi**, **bc**, and so on). The *arg* values are the operator's arguments.

If the instruction's location isn't within 64k of a recognizable symbol, the "function:" line is omitted, and the instruction's absolute address is given (in place of the "+offset" value).

The Commands

General Commands

Command	Meaning
h	Help; display the list of **db** commands.
cv *value*	Convert *value* from decimal to hex, or from hex to decimal (as *value* does or doesn't have a prepended '#').
a	Repeat the previous command verbatim.
<Enter>	Continue the previous command. (Exactly what "continuing" means depends on the command.)
es	Exit to Shell; attempts to exit the current application (which can be perilous).
exit	A synonym for **es**.

Displaying and Setting Memory, Registers, and Symbols

Command	Meaning
dm *address count*	Display *count* bytes of data starting at the given address.
sm *address value*	Set the data at the given address to *value*.
db | **dh** | **dw** *address*	Display *address* as a byte, half-word, or full-word.
sb | **sh** | **sw** *address value*	Set *address* as the byte, half-word, or full-word given by *value*.
wh *address*	Tell where *address* is (as an offset into a function or a portion of memory, for example).
regs	Display the contents of the CPU registers.
set *symbol value*	Sets the value of *symbol.* You can use this command to create new symbols.
sd *prefix*	Symbol dump; display the symbols whose names start with *prefix* (case insensitive).

svd	Displays all symbols, but in order of their values.
smd *prefix*	Symbol method dump; display the symbols whose method names start with *prefix*.

Displaying Frames and Instructions

Command	Meaning
sc	Stack crawl; list the frames on the stack.
il *address* [*count*]	List *count* instructions starting at *address*; *count* defaults to 5.
ip [*address*]	List a page of instructions starting at *address*; the argument defaults to pc – 20.

Stepping and Breaking

Command	Meaning
s	Step one instruction.
so	Step one instruction without entering a function call.
t	Same as **so**.
g	Go; continue without stopping until the program hits a breakpoint (or exits).
gt *address*	Go to; continue until the instruction at *address* is met. (This creates a one-shot breakpoint.)
mr	"Magic Return"; runs until returned form the saved return address in the current frame (usually the current function, unless it is a "leaf" routine with no stack frame).
br [*address*]	Set a breakpoint at *address*. Without the argument, this displays all breakpoints.
cb *address*	Clear the breakpoint at *address*.
cab	Clear all breakpoints.

Accessing Other Threads in the Team

Command	Meaning
ps	Display information about the threads in this team.
nx	Switch to the "next" suspended thread in the team.
stch	Toggle the "stop-child" state. If child stopping is on, threads that are (subsequently) spawned by this thread are suspended at their first instruction. A spawned thread assumes the stop-child state of its parent.

A The BeBox Ports

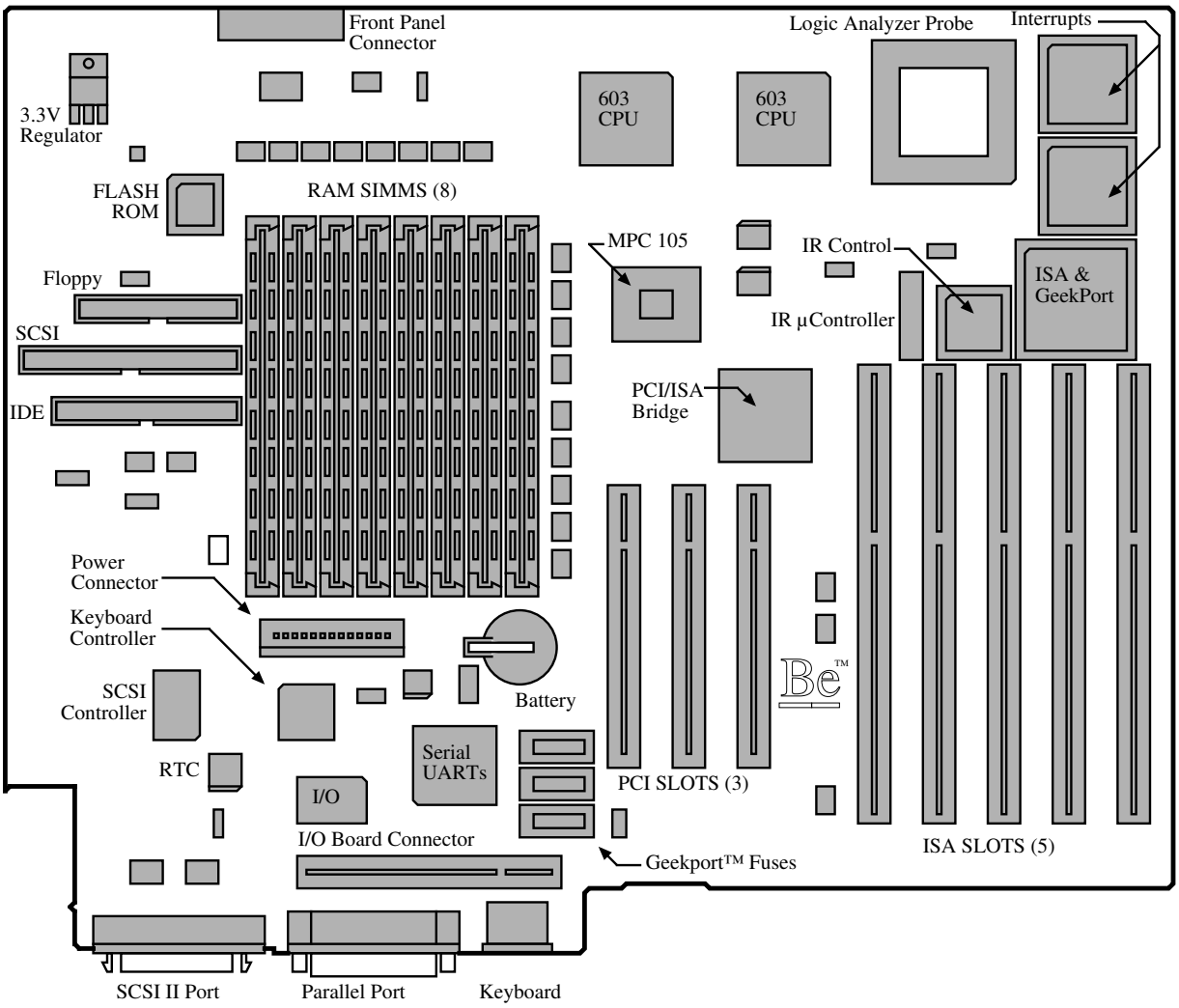
The BeBox is designed to facilitate a wide variety of I/O (input/output) connections. There are industry-standard and custom ports on the motherboard, some accessible on the back panel of the BeBox and some from inside. There are also a number of ports on the Be I/O board, most of which are available from the back panel.

This appendix describes the internal and external BeBox ports. The ports are illustrated from the perspective of a user looking into them. If a port has a label on the motherboard or I/O board, the label is included in parentheses in the title of the section that describes the port.

For information on opening the BeBox to access the internal ports, connecting internal and external devices, and devices that are compatible with the BeBox, see Chapter 1, “Assembling the BeBox.”

Internal Ports

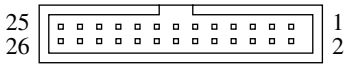
A number of BeBox ports, including those for internal disk drives, are available only by opening the BeBox.



The BeBox Motherboard

Internal BeBox Motherboard Ports

Front Panel Port (J1)



Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	O	PowerLED+	2	NA	PowerLED–
3	O	DiskLED+	4	NA	DiskLED–
5	I	Interrupt SW	6	NA	GND
7	I	Reset SW	8	NA	Reset RTN
9	O	LED_D0	10	NA	+5V
11	O	LED_D1	12	NA	+5V
13	O	LED_D2	14	NA	+5V
15	O	LED_D3	16	NA	GND
17	O	LED_D4	18	NA	GND
19	O	LED_D5	20	NA	GND
21	O	LED_D6	22	NA	GND
23	O	LED_D7	24	NA	GND
25	NA	Reserved	26	NA	Reserved

Speaker Port (J2)

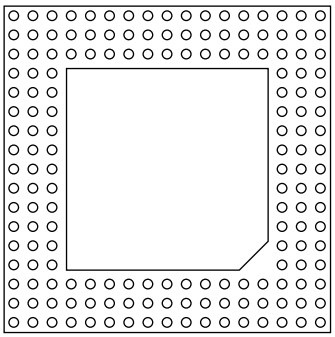
Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	O	SPEAKER +	2	O	SPEAKER –

Logic Analyzer Probe Port (J3)

The pin-out for the logic probe interface is available upon request from Be, Inc.

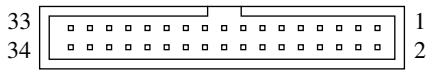
Be can supply a card and analyzer set-up libraries that connect the CPU bus to an HP 1660A.

Please contact Be developer support (e-mail: **devsupport@be.com**) for information on using this port.



Floppy Disk Drive Connector (J4)

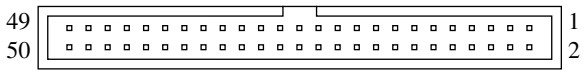
The 32-pin floppy connector is compatible with industry-standard, 3.5-inch floppy disk drives and flat-ribbon cables.



Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	NA	GND	2	O	DEN0
3	NA	GND	4	NA	N/C
5	NA	GND	6	O	DEN1
7	NA	GND	8	I	–INDEX
9	NA	GND	10	O	–MTR0
11	NA	GND	12	O	–DR1
13	NA	GND	14	O	–DR0
15	NA	GND	16	O	–MTR1
17	NA	GND	18	O	–DIR
19	NA	GND	20	O	–STEP
21	NA	GND	22	O	–WDATA
23	NA	GND	24	O	–WGATE
25	NA	GND	26	I	–TRK0
27	NA	GND	28	I	–WP
29	NA	GND	30	I	–RDATA
31	NA	GND	32	O	–HDSEL
33	NA	GND	34	I	–DSKCHG

SCSI-II Connector (J5)

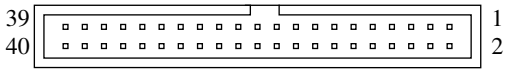
The internal 50-pin SCSI-II connector is compatible with industry-standard SCSI devices and flat-ribbon cables.



Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	NA	GND	2	I/O	SD0–
3	NA	GND	4	I/O	SD1–
5	NA	GND	6	I/O	SD2–
7	NA	GND	8	I/O	SD3–
9	NA	GND	10	I/O	SD4–
11	NA	GND	12	I/O	SD5–
13	NA	GND	14	I/O	SD6–
15	NA	GND	16	I/O	SD7–
17	NA	GND	18	I/O	SDP–
19	NA	GND	20	NA	GND
21	NA	GND	22	NA	GND
23	NA	Reserved	24	NA	Reserved
25	NA	N/C	26	NA	Term Power
27	NA	Reserved	28	NA	Reserved
29	NA	GND	30	NA	GND
31	NA	GND	32	I/O	ATN–
33	NA	GND	34	NA	GND
35	NA	GND	36	I/O	BSY–
37	NA	GND	38	I/O	ACK–
39	NA	GND	40	I/O	RST–
41	NA	GND	42	I/O	MSG–
43	NA	GND	44	I/O	SEL–
45	NA	GND	46	I/O	C/D–
47	NA	GND	48	I/O	REQ–
49	NA	GND	50	I/O	I/O–

IDE Connector (J6)

The 40-pin internal IDE connector is compatible with industry-standard IDE devices and flat-ribbon cables.



Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	O	–RESET	2	NA	GND
3	I/O	D7	4	I/O	D8
5	I/O	D6	6	I/O	D9
7	I/O	D5	8	I/O	D10
9	I/O	D4	10	I/O	D11
11	I/O	D3	12	I/O	D12
13	I/O	D2	14	I/O	D13
15	I/O	D1	16	I/O	D14
17	I/O	D0	18	I/O	D15
19	NA	GND	20	NA	KEY PIN
21	O	DMARQ [5]	22	NA	GND
23	O	DIOW–	24	NA	GND
25	O	DIOR–	26	NA	GND
27	NA	N/C (IORDY)	28	O	BALE–
29	I	DMACK– [5]	30	NA	GND
31	I	INTRQ (IRQ 14)	32	I	IOCS16–
33	O	DA1	34	NA	N/C (PDIAG)
35	O	DA0	36	O	DA2
37	O	CS1FX–	38	O	CS3FX–
39	I	DASP–	40	NA	GND

DRAM SIMM Slots (J7 through J14)

The BeBox uses PC-standard, 72-pin, 60-nanosecond, fast-page mode SIMMs.

The BeBox can use SIMMs that are either 32 or 36 bits wide. Though you can use parity SIMMs, parity isn't supported in the current release of the Be OS.

For more information on adding SIMMs to the BeBox, see "Adding Memory" on page 6.

PCI Slots (J16,17,18)

The BeBox uses PC-standard PCI connectors for a 32-bit, 5V card. 3.3V is supplied to the connector, as per the PCI specification. For more information about the PCI specification, see the *PCI Specification*, revision 2.0 or higher, or *PCI System Architecture* by Tom Shanley and Don Anderson (Mindshare), ISBN 1-881609-08-1.

Interrupts: INT A, B, C, and D are bused together on each slot. Each slot connects to both the legacy interrupt controller (inside the PCI-ISA bridge) and the Be Interrupt Controller.

	IDSEL	ISA Interrupt (82378)	Be Interrupt (Master/Slave)
J16	AD13	PCI1	DH11
J17	AD14	PCI2	DH12
J18	AD15	PCI3	DH13

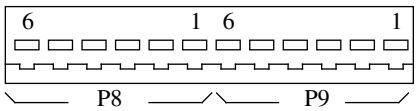
ISA Slots (J19 through J23)

The BeBox uses PC-standard, 16-bit ISA slots and timing. The timing is generated by the Intel 82378 PCI-ISA bridge. On-board I/O devices are isolated by buffers on the motherboard.

Please refer to the *ISA Specification* or *ISA System Architecture* by Tom Shanley and Don Anderson (Mindshare), ISBN 1-881609-05-7.

Power Connectors (J15)

The power connectors are compatible with industry-standard PC devices.



P8 Power Connector:

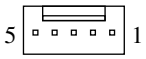
Pin	Signal Name
1	GND
2	GND
3	–12V
4	+12V
5	+5V
6	Power Good

P9 Power Connector:

Pin	Signal Name
1	+5V
2	+5V
3	+5V
4	–5V
5	GND
6	GND

Internal I/O Board Port

CD-ROM Drive Pass-Through Connector

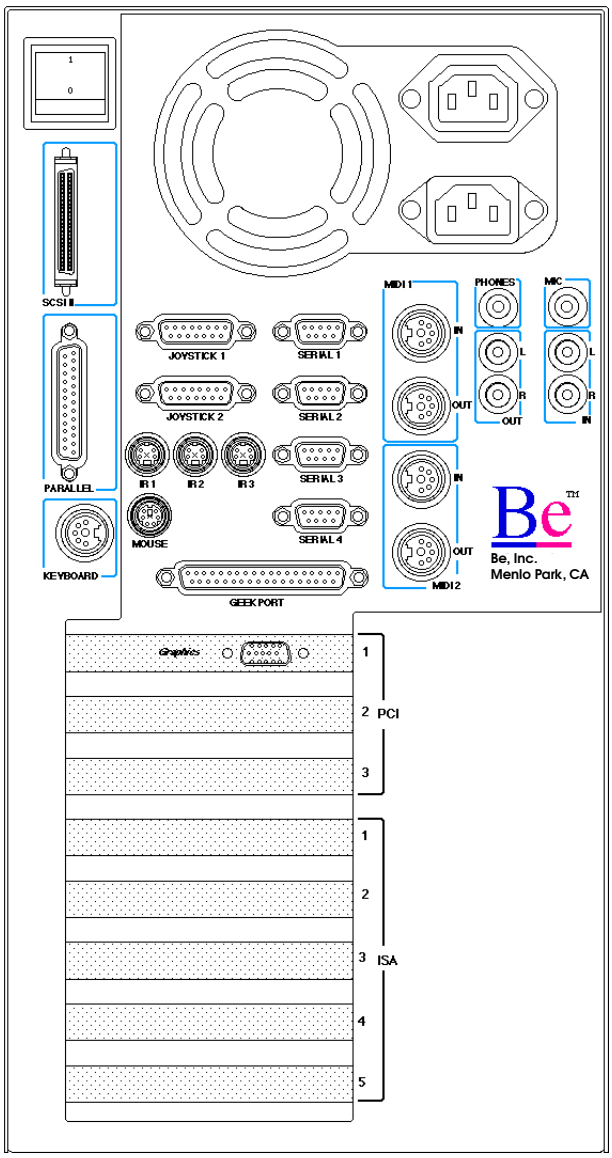


The CD-ROM drive pass-through cable connector on the I/O board is compatible with the pass-though port on a MediaVision ProAudio Spectrum 16 sound card.

Pin	I/O	Signal Name
1	NA	N/C
2	I	CD Audio Right
3	I	CD Audio Shield Return
4	I	CD Audio Left
5	NA	N/C

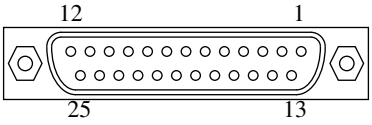
External BeBox Ports

Many of the BeBox ports are available from the back panel. (In the following illustration, the power supply fan vents and power sockets are rotated 180 degrees.)



Parallel Port (J27)

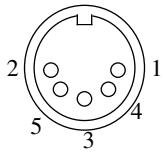
The Be parallel port is a PC-standard parallel port. It supports EPP and ECP modes.



Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	O	–STB	14	O	–AFD
2	I/O	Data 0	15	I	–ERR
3	I/O	Data 1	16	O	–INIT
4	I/O	Data 2	17	O	–SLIN
5	I/O	Data 3	18	NA	GND
6	I/O	Data 4	19	NA	GND
7	I/O	Data 5	20	NA	GND
8	I/O	Data 6	21	NA	GND
9	I/O	Data 7	22	NA	GND
10	I	–ACK	23	NA	GND
11	I	BUSY	24	NA	GND
12	I	PE	25	NA	GND
13	I	SLCT			
SHELL	NA	Chassis GND			

Keyboard Connector (J25)

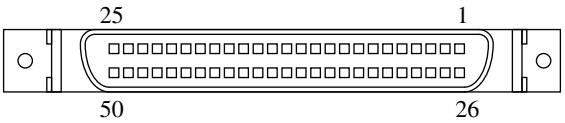
The Be keyboard port is a PC-standard keyboard port.



Pin	I/O	Signal Name
1	I/O	Clock
2	I/O	Data
3	NA	Reserved
4	NA	GND
5	NA	+5VDC (Fused)
SHELL	NA	Chassis GND

External SCSI-II Connector (J26)

The external SCSI-II connector is a PC-standard, compact 50-pin connector.



Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	NA	GND	26	I/O	SD0–
2	NA	GND	27	I/O	SD1–
3	NA	GND	28	I/O	SD2–
4	NA	GND	29	I/O	SD3–
5	NA	GND	30	I/O	SD4–
6	NA	GND	31	I/O	SD5–
7	NA	GND	32	I/O	SD6–
8	NA	GND	33	I/O	SD7–
9	NA	GND	34	I/O	SDP–
10	NA	GND	35	NA	GND
11	NA	GND	36	NA	GND Sense*
12	NA	Reserved	37	NA	Reserved
13	NA	N/C	38	O	Term Power
14	NA	Reserved	39	NA	Reserved
15	NA	GND	40	NA	GND
16	NA	GND	41	I/O	ATN–
17	NA	GND	42	NA	GND
18	NA	GND	43	I/O	BSY–
19	NA	GND	44	I/O	ACK–
20	NA	GND	45	I/O	RST–
21	NA	GND	46	I/O	MSG–
22	NA	GND	47	I/O	SEL–
23	NA	GND	48	I/O	C/D–
24	NA	GND	49	I/O	REQ–
25	NA	GND	50	I/O	I/O–
SHELL	NA	Chassis GND			

* Pin 36 must be externally connected to ground to disable on-board SCSI termination.

Headphone Port

The headphone port is compatible with standard stereo headphones with 3.5 mm jacks.

Contact	Signal Name
Sleeve	Common
Ring	Right
Tip	Left

Microphone Port

The microphone port is compatible with standard microphones with 3.5 mm jacks.

Contact	Signal Name
Sleeve	Common
Ring	Right
Tip	Left

Line-In Left Port

The four BeBox audio ports are line level, not microphone level; they're compatible with standard audio equipment.

Contact	Signal Name
Sleeve	Common
Tip	Left

Line-In Right Port

Contact	Signal Name
Sleeve	Common
Tip	Right

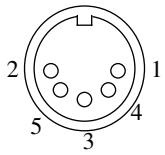
Line-Out Left Port

Contact	Signal Name
Sleeve	Common
Tip	Left

Line-Out Right Port

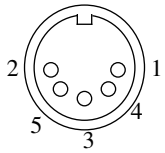
Contact	Signal Name
Sleeve	Common
Tip	Right

MIDI-In Ports



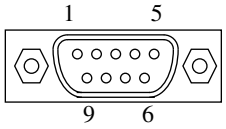
Pin	I/O	Signal Name
1	NA	N/C
2	NA	N/C
3	NA	N/C
4	I	MIDI RXDAT +
5	I	MIDI RXDAT –
Shell	N/A	N/C

MIDI-Out Ports



Pin	I/O	Signal Name
1	NA	N/C
2	NA	Cable Shield
3	NA	N/C
4	I	MIDI RXDAT +
5	I	MIDI RXDAT –
Shell	N/A	Chassis Ground

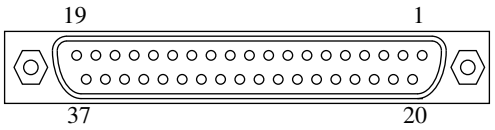
Serial Ports



Pin	I/O	Signal Name
1	I	DCD
2	I	SIN
3	O	SOUT
4	O	DTR
5	NA	Signal GND
6	I	DSR
7	O	RTS
8	I	CTS
9	I	RI
SHELL	NA	Chassis GND

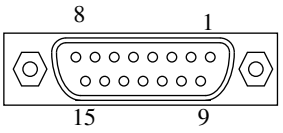
GeekPort

The GeekPort is a custom I/O port designed for the BeBox.



Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	NA	DGND	20	I/O	PA0
2	I/O	PA1	21	I/O	PA2
3	I/O	PA3	22	I/O	PA4
4	I/O	PA5	23	I/O	PA6
5	I/O	PA7	24	NA	AGND
6	NA	DGND	25	I	A/D_0
7	NA	+5V (Fused)	26	I	A/D_1
8	NA	Power GND	27	I	A/D_2
9	NA	+12V (Fused)	28	I	A/D_3
10	NA	Power GND	29	0	D/A_0
11	NA	–12V (Fused)	30	0	D/A_1
12	NA	Power GND	31	0	D/A_2
13	NA	+5V (Fused)	32	0	D/A_3
14	NA	DGND	33	NA	AGND
15	I/O	PB0	34	I/O	PB1
16	I/O	PB2	35	I/O	PB3
17	I/O	PB4	36	I/O	PB5
18	I/O	PB6	37	I/O	PB7
19	NA	DGND	SHELL	NA	Chassis Ground

Joystick Ports

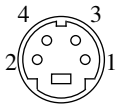


Pin	I/O	Signal Name	Pin	I/O	Signal Name
1	NA	+5V	9	I/O	+5V
2	I	BTN0_0	10	I	BTN1_0
3	I	X_0	11	I	X_1
4	NA	GND	12	NA	GND
5	NA	GND	13	I	Y_1
6	I	Y_0	14	I	BTN1_1
7	I	BTN0_1	15	NA	+5V
8	NA	+5V	SHELL	NA	Chassis Ground

IR Ports

The IR (infrared) ports are a new feature of the BeBox. The connector is a 4-pin DIN type.

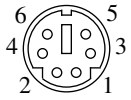
Each IR port supports input, output, or I/O devices. The interface is designed so that a single IR LED may be connected between pin 2 (+5V) and pin 1(IROUT), without additional external components.



Pin	I/O	Signal Name
1	I/O	IROUT
2	I	IRIN
3	NA	+5V(Fused)
4	NA	GND
SHELL	NA	Chassis Ground

Mouse Port

The BeBox mouse port is compatible with PC-standard, PS/2 mice.



Pin	I/O	Signal Name
1	I/O	Data
2	NA	Reserved
3	NA	GND
4	NA	+5V(Fused)
5	I/O	Clock
6	NA	Reserved
SHELL	NA	Chassis Ground