

Genera 8.1 Software Installation Guide

Genera 8.1 Distribution Media

Genera 8.1 is offered on two types of distribution media: tape or CD-ROM. This section describes these media and the software they contain.

Tape Distribution Media for Genera 8.1

This information is presented for each type of Symbolics machine: the 3600-Family and machines based on the Ivory chip: the XL-Family, MacIvory, and the Symbolics UX-Family.

Note that Symbolics has consolidated the distribution of sources for Genera 8.1; Regardless of how many different types of Symbolics machines you have at your site, you will receive only one set of source tapes, which contains sources, examples, fonts, documentation, and loadable systems for all machine types.

Tape Distribution Media for All Sites

All tape sites receive the following tapes:

- *Genera 8.1 Sources (All Platforms)*, two QIC-11 or two QIC-100 tapes, Distribution Format, P.N. 995735 or 995736 (These tapes contain Genera 8.1 system sources, examples, fonts, documentation and loadable systems for all machine types.)
- All tape sites will receive tapes for any layered products for which they are licensed.

Tape Distribution Media for 3600 Family Machines

Sites with only 3600 Family machines receive:

- *Genera 8.3 Distribution World for 3600-Family Machines*, one QIC-11 tape, IFS Format, P.N. 995855 (This tape contains microcodes and the 8.3 World and Net-boot Core for Symbolics 3600 Family machines.)
- *Genera 8.3 Sources (All Platforms)*, two QIC-11tapes, Distribution Format, P.N. 995856 (These tapes contain Genera 8.3 system sources, examples, fonts, documentation and loadable systems for all machine types.)

Tape Distribution Media for XL Family Machines

Sites with XL Family machines also receive:

- *Genera 8.1 Distribution Worlds for XL Family and MacIvories 1/2, 2/2*; two QIC-100 tapes, IFS Format, P.N. 995731 (These tapes contain a FEP Kernel, Flod files, the Genera 8.1 Development World for MacIvory and XL Family, Netboot Core for Ivory machines, and the Genera 8.1 MacIvory Delivery World.)

Tape Distribution Media for MacIvory Machines

Sites with MacIvory machines also receive:

- *Genera 8.1 Distribution Worlds for XL Family and MacIvories 1/2, 2/2*; two QIC-100 tapes, IFS Format, P.N. 995731 (These tapes contain a FEP Kernel, Flod files, the Genera 8.1 Development World for MacIvory and XL Family, Netboot Core for Ivory machines, and the Genera 8.1 MacIvory Delivery World.)
- *Symbolics MacIvory Applications*, version 4.0, Genera 8.1; one diskette, Macintosh Format, P.N. 995737
- *Symbolics MacIvory System*, version 4.0, Genera 8.1; one diskette, Macintosh Format, P.N. 995738
- *Symbolics MacIvory Installer*, version 4.0, Genera 8.1; one diskette, Macintosh Format, P.N. 995739
- *Symbolics MacIvory Development*, version 4.0, Genera 8.1; one diskette, Macintosh Format, P.N. 995740
- *Symbolics MacIvory FEP*, version 4.0, Genera 8.1; one diskette, Macintosh Format, P.N. 995741
- *Symantec AntiVirus for Macintosh (SAM)*, version 3.0; two diskettes, Macintosh Format, P.N. 995742

Tape Distribution Media for UX Family Machines

Sites with Symbolics UX Family machines receive:

- *Genera 8.1 Distribution Worlds for UX Family 1/2, 2/2*; two QIC-11 tapes, IFS Format, P.N. 995732 (These tapes contain a FEP Kernel, Flod files, and the Genera 8.1 Development World, Netboot Core for Ivory machines, and Delivery World for UX Family machines)
- *Symbolics UX UNIX Software for SunOS 4.1, Genera 8.1*, one QIC-11 tape, extract_unbundled Format, P.N. 995743

CD-ROM Distribution Media for Genera 8.1

CD-ROM Distribution Media for All Sites

All sites with Ivory-based machines receive the following CD-ROM distribution media:

Genera 8.3 Distribution Worlds, Sources, and Bundled Products, all platforms, 1 CD, P.N. 995854 (This CD contains Genera 8.3 software for all machine types, including 3600 Family microcodes, Ivory FEP Kernel and Flods, MacIvory Macintosh software, UX UNIX software, both Netboot cores, all Distribution worlds, system sources, examples, fonts, documentation and loadable systems. In addition, the CD contains sources and binaries for the layered products Symbolics C, Fortran, and Pascal.) This CD also contains the Symbolics Color layered product. **Note:** You must be under license for Symbolics Color in order to restore the Color product. Contact your sales representative for more information on the color product.

CD-ROM Distribution Media for XL Family Machines

All CD-ROM sites with XL Family machines also receive:

- *Symbolics Genera 8.1 CD-ROM XL Bootstrap Tape*, one QIC-100 tape, IFS Format, P.N. 995781 (This tape contains the Breath-of-Life IFEP kernel for XL Family machines, to enable CD-ROM installation on XL.)

CD-ROM Distribution Media for MacIvory Machines

CD-ROM sites with MacIvories also receive:

- *Symbolics MacIvory Installer*, version 4.2, Genera 8.3; one diskette, Macintosh Format, P.N. 995857
- *Symbolics MacIvory System*, version 4.2, Genera 8.3; one diskette, Macintosh Format, P.N. 995858
- *Symbolics MacIvory FEP*, version 4.2, Genera 8.3; one diskette, Macintosh Format, P.N. 995859
- *Symbolics MacIvory Applications*, version 4.2, Genera 8.3; one diskette, Macintosh Format, P.N. 995860

- *Symbolics MacIvory Development*, version 4.2, Genera 8.3; one diskette, Macintosh Format, P.N. 995861

Disk Space Requirements for Genera 8.1

Loadable systems and basic sources

Restoring all of the basic sources and loadable systems distributed with Genera 8.1 requires 33,000 LMFS blocks (EQUIVALENT TO 160,000 KBYTES FOR UNIX).

Sites with limited disk space may, of course, choose not to restore all of this data.

World sizes

3600 family worlds (in 3600 FEP filesystem blocks):

Genera-8-1.load	35,547
Netboot-Core-from-Genera-8-1.load	159

Ivory worlds (in Ivory FEP filesystem blocks):

Genera-8-1.ilod	40,437 (see note 1)
Network-from-Genera-8-1.ilod	10,694 (see note 2)
Genera-8-1-MacIvory-Delivery.ilod	17,644
Genera-8-1-UX-Delivery.ilod	18,532 (see note 3)
Netboot-Core-from-Genera-8-1.ilod	243

Note 1: This world is not distributed to MacIvory customers who have purchased Delivery Genera only.

Note 2: This world is required for UX-family machines, and may optionally be restored on XL- and MacIvory-family machines. This world is not distributed to MacIvory customers who have purchased Delivery Genera only.

Note 3: This world is distributed to UX customers (and those receiving CD-ROMs) only.

The following computation may be performed to estimate an acceptable FEP filesystem size for a Genera 8.1 Ivory machine. Sites that already have Ivory machines will of course be able to use closer estimates of their own usage, based on experience. First, sum the sizes of the Genera 8.1 worlds you plan to restore. Add 4,000 blocks for a site-configured incremental. Add 1,000 blocks for the FEP kernel, overlays, and other small files such as boot files. Finally, add space for paging files to accommodate your expected usage. 50,000 blocks is enough for running Development Genera; machines with large programs or data sets running on Genera will require more paging, while machines that run Delivery Genera can get by with less.

Tape Installation of Genera on 3600 Family Machines

This section describes how to install the Genera software from tape on 3600 Family machines. It consists of an overview and several sections that tell you how to perform the installation.

Read the overview so that you will understand any installation prerequisites, and to see a list of the installation steps you'll need to perform. After you have read the overview, go on to each section and perform the Genera 8.1 installation.

For a description of how the software is shipped, see the section "Genera 8.1 Distribution Media".

Overview of the Tape Installation Procedure for 3600 Family Machines

If you are a user of the Symbolics 3600 series of machines you need to:

1. Restore the Distribution World tape. See the section "Restoring the Genera Distribution World From Tape on a 3600-Family Machine".
2. Make a site-configured Genera 8.3 world for your site. See the section "Making a Genera Site-Configured World".
3. Restore the Genera 8.3 documentation, sources, examples, fonts, and non-loaded systems into your Lisp Machine File System (LMFS). See the section "Restoring the Genera Sources From Tape on a 3600-Family Machine".
4. Copy the new flod files. See the section "Copying New Flod Files to a 3600".
5. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
6. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

In the following sections, we explain how to perform each of the installation steps.

Tape Installation Steps for 3600 Family Machines

Restoring the Genera Distribution World From Tape on a 3600-Family Machine

Restore the Genera 8.3 world load files into your FEP file system, using the tape labeled *Symbolics Genera 8.3 Distribution World for 3600 Family Machines*.

1. Insert the *Symbolics Genera 8.3 Distribution World for 3600 Family Machines* tape into the tape drive of a Symbolics machine at your site. Type the CP command `Select Activity FEP-Tape`. In the FEP-Tape activity, type the com-

mand

Read Tape

and provide a tape specification for your tape drive.

The Read Tape command is also available in the FEP-Tape menu, but we recommend that you do not use the menu, because typing the command gives you more control over the tape specification.

2. You are asked if you want to restore the microcode files. Answer Y (for Yes) for the microcode file for your machine model.
3. You are asked if you want to restore the Netboot Core. Answer Y if you want to use netbooting at your site. For more about Netbooting, see the section "Netbooting".
4. You are asked if you want to restore the world load file. Answer Y (for Yes) and accept the default pathname, or specify a FEP pathname that indicates a world load, such as:

```
FEP0:>Genera-8-3.load
```

The FEP-Tape loading program then loads the world from tape onto disk.

5. Update the boot.boot file on your machine to load the new world. Your updated boot.boot file should look similar to this:

```
Clear Machine
Load Microcode fep0:>3640-mic.mic.430
Load World FEP0:>Genera-8-3.load
Enable IDS
Start
```

See the section "Contents of Hello.Boot Files and Boot.Boot Files".

6. Boot the Genera 8.3 Distribution world. In a Lisp Listener, type

```
Halt Machine
```

In the FEP, type

```
Boot
```

Restoring the Genera Sources From Tape on a 3600-Family Machine

This section describes how to restore the Genera documentation, sources, examples, fonts, and non-loaded systems. This step applies to sites with *only* 3600 machines.

Use the tapes labeled *Genera 8.3 Sources 1/2* and *2/2*.

Restoring the Genera sources in their entirety requires approximately 41,250 LMFS records (equivalent to approximately 165,000 FEP blocks or 204,000 KBytes for UNIX). If you have limited disk space (for example, if you have only one machine at your site), you may want to restore the sources selectively, to conserve disk space.

You can use a non-Symbolics machine to store the Genera documentation, sources, examples, fonts, and non-loaded systems. For information on how to do this, see the section "Storing Genera Sources on a Non-Symbolics Machine".

1. Boot the a site-configured Genera world and put the first Genera 8.3 source tape into the Symbolics tape drive.
2. Use the following command to restore the files from the source tape:

Restore Distribution :Menu yes

The system responds by displaying a menu, as shown in Figure !.

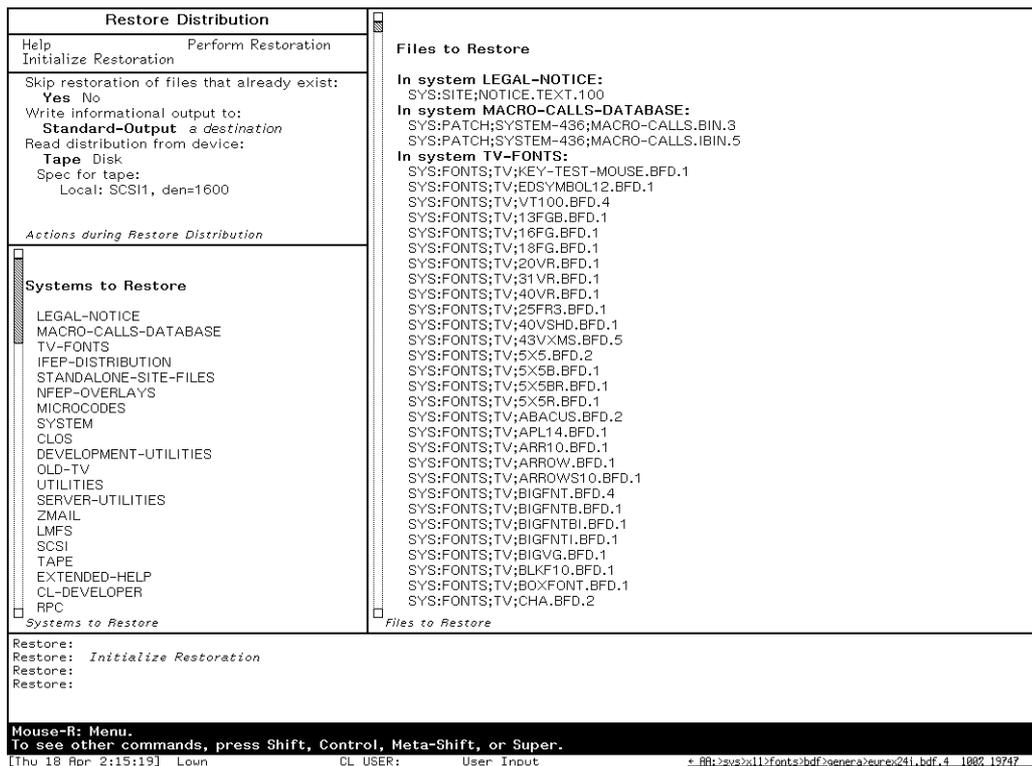


Figure 4. System Selection Menu for the Restore Distribution Command

In the restore distribution frame, specify a tape spec for a tape drive as follows:

3600-Host: Cart

3. For each source tape, click on [Initialize Restoration], and use the mouse to deselect systems that you do not want to restore from the menu. For more information on using the Restore Distribution frame, see the section "Restore Distribution Frame".

Once you have deselected those systems that you don't want to restore, click on [Perform Restoration]. This restores the selected systems from the source tape. Use [Perform Restoration] once for each tape.

All sites must restore the following systems:

LEGAL-NOTICE
MACRO-CALLS-DATABASE
TV-FONTS

All sites that include a 3600 Family machine must restore the following systems:

NFEP-OVERLAYS
MICROCODES

All sites who want up-to-date online Genera sources should restore the following systems: (Note that most of the non-loaded systems also include sources.)

SYSTEM
CLOS
DEVELOPMENT-UTILITIES
OLD-TV
UTILITIES
SERVER-UTILITIES
ZMAIL
LMFS
LMFS-DEFSTORAGE
SCSI
TAPE
EXTENDED-HELP
CL-DEVELOPER
RPC
EMBEDDING-SUPPORT
MACIVORY-SUPPORT
UX-SUPPORT
RPC-DEVELOPMENT
MACIVORY-DEVELOPMENT
UX-DEVELOPMENT
ZWEI
SERIAL
HARDCOPY
NSAGE

IP-TCP
X-REMOTE-SCREEN
X-CLIENT-FONTS

All sites who want up-to-date online Genera documentation should restore the following systems:

DOC (Genera Documentation Database)
IP-TCP-DOC
NFS-DOCUMENTATION
X-DOCUMENTATION

All sites might need to restore the following non-loaded systems:

PRINT (Print Spooler)
LGP2-SUPPORT (LGP2/3 Printer support)
DMP1-SUPPORT
MAILER
DOMAIN-NAME-SERVER

All sites may choose to restore the following non-loaded systems:

CLIM (Common Lisp Interface Manager)
GENERA-CLIM
CLX-CLIM
POSTSCRIPT-CLIM
CLIM-DEMO
CLIM-COMPATIBILITY
CLIM-DOC
CLIM-TEST-SUITE
CLIM-TUTORIALS
METERING
CONVERSION-TOOLS (Zetalisp to Common Lisp; Flavors to CLOS)
HACKS (demos)
EXAMPLES
NFS-SERVER (includes NFS-CLIENT)
X-SERVER
X-SERVER-FONTS
STATIC-RTIME (CLOS-based Static-Runtime)
FLAVORS-STATIC-RTIME (Flavors-based Static-Runtime)
DBFS-UTILITIES (Static backup utilities)
MAC-DEX (a sample MacIvory application)
HYPERCARD-MACIVORY (MacIvory interface to HyperCard)
HELPFUL-TOOLS (utilities for upgrading namespace and dialnet to Genera 8.0)
8.1-NETBOOT-COMPATIBILITY-PATCH (allows 8.1 netboot servers to serve 8.3 clients)
UNSUPPORTED (systems that are no longer maintained)

Sites interested in experimenting with RPC can choose to restore the following systems:

MACINTOSH-EXAMPLES (MacIvory example files)
ADDITIONAL-RPC-EXAMPLES (RPC example files)
UX-RPC-AND-FRIENDS (UX RPC code)

Sites with 3600 Family machines can choose to restore the .bin files for the following systems, which can be loaded into 3600 Family Genera worlds:

DEPENDENT-MODULES
X-REMOTE-SCREEN
X-DOCUMENTATION
MACIVORY-SUPPORT
UX-SUPPORT
RPC-DEVELOPMENT
MACIVORY-DEVELOPMENT
UX-DEVELOPMENT

Sites with 3600 Family machines can choose to restore the following system:

MICROCODE-SYMBOL-TABLES (for debugging purposes)

Tape Installation Instructions for XL Machines

This section describes how to install the Genera 8.1 software for XL machines. It consists of an overview and several sections that tell you how to perform the installation.

Read the overview so that you will understand any installation prerequisites, and to see a list of the installation steps you'll need to perform. After you've read the overview, go on to each section and perform the Genera 8.0 installation.

For a description of how the software is shipped, see the section "Genera 8.1 Distribution Media".

Overview of the Tape Installation Procedure for an XL

If you are an existing XL machine user (upgrading from Genera 8.0 software to Genera 8.1), you need to:

1. Restore the Distribution World tapes. See the section "Restoring the Genera Distribution Worlds From CD-ROM". Do not delete your old flod files when performing this step. If you are setting up a new XL Family machine shipped from Symbolics, you do not need to do this. The Genera 8.1 distribution worlds should already be on the disk of the new machine.

2. *(New sites only)* Create a site. See the section "Creating a New Site for Genera".
3. Make a site-configured Genera 8.1 world for your site. See the section "Making a Genera Site-Configured World".
4. Restore the Genera documentation, sources, examples, fonts, and non-loaded systems into your Lisp Machine File System (LMFS). See the section "Restoring the Genera Sources".
5. Copy the new flod files. You must perform this step even if you restored the new flod files and FEP kernel from the distribution world tape. See the section "Copying New Flod Files".
6. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
7. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

In the following sections, we explain how to perform each of the installation steps.

Tape Installation Steps for XL-family Machines

Restoring the Genera Distribution Worlds From CD-ROM

Note: CD-ROM installation only works properly in Genera 8.1.1 or later versions. Be sure to install the Genera 8.3 worlds using a machine running Genera 8.1.1. or later.

Restore the Genera 8.3 world load files into your FEP file system, using the CD-ROM labeled GENERA 8.3 DISTRIBUTION WORLDS, SOURCES, AND BUNDLED PRODUCTS.

1. Insert the Genera 8.3 CD-ROM into the CD-ROM drive at your site. Type the CP command `Select Activity FEP-Tape`. In the FEP-Tape activity, type the command:

```
Read Image File
```

We recommend that you type this command, rather than choose it from the FEP Tape menu, because the typed command gives you more control over the tape specification. Supply the pathname of a tape image on the Genera 8.3 CD-ROM according to the following:

CD-ROM drive on a Lisp machine

```
Lisp-host|CDROMn:>DISTRIBUTION>BASIC_WORLD.TAPE
```

CD-ROM drive on a Sun

```
Sun-host:/cdrom/distribution/basic_world.tape
```

where "n" is the SCSI address of the ROM Drive.

2. On Ivory-based machines, you are asked if you want to restore the FEP kernel and flod files. Answer Y (for Yes) and accept the default pathname.

Note: Do not delete your old FEP kernel and flod files. We recommend that you always save the previous version of the kernel and flod files for a backup. See the section "Saving Previous FEP Kernels and FLOD Files".

3. You are asked if you want to restore the Genera 8.3 Netboot Core. Answer Y (for Yes) and accept the default pathname.
4. You are asked if you want to restore the Genera 8.3 world load file. Answer Y (for Yes) and accept the default pathname, or specify a FEP pathname that indicates an Ivory world load, such as:

```
FEP0:>Genera-8-3.ilod
```

The FEP-Tape loading program then loads the world from the CD-ROM.

5. Update the boot.boot file on your machine to load the new world. Your updated boot.boot file should look similar to this:

```
Load World
Start
```

This automatically loads the most recently created world on your disk (excluding netboot cores). **Note:** The next time you boot your machine, you must cold boot the FEP in order to use the new FEP kernel and FLOD files.

See the section "Contents of Hello.Boot Files and Boot.Boot Files".

6. For the remaining Genera 8.3 distribution worlds, use the Copy File command to copy the appropriate Genera worlds from the "worlds" directory on the CD-ROM to your local FEP filesystem.

If the CD-ROM drive is connected to an NXP1000, XL or a MacIvory, the world pathnames will look similar to this (where *host* is the name of the CD-ROM host, and *n* in CDROM*n* is the SCSI address of the CD-ROM drive):

```
cdrom-host|CDROMn:>WORLDS>GENERA_8_3.ILOD
```

If the CD-ROM drive is connected to a Sun workstation, the world pathnames will look similar to this (where *host* is the name of the Sun CD-ROM host, and */cdrom* is the directory the CD-ROM is mounted onto).

```
cdrom-host:/cdrom/worlds/genera_8_3.load
```

Note: The ISO9660 pathnames on the CD-ROM contain underscores which should be changed to hyphens when you copy the files to your FEP.

For example, when copying the Genera 8.3 world from CD-ROM to a 3600, with the CD-ROM drive connected to a Sun, you would type the following to a Lisp Listener (where *3600-host* is the name of the 3600):

```
Copy File cdrom-host:/cdrom/worlds/genera_8_3.load 3600-host|FEP0:>genera-8-3.load
```

Similarly, with the CD-ROM drive, SCSI address 3, connected to a MacIvory or XL, you would type the following:

```
Copy File CDROM-HOST|CDROM3:>WORLDS>GENERA_8_3.LOAD (to)  
3600-host|FEP0:>genera-8-3.load
```

The worlds directory on the CD-ROM contains the following worlds:

```
GENERA_8_3.ILOD  
GENERA_8_3.LOAD  
GENERA_8_3_MACI_DELIVERY.ILOD  
GENERA_8_3_UX_DELIVERY.ILOD  
NETBOOT_CORE_GENERA_8_3.ILOD  
NETBOOT_CORE_GENERA_8_3.LOAD
```

If your site is set up, you are ready to make a Genera 8.1 site-configured world. See the section "Making a Genera Site-Configured World".

If you are a new site, you are now ready to set up your site. See the section "Creating a New Site for Genera".

If you are adding a new machine to your site, see the section "Registering a New Symbolics XL, MacIvory Machine, or 3600".

Tape Installation Instructions for MacIvory Machines

This section describes how to install the Genera 8.1 software for MacIvory machines. It consists of an overview and several sections that tell you how to perform the installation.

Read the overview so that you will understand any installation prerequisites, and to see a list of the installation steps you need to perform. After you have read the overview, go on to each section and perform the Genera 8.1 installation.

For a description of how the software is shipped, see the section "Genera 8.1 Distribution Media".

Overview of the Genera 8.1 MacIvory Installation Procedure

You can install Macintosh or Lisp software first. Any minor release of Genera 8 runs with any of the Macintosh software supplied with any of the Genera 8 re-

leases. Genera 8.1 runs only with Macintosh system 6.0.7 or later. If you are upgrading to a MacIvory model 3, you must be running 8.1, so you must install the Macintosh software first. See the section "Prerequisites for Installing New Macintosh Software".

There are five installation situations:

1. A new MacIvory user who bought the entire system from Symbolics
2. A new MacIvory user who bought only boards and a disk from Symbolics
3. A new MacIvory user who bought only boards from Symbolics
4. An existing MacIvory user upgrading from Genera 8.0 to 8.1
5. An existing MacIvory user upgrading to the MacIvory model 3

If you are a new MacIvory user who bought the entire system from Symbolics, you need to:

1. Set up the Macintosh as described in the Apple and any other vendor manuals.
2. (*New sites only*) Create a site. See the section "Creating a New Site for Genera". This step is optional if the MacIvory is the only Symbolics machine at your site.
3. Make a site-configured Genera 8.1 world for your site. (This step is optional if the MacIvory is the only Symbolics machine at your site.) See the section "Making a Genera Site-Configured World".
4. Restore the Genera documentation, sources, examples, fonts, and non-loaded systems into your Lisp Machine File System (LMFS). See the section "Restoring the Genera Sources".
5. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
6. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

If you are a new MacIvory user who bought only boards and a disk from Symbolics, you need to:

1. Set up your Macintosh according to the Apple and any other vendor manuals, if necessary.
2. Attach the disk according to the manufacturers instructions.
3. Install the Symbolics keyboard (including keyboard software) if you have it. See the section "Installing a Symbolics Keyboard and Three-Button Mouse on a MacIvory".

4. Install the MacIvory board. See the section "Installing the MacIvory model 3 Hardware".
5. If the new disk is *not* the boot disk, install the Macintosh software on the boot disk. See the section "Installing the MacIvory Diskettes".

If the new disk is the boot disk, install the software for your other peripherals on the new disk.

6. (*New sites only*) Create a site. See the section "Creating a New Site for Genera". This step is optional if the MacIvory is the only Symbolics machine at your site.
7. Make a site-configured Genera 8.1 world for your site. See the section "Making a Genera Site-Configured World". This step is optional if the MacIvory is the only Symbolics machine at your site.
8. Restore the Genera documentation, sources, examples, fonts, and non-loaded systems into your Lisp Machine File System (LMFS). See the section "Restoring the Genera Sources".
9. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
10. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

If you are a new MacIvory user who bought only boards, you need to:

1. Set up your Macintosh according to the Apple and any other vendor manuals, if necessary.
2. Install the Symbolics keyboard (including keyboard software) if you have it. See the section "Installing a Symbolics Keyboard and Three-Button Mouse on a MacIvory".
3. Install the MacIvory board. See the section "Installing the MacIvory model 3 Hardware".
4. Install the Macintosh software on the boot disk. See the section "Installing the MacIvory Diskettes".
5. Set up your Ivory partition.
 - a. Determine the size for the new partition. See the section "Decide How Much of the Disk to Dedicate to Ivory Use, and How Much to Macintosh Use".

- b. Create it using the MacIvory control panel. See the section "Create the Ivory Disk Partitions".
6. Restore the the FEP and Genera 8.1 worlds via the Ivory Breath of Life application. Be sure to specify [Create New Filesystem]. See the section "Loading a FEP File System From Tape for MacIvories".
7. (*New sites only*) Create a site. See the section "Creating a New Site for Genera". This step is optional if the MacIvory is the only Symbolics machine at your site.
8. Make a site-configured Genera 8.1 world for your site. (This step is optional if the MacIvory is the only Symbolics machine at your site.) See the section "Making a Genera Site-Configured World".
9. Restore the Genera documentation, sources, examples, fonts, and non-loaded systems into your Lisp Machine File System (LMFS). See the section "Restoring the Genera Sources".
10. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
11. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

If you are an existing MacIvory user (upgrading from Genera 8.0 software to Genera 8.1), you need to:

1. Restore the Distribution World tapes. See the section "Restoring the Genera Distribution Worlds From CD-ROM".
2. Install the MacIvory Support diskettes. See the section "Installing the MacIvory Diskettes".
3. Make a site-configured Genera 8.1 world for your site. See the section "Making a Genera Site-Configured World".
4. Restore the Genera documentation, sources, examples, fonts, and non-loaded systems into your Lisp Machine File System (LMFS). See the section "Restoring the Genera Sources".
5. Copy the new flod files. See the section "Copying New Flod Files".
6. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
7. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

If you are upgrading to a MacIvory model 3, you need to:

1. Install the new Macintosh software. Your Macintosh must be running system 6.0.7 or later. See the section "Prerequisites for Installing New Macintosh Software".
2. Install 8.1 following the procedure for upgrading from Genera 8.0 to 8.1.
3. Install the MacIvory board. See the section "Installing the MacIvory model 3 Hardware".

For information on setting up a blank disk for a MacIvory, see the section "Setting up a Disk for Use with MacIvory".

Background Information About Ivory Partitions

Before beginning this procedure, you should be familiar with some basics of the MacIvory, including the following information: A MacIvory system contains two processors, the Macintosh and the Ivory, along with two corresponding software environments, the Macintosh operating system for the Macintosh, and Genera for the Ivory. The front end of Genera is IFEP, a minimal system that provides various bootstrap and maintenance tools.

In a MacIvory system you deal with two file systems: MacFS, the Macintosh file system, and the FEP file system.

A disk used with MacIvory is partitioned into one or more Macintosh partitions containing Macintosh file systems, and one or more Ivory partitions containing the FEP file system. Macintosh programs can access only the Macintosh file system, while Ivory programs can access both file systems. However, certain Ivory files must be in the FEP file system. These include world files (.ilod), paging files (.page), and FEP files (.flod, .fep, and .boot).

If you use the LMFS file system or the Statice object-oriented database (not available in the Genera 8.1 release of Delivery MacIvory), note that their disk space is organized as files that also must be in the FEP file system.

Ivory disk partitions are distinct from Macintosh disk partitions. An actual disk drive can be divided into several Macintosh disk partitions, each appearing to the Macintosh software as a separate disk. An Ivory disk partition resides inside a Macintosh disk partition (it is actually an invisible file). Macintosh restrictions on dynamic creation of partitions require this. If Ivory disk partitions were implemented directly as separate Macintosh disk partitions, rather than as Macintosh files residing inside Macintosh disk partitions, you would not be able to create an Ivory disk partition without discarding the previous contents of the Macintosh file system. Also, not all Macintosh disk vendors support partitioning of their disks.

Each Ivory disk partition is known by a FEP file system unit number. The MacIvory control panel shows a scrollable list of the Ivory disk partitions in the order that their FEP file systems are numbered. Thus, the Ivory disk partition at the top of the list is known as FEP0:, the second is known as FEP1:, the next one down is known as FEP2:, and so on. We recommend naming your Ivory disk parti-

tions "MacIvory FEP0", "MacIvory FEP1", and so on, so that the names correspond to the numbers. See the section "Using the MacIvory Control Panel".

In most cases, you will not need to create more than one Ivory partition per physical disk drive. However, when working with a large disk, it can be useful to create more than one Macintosh partition, because the Macintosh file system is inefficient for large disks. Partitioning is explained in the documentation that comes with your disk. If you use multiple Macintosh partitions, we recommend that you keep Macintosh files and Ivory disk partitions in separate Macintosh partitions. Create one Macintosh partition to contain the Macintosh file system from which the Macintosh will boot. Create a second Macintosh partition that contains only your Ivory partition. If you want, you can create additional Macintosh partitions containing additional Macintosh files. This is not a concern if you do not use multiple Macintosh partitions.

In the following sections, we explain how to perform each of the installation steps.

Tape Installation Steps for MacIvory Machines

Prerequisites for Installing New Macintosh Software

In order to install Genera 8.1, your Macintosh must have the following Apple software:

1. System 6.0.7 or later (but not 7.0).
2. 32-bit QuickDraw
3. If you use our Hypercard interface, you need Hypercard 2.0v2.

If you do not have the appropriate software, contact your Symbolics Sales representative and we will supply you with upgrade kits as needed at no charge. If you receive Genera 8.1 on CD-ROM, system 6.0.7 and Hypercard 2.0v2 are included on the CD-ROM.

If you have a radius monitor, it must have a video card that is compatible with 32-bit QuickDraw. This means V2.0 or greater. Show Machine Configuration shows you the version of the video card in your radius monitor. Contact your radius dealer for an upgrade.

Upgrading to 6.0.7, 32-Bit QuickDraw, and Hypercard

If you have an upgrade kit, Apple documentation is included and you should do the installation according to that documentation.

If you have a CD-ROM containing the upgrade software, proceed as follows:

Installing 6.0.7 and 32-bit QuickDraw

1. Take the floppy disk named *Macintosh CD Setup* and boot the Macintosh. This is an Apple Diskette shipped with Apple CD-ROM Drives.
2. Insert CD-ROM into CD-ROM drive.
3. Open the CD-ROM icon by double clicking on it. You should see two folders,
 System Startup
 System Additions
4. Open the folder named System Startup.
5. Open the Installer. An alert box pops up saying *can't switch startup volumes*. Click on OK.
6. The Welcome to the Apple Installer splash screen comes up. Click on OK.
7. The "Easy Install" Dialogue box appears. Check that it has chosen the correct disk (click on Switch Disk until the right disk appears if it has chosen wrong). If the list of software looks wrong, contact Symbolics Customer Support.
8. Click on Install. An alert box appears. If you have a Macintosh IIfx, IIfx, or a IIsi, 32-bit QuickDraw is part of the system, so you should click on Quit. Otherwise, click on Continue.
9. Click on Customize.
10. Click on 32-bit QuickDraw.
11. Click on Install. When the alert box comes up, click on Quit.
12. Restart the Macintosh.

Installing Hypercard

1. Throw out your old Hypercard folder on your hard disk.
2. Empty the trash.
3. Insert the CD-ROM
4. Double click to open the CD-ROM
5. Find the Hypercard folder.

6. Drag the Hypercard folder to your hard disk. (You may want to move it to wherever your previous version was kept.)

Setting up a Disk for Use with MacIvory

This section tells you how to set up a SCSI hard disk for use with a MacIvory system. If you need information about Ivory Partitions, see the section "Background Information About Ivory Partitions".

The procedure to set up a disk consists of these general steps:

1. Deciding how much of the disk to dedicate to Ivory use, and how much to Macintosh use.
2. Creating the necessary Macintosh disk partitions.
3. Installing Macintosh system software and device drivers.
4. Installing Symbolics software that runs on the Macintosh half of the MacIvory system.
5. Creating the necessary Ivory disk partitions.
6. Installing software that runs on the Ivory half of the MacIvory system.

Decide How Much of the Disk to Dedicate to Ivory Use, and How Much to Macintosh Use

You must decide how to partition your disk between the Macintosh and the Ivory. This decision is based on what types of software you are going to run on the system, what size disk you have, and whether you are networked to a file server or need to store all your files locally. We can give some general guidelines here.

There are three different units of measurement of disk capacity in common use: megabytes, kilobytes, and pages. It is helpful to know how to convert among them.

Disk hardware sizes are quoted in megabytes. A megabyte is 1,048,576 bytes. These are sometimes unformatted capacities, which slightly overstate the amount of data you can store on the disk. For example, a 300-megabyte disk might actually hold around 287 megabytes, and a 380-megabyte disk might actually hold around 333 megabytes. Some disk partitioning tools also quote sizes in megabytes.

Macintosh disk partitions are measured in kilobytes. A megabyte is 1024 kilobytes. When the Finder tells you "288,167K in disk, 3,132K available", for example, this means the disk's size is 291,299 kilobytes, of which all but 3,132 kilobytes are in use to store files. The actual numbers will be different on your disk, of course.

Ivory disk partitions are measured in Ivory pages of 256 40-bit words (1280 bytes) each. There are 800 Ivory pages in 1000 kilobytes, and there are 819.2 Ivory pages in a megabyte. Each 1000 Ivory pages occupies 1250 kilobytes or 1.22 megabytes.

If you have a 300-megabyte disk, you might create an Ivory partition of 200,000 Ivory pages. This will occupy 250,000 kilobytes, leaving about 40,000 kilobytes for Macintosh files. If you needed more space for Macintosh files, either to run a large number of Macintosh applications or to use the Macintosh file system to store your Genera files, you might decrease the Ivory partition to 150,000 pages, leaving about 100,000 kilobytes for Macintosh files. On the other hand, if you were going to make very little use of the Macintosh file system, you could reduce its size to 20,000 kilobytes and create a 217,000-page Ivory partition.

The considerations for other disk sizes are similar. A typical partitioning of a 760 megabyte disk (with an actual capacity of about 650,000 kilobytes) might be 85,000 kilobytes for Macintosh files and a 450,000-page Ivory disk partition. On a 160-megabyte disk used with a small system, you might create a 100,000-page Ivory disk partition (just enough to hold the FEP files, one world, and a modest amount of paging space) and leave 20,000 kilobytes for Macintosh files.

Create the Macintosh Disk Partitions

If you are using a disk that already contains a Macintosh file system and Macintosh system software, skip to step 4.

If this is a fresh disk, start by formatting it and creating the Macintosh partitions, following the disk manufacturer's instructions. If you decide to put the Ivory partition in its own Macintosh partition, compute the size of that partition as outlined above, then figure out how much is left for Macintosh files. For rough calculations, assume 1-1/4 megabytes per 1000 Ivory pages. Depending on your disk software, in most cases you will need to first create the partition for Macintosh files that will hold the system folder. Then, create the Macintosh partition that will contain the Ivory partition.

Install Macintosh System Software and Device Drivers

(**Note:** if you are adding an additional disk to a MacIvory system and you do not want the new disk to be usable as a Macintosh startup disk, skip this step and skip installing the Symbolics Macintosh software. Proceed to "Create the Ivory Disk Partitions".)

Install the Macintosh system software onto the disk from diskettes according to Apple's instructions. Install any necessary device drivers according to the instructions that came with them. For example, if you have an Ethernet card, install its device driver.

Now install your Symbolics Macintosh software. See the section "Installing the MacIvory Diskettes" or see the section "Installing Macintosh Software From a CD-ROM".

Create the Ivory Disk Partitions

1. Bring up the MacIvory control panel. To find it, pull down the Apple menu and click on Control Panel. The left-hand side of the control panel is a list of control panel element icons. Scroll through this list to the MacIvory icon and click on it; this brings up the MacIvory control panel. If you don't see a MacIvory icon, recheck what you did in step 4.
2. Create an Ivory disk partition using the MacIvory Disk Partitions pane. Begin by clicking on the [New] button in the pane. The partition will be contained in the Macintosh disk partition displayed in the lower left corner of the control panel; if this is not the right disk partition, click on it until you see the disk partition you want. Once you have selected the disk partition, the control panel displays the number of Ivory pages available and fills in the size box. You can edit the size box and name box as needed. When you are finished, click on the [Make] button.

If you are creating multiple Ivory partitions, for example, if you have multiple disk drives, repeat this procedure as necessary.

For help with the MacIvory control panel, click on the [Help] button.

If this is not a fresh disk, it might be fragmented. The number of Ivory pages available on a fragmented disk can be smaller than the amount of free space. If you cannot create an Ivory partition of the desired size, you may need to run a disk compaction utility first. Such utilities are available from third-party software vendors. Alternatively, you can delete some of the files on the disk and restore them from backup copies later.

Install Software That Runs on the Ivory Half of the MacIvory System

As a final step, fill the Ivory disk partition you just created with a FEP file system containing the desired software and paging files. There are three different ways to do this, depending on what equipment you have available:

- If you have a Macintosh and a disk configured for MacIvory use, you can duplicate the existing disk provided one of the disks is an internal disk. See the section "Copying a FEP File System with a Macintosh".
- If you have a MacIvory with two disks attached, you can set up the second disk while running from the first disk. See the section "Copying a FEP File System Using Your MacIvory System".
- If your MacIvory does not have an initialized MacIvory disk you can load software from tape. You will need a QIC 100 tape drive on the local MacIvory for this operation. See the section "Loading a FEP File System From Tape for MacIvories".

In summary, the first method is fast but inflexible, the second method is fast and flexible; but both require an already initialized MacIvory disk. The last option requires a tape drive.

Note: For both disk-to-disk methods of copying, you should power down the Macintosh when adding or removing SCSI devices.

The following sections present detailed instructions for each method.

Copying a FEP File System with a Macintosh

This procedure requires that you have an initialized MacIvory disk that contains an Ivory partition of exactly the same size as the partition to be initialized. It also requires you to connect two disks to one Macintosh. Since a Macintosh can have only one internal disk drive, either the good disk or the disk to be initialized must be an external drive. Be sure that your disk has been configured for use with MacIvory as described in steps 1 through 5 of "Setting up a Disk for Use with MacIvory".

Refer to your disk manufacturer's instructions if you need information about how to set SCSI ID switches on the disk and what kind of cables and terminators to use.

Procedure

1. Connect both disk drives.
2. Bring up the MacIvory control panel and click on the [Copy] button to bring up the Disk Copy/Compare utility.
3. You should see two lists of Ivory disk partitions. Select the existing partition on the left (labelled "from") and the partition to be initialized on the right (labelled "to"). The SCSI ID and size of the selected partitions appear underneath. Be sure that you have selected the intended partitions before clicking on the [OK] button. If by accident you copy onto your good disk, there is no way to recover its contents. After you click on [OK], the utility first copies one partition into the other and then compares the two partitions to make sure there are no disk errors. The bar at the bottom of the window shows the progress of this operation, which takes a few minutes.
4. After you copy the FEP file system, both copies will have the same Hello.boot file. Since the Hello.boot file defines the network address, if you put the two disks on different MacIvory systems they will think they are the same machine. You have to set the network address by hand the first time you boot the new machine. After booting the machine, you will need to edit the Hello.boot file to contain the correct network address and reboot. This is not an issue for standalone machines.

Large sites that perform this procedure frequently may wish to dedicate an external disk drive as a master disk. This disk can be connected to any MacIvory at the site to provide that MacIvory's own disk with the desired software.

For mass production of disks, you can also copy an entire Macintosh partition, including both Macintosh files and Ivory partitions. Click on "Macintosh disks" in the upper left-hand corner of the Disk Copy/Compare utility to enable this operation.

This procedure does not require the Ivory coprocessor; it can be performed on a plain Macintosh II. Install the MacIvory control panel document as indicated in Step 4 of procedure documented in "Setting up a Disk for Use with MacIvory" and you will be able to copy disks. If you do have an Ivory coprocessor, it must not be running while you are copying disks. If the Ivory writes into its paging files, the compare operation will find differences between the two disks. If you have just restarted the Macintosh, the Ivory will not be running. Otherwise, you can stop it by dragging down the Ivory menu and clicking on Transfer to FEP. You can start it again later with Transfer to Lisp.

Copying a FEP File System Using Your MacIvory System

If you have a working MacIvory system with a good disk on it, you can connect a second disk to the system and load software onto the second disk. Since a Macintosh can have only one internal disk drive, either the good disk or the disk to be initialized must be an external drive. Be sure that your disk has been configured for use with MacIvory as described in steps 1 through 5 of "Setting up a Disk for Use with MacIvory".

Refer to your disk manufacturer's instructions if you need information about how to set SCSI ID switches on the disk and what kind of cables and terminators to use.

Procedure

1. Connect both disks to the MacIvory system.
2. Bring up the MacIvory control panel. You will see a list of Ivory disk partitions. The good partition(s) should be at the top and the partition to be initialized should be the last in the list; this ensures that existence of the new partition does not renumber any FEP file systems.
3. If the new partition is not last, rename it by clicking on it to select it, then click on the name box, editing the name, and click on the [Rename] button. For example, if you have two partitions named "MacIvory FEP0", you can rename the one on the new disk to "MacIvory FEP1". After renaming a partition, close the control panel.
4. Boot the Ivory by double-clicking on the Genera icon.
5. Once you reach Genera command level, pull down the Ivory menu and click on Transfer to FEP. You will use the FEP to create an empty FEP file system on the new disk. The .flod file that contains the command to do this is normally not scanned by the Hello.boot file, to avoid accidents.

6. Type `Scan I325-disk` and press RETURN (substitute your FEP version number for I325 if you are running a different version; the default filename shown by the Scan command includes this version).
7. Now type `Create Initial FEP File System` and the unit number of the new disk partition. Specify unit number 1 if the MacIvory system you are using has one good disk. Press RETURN.
8. Double-check the unit number before answering the confirming question; if you accidentally specify the unit number of a good disk, its previous contents will be lost.
9. Pull down the Ivory menu and click on `Transfer to Lisp`.
 - a. Now that you have an empty FEP file system, you must load software into it. To copy the FEP files, use Genera's `Copy Flod Files` command. Use the `:Disk Unit` argument to specify the Ivory partition being initialized. By default this will copy from the SYS host; you can also specify `:From Directory FEP0:>` to copy from the FEP files installed on the good disk.
 - b. To copy the world, use Genera's `Copy World` command. If the world you booted is `FEP0:>Genera-8-1.ilod` and the Ivory partition being initialized is `FEP1:`, you would write `Copy World FEP0:>Genera-8-1.ilod FEP1:`
 - c. To create a paging file, use Genera's `Create FEP File` command. For example, `:Create FEP File FEP1:>Page.page.newest 100000`
 - d. The above commands offer to create `Hello.boot` and `Boot.boot` files. The final step is to edit these files and make any necessary corrections. If you remove the disk and attach it to a different MacIvory system, it will be renumbered from `FEP1:` to `FEP0:` and the network address will be changed. Edit any numbers appearing in these files accordingly and save the files.

If you renamed the partition at the beginning of the procedure, go into the control panel and rename it back.

Shut down the Macintosh and disconnect the newly initialized disk.

Loading a FEP File System From Tape for MacIvories

You can use a tape drive to load a FEP file system into an Ivory disk partition on a MacIvory. Use this method when you do not have a working disk initialized for MacIvory to start from. If the MacIvory system does not normally have a tape drive connected to it, borrow your site's QIC-100 tape drive and connect it to the MacIvory system whose disk is to be initialized.

Be sure that your disk has been configured for use with MacIvory. See the section "Background Information About Ivory Partitions".

Procedure

1. Install the Macintosh software, see the section "Setting up a Disk for Use with MacIvory".. Place the FEP tape in the tape drive or the CD-ROM in the CD-ROM drive.
2. Run the Ivory Breath of Life Application, found in the MacIvory Applications folder. This loads the FEP kernel from the Macintosh File System.

Caution: The Ivory Breath of Life Application (or BOL FEP) prevents you from warm booting your Lisp system. **Do not try to boot Lisp using the BOL FEP, and do not experiment with it.**

A dialogue box comes up displaying a list of available MacIvory FEP partitions. The chosen one is highlighted and its disk icon appears next to the list. If you have more than one partition, click on the one that you are initializing.

Choose the appropriate parameters:

- a. Whether you are restoring from tape or CD-ROM.
- b. Whether you have enough disk space. See the section "Disk Space Requirements for Genera 8.1". The total number of available blocks is displayed.
- c. Number of blocks to allocate for paging. The default number of blocks to allocate for paging is half of the total available disk space.
- d. Whether or not you are running as a standalone site. The default is standalone. If you have more than one Symbolics machine at your site, Click on the box next to Standalone to deselect standalone and type in the correct network address.
- e. Select Restore Files Only if you are just installing new software. Select Create New Filesystem if you are initializing a new disk.

Caution: Create New Filesystem destroys the previous contents of the disk. Be sure you have the appropriate disk selected.

When you are satisfied, check that the FEP tape or CD-ROM is in the drive and click on Restore.

3. The cold load stream is displayed and the Disk Restore command is issued automatically for you. This loads all the files on the tape into the FEP file system, and sets up default paging files and boot files. If the FEP files take more than one tape, it pauses when it comes to the end of each tape and prompts you to mount the next one.

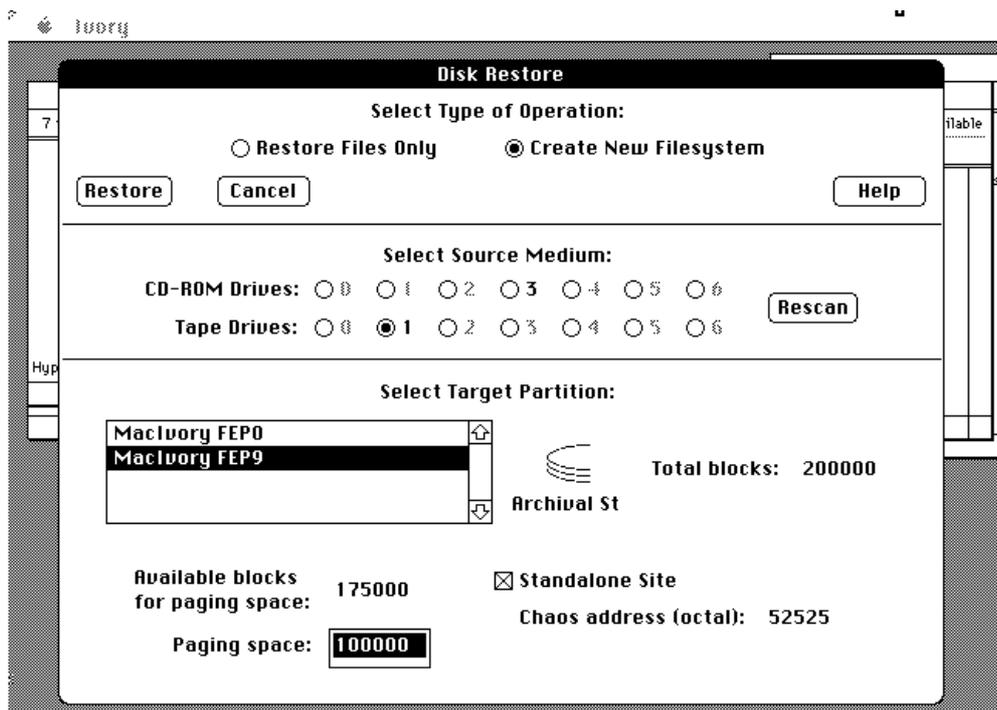
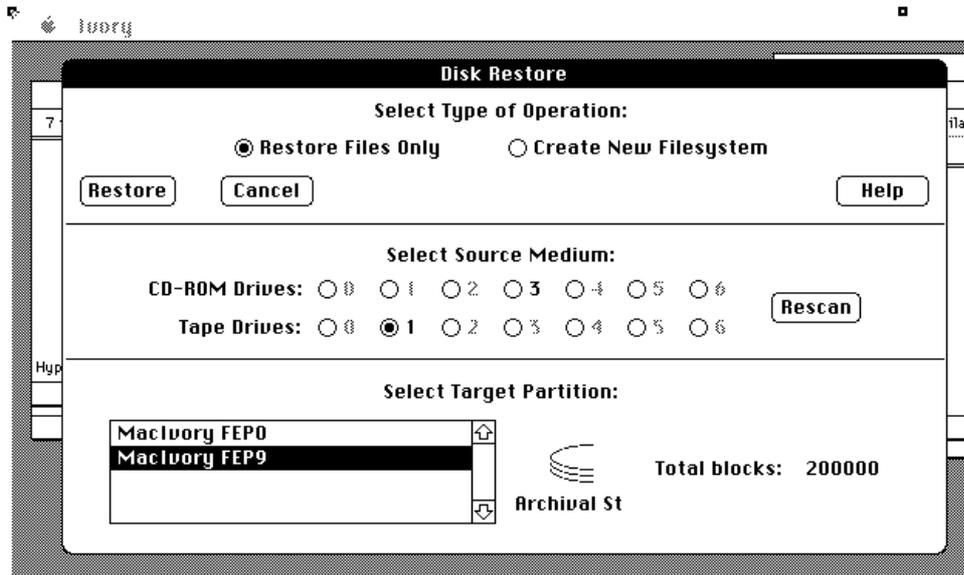


Figure 5. The Ivory Breath of Life

Once this operation is complete, quit the Ivory Breath of Life Application by clicking on Quit in the Ivory menu. You can do an additional disk restore from the

Ivory menu, if you wish. However, if you have additional tapes to load, it is preferable to boot Genera and use the FEP Tape Application under Lisp. Double-click on the Genera icon to cold boot.

CAUTION: Disk Restore using this method destroys the previous contents of a disk. Use the Ivory Breath of Life (BOL) application with care.

Restoring the Genera Distribution Worlds From CD-ROM

Note: CD-ROM installation only works properly in Genera 8.1.1 or later versions. Be sure to install the Genera 8.3 worlds using a machine running Genera 8.1.1. or later.

Restore the Genera 8.3 world load files into your FEP file system, using the CD-ROM labeled GENERA 8.3 DISTRIBUTION WORLDS, SOURCES, AND BUNDLED PRODUCTS.

1. Insert the Genera 8.3 CD-ROM into the CD-ROM drive at your site. Type the CP command Select Activity FEP-Tape. In the FEP-Tape activity, type the command:

```
Read Image File
```

We recommend that you type this command, rather than choose it from the FEP Tape menu, because the typed command gives you more control over the tape specification. Supply the pathname of a tape image on the Genera 8.3 CD-ROM according to the following:

CD-ROM drive on a Lisp machine

```
Lisp-host|CDROMn:>DISTRIBUTION>BASIC_WORLD.TAPE
```

CD-ROM drive on a Sun

```
Sun-host:/cdrom/distribution/basic_world.tape
```

where "n" is the SCSI address of the ROM Drive.

2. On Ivory-based machines, you are asked if you want to restore the FEP kernel and flod files. Answer Y (for Yes) and accept the default pathname.

Note: Do not delete your old FEP kernel and flod files. We recommend that you always save the previous version of the kernel and flod files for a backup. See the section "Saving Previous FEP Kernels and FLOD Files".

3. You are asked if you want to restore the Genera 8.3 Netboot Core. Answer Y (for Yes) and accept the default pathname.
4. You are asked if you want to restore the Genera 8.3 world load file. Answer Y (for Yes) and accept the default pathname, or specify a FEP pathname that indicates an Ivory world load, such as:

```
FEP0:>Genera-8-3.ilod
```

The FEP-Tape loading program then loads the world from the CD-ROM.

5. Update the boot.boot file on your machine to load the new world. Your updated boot.boot file should look similar to this:

```
Load World
Start
```

This automatically loads the most recently created world on your disk (excluding netboot cores). **Note:** The next time you boot your machine, you must cold boot the FEP in order to use the new FEP kernel and FLOD files.

See the section "Contents of Hello.Boot Files and Boot.Boot Files".

6. For the remaining Genera 8.3 distribution worlds, use the Copy File command to copy the appropriate Genera worlds from the "worlds" directory on the CD-ROM to your local FEP filesystem.

If the CD-ROM drive is connected to an NXP1000, XL or a MacIvory, the world pathnames will look similar to this (where *host* is the name of the CD-ROM host, and *n* in *CDROMn* is the SCSI address of the CD-ROM drive):

```
cdrom-host|CDROMn:>WORLDS>GENERA_8_3.ILOD
```

If the CD-ROM drive is connected to a Sun workstation, the world pathnames will look similar to this (where *host* is the name of the Sun CD-ROM host, and */cdrom* is the directory the CD-ROM is mounted onto).

```
cdrom-host:/cdrom/worlds/genera_8_3.load
```

Note: The ISO9660 pathnames on the CD-ROM contain underscores which should be changed to hyphens when you copy the files to your FEP.

For example, when copying the Genera 8.3 world from CD-ROM to a 3600, with the CD-ROM drive connected to a Sun, you would type the following to a Lisp Listener (where *3600-host* is the name of the 3600):

```
Copy File cdrom-host:/cdrom/worlds/genera_8_3.load 3600-host|FEP0:>genera-8-3.load
```

Similarly, with the CD-ROM drive, SCSI address 3, connected to a MacIvory or XL, you would type the following:

```
Copy File CDROM-HOST|CDROM3:>WORLDS>GENERA_8_3.LOAD (to)
3600-host|FEP0:>genera-8-3.load
```

The worlds directory on the CD-ROM contains the following worlds:

```
GENERA_8_3.ILOD  
GENERA_8_3.LOAD  
GENERA_8_3_MACI_DELIVERY.ILOD  
GENERA_8_3_UX_DELIVERY.ILOD  
NETBOOT_CORE_GENERA_8_3.ILOD  
NETBOOT_CORE_GENERA_8_3.LOAD
```

Installing Macintosh Software From a CD-ROM

CD-ROM installation is the same as floppy disk installation, except that you are using a CD-ROM drive instead of a floppy.

CAVEAT: To install onto a MacIvory, the CD-ROM has to be on the local MacIvory machine.

1. Insert the CD-ROM.
2. Double click on the CD-ROM icon to open it.
3. Open the folder named MacIvory.
4. Double click on the Installer icon.
5. Continue as for floppy disks.

Installing the MacIvory Diskettes

1. At a Lisp Listener, log out and halt the machine.
2. Choose Shutdown from the Ivory menu.
3. When the alert box appears, click on "OK" to shut down Ivory.
4. Choose Quit from the File menu.
5. If you have a MacIvory System folder on your hard disk, drag it to the trash and empty the trash.
6. If you have copies of Genera on your desktop, either remove them or put them back into the MacIvory Applications folder.

Then, perform the following steps:

1. Insert the MacIvory Installer floppy.
2. Double click on the floppy icon to open the MacIvory Installer.

3. Double click on the Installer icon in the MacIvory Installer.
4. Check the "Easy Install" Dialog box that pops up to make sure it is going to install the things you want. It should offer to install
 - MacIvory software for System 6 or System 7
 - MacIvory Applications and Fonts
 - Symbolics Keyboard for System 6 or System 7 (if you use the Symbolics keyboard)
 - MacIvory Development files (if you use MacIvory Development)

If the list of software is wrong, click on Customize.

If the disk it is going to install on is wrong, click on Switch Disk until the right disk appears (the *right* disk is the one from which you ordinarily boot).

5. Click on Install. The installer will ask for floppies as needed, and only asks for any given floppy once. If you are notified that any files on your hard disk are more recent than the files on the Genera 8.3 diskettes, click on Continue to install the 8.3 version of those files
6. The installer restarts the Macintosh when it is finished.

Installing a Symbolics Keyboard and Three-Button Mouse on a MacIvory

You can use a Symbolics keyboard and three-button mouse with your MacIvory and with any Macintosh application. To do so you need:

- An ADB box and cable
- A Symbolics keyboard
- A three-button mouse
- A floppy disk labeled MacIvory Installer

The MacIvory Installer floppy disk contains the resources necessary to ensure proper operation of your Symbolics keyboard and mouse with your Macintosh II.

Procedure

1. Install your ADB box, keyboard, and mouse:
 - a. Turn off your Macintosh by using the "Shutdown" item in the "Special" menu. If your hard disk has its own power connection, do not turn it off at this time.

- b. Disconnect the Apple keyboard and mouse from the Macintosh. Connect the Symbolics ADB box to the Macintosh using the ADB cable from your Apple keyboard.
- c. Connect the Symbolics keyboard and mouse to the ADB box by inserting the keyboard cable into the leftmost plug and the mouse cable into the rightmost plug.

Warning: Do not connect the Symbolics keyboard and Apple keyboard to your Macintosh simultaneously, as doing so may blow the ADB bus's internal fuse.

2. Install the software for the Symbolics keyboard and three-button mouse:
 - a. Turn on the Macintosh by pressing the power button on top of the ADB box. The Macintosh Responder asks you for the name of your machine. You may type in a machine name or just press RETURN.
 - b. Insert the MacIvory Installer floppy disk into the Macintosh's internal floppy drive.
 - c. Run the Installer program on this floppy disk by double-clicking the mouse on its icon. (Always use the leftmost mouse button while using the Finder or any other Macintosh software.)
 - d. Select the disk on which to install the Keyboard software by clicking on the "Switch Disk" button, if necessary. (Usually you install the Keyboard software on your start-up disk.)
 - e. Click on "Customize". Select Symbolics Keyboard Software.
 - f. Click on the "Install" button to install the resources on your disk.
 - g. Click on the "Quit" button to exit the Installer.

This adds the software for the Symbolics keyboard to the existing Apple software; it does not replace it. You can use either keyboard; simply unplug one keyboard and reattach the other one.

Your Symbolics keyboard and mouse may now be used with any Macintosh application.

Note: If you install new Macintosh OS software, you need to reinstall the keyboard software.

Configuring the Logitech MouseMan for Use with a MacIvory

You must be running MacIvory Support Software Version 4.2 to use the Logitech MouseMan with MacIvory. These instructions presume that you have already installed the MouseMan and its software according to Logitech's documentation.

1. Open the Mouse Key control panel.
2. Click on the Add button and select your copy of the Genera application.
3. Change the assignments for the middle and right mouse buttons to be Click.

Repeat steps 2 and 3 for the Unassigned application as well.

4. Close the Control Panel.

You can now use all three buttons on the MouseMan just as you would use the buttons on a Symbolics mouse while in Genera. The MouseMan will continue to behave according to your previous configuration instructions outside the Genera application.

If your site is set up, you are ready to make a Genera 8.1 site-configured world. See the section "Making a Genera Site-Configured World".

If you are a new site, you are now ready to set up your site. See the section "Creating a New Site for Genera".

If you are adding a new machine to your site, see the section "Registering a New Symbolics XL, MacIvory Machine, or 3600".

Installing the MacIvory model 3 Hardware

You must install the MacIvory model 3 hardware after installing the Genera 8.1 software. The MacIvory model 3 is a single card. It can be installed into a Macintosh II, IIx, IIcx, or IIci.

Important note: If installing the MacIvory model 3 card fills all available expansion slots in your computer, you should check the total power consumption of the cards to make sure they do not collectively exceed the recommended limit for your Macintosh computer.

Follow these steps to install the MacIvory model 3 card:

1. Shut down the Macintosh computer.
2. Remove the lid from the computer, according to the instructions that came with your Macintosh.
3. Touch the metal part of the power supply case inside the computer to discharge any static electricity that might be on your clothes or body. Do this before you touch any parts, or install any components, inside the computer.
4. **If you are upgrading** from a MacIvory model 1 or model 2, remove the MacIvory card and the NuBus memory card or cards.

You need to send these cards back to Symbolics as part of the upgrade agreement. To remove a card, pull it straight up along the top edge of the card.

Try not to wiggle the card or touch the connector on its lower edge. Now go to step 6.

5. **If you are adding a MacIvory card**, remove the metal shield (if present) behind the expansion slot you want to use.
6. Remove the MacIvory model 3 card from its static-proof bag. Hold the card by its top edge to avoid touching the connector on the bottom of the card.
7. Align the card over the expansion slot.
8. Press down firmly on the card until the connector is seated.

Don't force the card. If you meet a lot of resistance, pull it out and try again. To see if the card is properly connected, lift it gently. If it resists and stays in place, it is connected.

9. **If you are upgrading** from a MacIvory model 1 or model2, you will have some empty slots. If you still have the metal shields, it is good practice to put them in the empty slots.
10. Replace the lid on the computer.
11. Turn on the Macintosh computer.
12. Boot Genera 8.1.

Tape Installation Instructions for Symbolics UX Machines

Overview of the Tape Installation for Symbolics UX Machines

These are the main parts of the Genera 8.1 installation:

1. Installing the UX board (if necessary).

This part of the installation is done by Symbolics.

2. Installing and configuring the UX UNIX software.

The UNIX software is provided on a single QIC-11 cartridge tape labeled *Symbolics UX UNIX Software for SunOS 4.1, Genera 8.1*. The UNIX software is compatible with Sun's layered-product installer. Symbolics provides an installer program which asks some questions, automatically restores the tape, copies programs and other files to the correct places in the UNIX file system, and builds a new kernel that includes the UX device driver. For detailed in-

formation about this installation step, see the section "Installing and Configuring the Symbolics UX UNIX Software".

3. Installing and configuring the Genera 8.1 software.

The Genera software (FEP programs and Ivory world loads) is provided on two QIC-11 cartridge tapes labeled *Genera 8.1 Distribution Worlds for UX Family 1/2, 2/2*.

Note for new machines: When installing a new Symbolics UX machine, the installer program installs the Genera software automatically into a FEP partition file and writes the appropriate Hello.Boot and Boot.Boot files.

Note for existing machines: When upgrading an existing UX machine to Genera 8.1, you need to restore the Genera software by hand; see the section "Restoring the Genera Distribution Worlds From CD-ROM".

4. Completing Genera configuration and restoring the Genera sources.

Genera sources are provided on a pair of QIC-11 cartridge tapes, labeled *Genera 8.1 Sources (All Platforms)*. They are restored onto your file server by the Genera Restore Distribution command. See the section "Restoring the Genera Sources".

Note for new sites: You must create a site definition. See the section "Creating a New Site for Genera".

Note for existing sites: You must update your site definition to include any new UX-family machines. See the section "Registering a New Symbolics UX Machine".

All sites: Finally, you must build a site-configured world and any specially configured worlds you use. See the section "Making a Genera Site-Configured World". See the section "Making Customized Genera User and Server Worlds".

Where Do I Start?

The first step is to have Symbolics install the UX board.

The order of the second and third installation steps depends on whether you are installing a new machine or upgrading an existing machine.

New machine: If you are installing a new UX machine, you must install the UNIX software *before* installing Genera. To proceed: see the section "Installing and Configuring the Symbolics UX UNIX Software".

Existing machine: If you already have a UX machine and are upgrading the software to Genera 8.1, you must install Genera 8.1 *before* installing the UNIX software. To proceed: see the section "Installing the Genera Software on a UX".

Installing and Configuring the Symbolics UX UNIX Software

This section describes how to restore, install, and configure the UX UNIX software, and how to restore the UX Genera software. The UX UNIX software is supplied on a single quarter-inch QIC-11 tape labelled *Symbolics UX UNIX Software for SunOS 4.1, Genera 8.1*, and is in the distribution format expected by the SunOS extract-unbundled program. The UX Genera software is supplied on two quarter-inch QIC-11 tapes in IFS format, labelled *Genera 8.1 Distribution Worlds for UX Family, 1/2, 2/2*.

New machine: If you are installing a new UX machine, you must install the UNIX software *before* restoring Genera sources and installing any layered products.

Existing machine: If you already have a UX machine and are upgrading the software to Genera 8.1, you must install Genera 8.1 *before* installing the UNIX software. See the section "Installing the Genera Software on a UX".

Background Information and Prerequisites for UX Installation

Symbolics UX Configuration Considerations

Before you can load the UX UNIX software onto your UNIX machine, you must configure the UNIX machine properly and gather certain other data. There are additional UNIX configuration requirements for installing Genera, see the section "UNIX Requirements for Installing Genera on a Symbolics UX Machine".

UX life support has the following requirements:

1. You must be running the correct version of the UNIX operating system. Genera 8.1 UX life support runs under SunOS 4.1. If you are not running the proper operating system version, obtain the software from your UNIX vendor and upgrade your machine.
2. You must be running an X server that supports the correct revision of the X protocol. Genera 8.1 supports X version 11 revision 4. If you do not have an X server, contact your UNIX vendor or the Massachusetts Institute of Technology X Consortium. Your Symbolics sales representative can also help you find an X server for your machine. The X server must be running on every UNIX machine from which you plan to access the UX.
3. UX life support needs a large UNIX file or files for use as a FEP partition. You will probably want to create this file in a separate UNIX disk partition. If so, you should reconfigure your UNIX filesystem before starting the UX installation procedure.
4. Genera and the UNIX machine must have separate Internet addresses. These addresses should be registered in your UNIX hosts database (one or more of Network Information Service, the Domain Name System, and the `/etc/hosts` file). Genera will also need a Chaosnet address. See the section "Choosing Ma-

chine Names and Network Addresses". You must choose the network addresses before you install the Genera software.

We recommend that you back up your filesystems before installing this software.

Overview of Symbolics UX Life Support

Genera on the UX relies on UNIX for all its I/O services, including access to tape drives, the network, and a console for its user interface. The UNIX software that provides these services is collectively known as Genera Life Support, because without these services Genera cannot be run. Life support on the UX consists of a device driver and other support code configured into the UNIX kernel, and two UNIX programs. NFS and the X server running on the host UNIX are also part of UX life support, although they're not supplied by Symbolics.

In order to provide life support services, the life support programs need to know the names of various files and devices, such as the FEP partition files or the network interface. These names are provided in a configuration file stored in the UNIX filesystem and read by life support programs as necessary.

UNIX Permissions Considerations for Symbolics UX Life Support

UNIX permissions are used to restrict access to data and capabilities, on a per-user or per-group basis. Users unfamiliar with the UNIX permissions scheme should see Chapter 2, "More About Files", in the Sun document *Doing More with SunOS: Beginner's Guide*.

The following table describes the permissions requirements of the UX400S and UX1200S UNIX software:

<code>/dev/ivoryn</code>	Restricting write access to one of these files restricts access to the particular UX400 or UX1200. <i>n</i> is from 0 to 5 for a UX400 and from 16 to 21 for a UX1200.
<code>create-fep-file</code> , <code>create-fep-partition</code> , <code>install-fep-kernel</code> , <code>write-fep-file</code>	No special permissions required. The user must have read and write access to the FEP partition file(s) in question.
<code>describe-ivory</code>	This program must be able to read <code>/dev/kmem</code> , and will normally be set-gid to the <code>kmem</code> group. The user must have read access to <code>/dev/ivoryn</code> .
<code>genera</code>	This program must be able to read <code>/dev/kmem</code> , and will normally be set-gid to the <code>kmem</code> group. The user must have read and write access to <code>/dev/ivoryn</code> .
<code>halt-ivory</code>	This program must be able to read <code>/dev/kmem</code> , and will normally be set-gid to the <code>kmem</code> group. The user must have read and write access to <code>/dev/ivoryn</code> .
<code>ivory-life</code>	This program must be run by or set-uid to the super-user.

ivory-servicesd This program must be run by or set-uid to the super-user.

read-fep-file, show-fep-directory

No special permissions required. The user must have read access to the FEP partition file(s) in question.

Symbolics UX Disk Partitions

Genera stores its world loads, paging files, FEP programs, and partitions for higher-level file systems in a simple file system called the FEP file system. The FEP file system is a convenient mechanism to organize very large, slowly changing files; it is not a general-purpose file system such as LMFS or the UNIX file system. See the section "FEP File Systems". On Symbolics workstations such as the XL-family machines, exactly one FEP file system is created on each physical disk drive during system installation at the factory, and the resulting FEP file systems are referred to as FEP0, FEP1, ... FEP n , where n is the unit number of the corresponding disk drive.

The Symbolics UX also uses FEP file systems to store its world loads, paging files, and other information. However, its FEP file systems do not necessarily correspond to physical disk drives. Instead, Genera FEP file systems are created inside large UNIX files, called *FEP partition files*. These FEP partition files are ordinary UNIX files; there's nothing special about them except that they tend to be very large and do not contain information that can be manipulated by standard UNIX tools.

A Symbolics UX board needs at least one FEP partition file to boot, but may have up to 24 of them, as specified in the configuration file. See the section "Symbolics UX Configuration File". Genera and the FEP view these partitions as virtual disks, referred to as FEP0, FEP1, and so on just as though they were physical disk drives attached to a Symbolics workstation. From Genera's perspective, it is better to have fewer, larger FEP partitions (ideally one). From the UNIX perspective, though, it may be more convenient to have more, smaller FEP partitions. See the section "How Big Should Symbolics UX Disk Partitions Be?".

FEP partition files are created by the create-fep-partition UNIX program, which is run automatically by the installation program. The create-fep-partition program creates the actual UNIX file to hold the partition, creates the FEP file system within it, and optionally loads a world tape (that is, a tape in IFS format) containing initial Genera worlds and FEP software. Before using the installation program, however, you must decide where in the UNIX file system to create the partitions. See the section "Where Should Symbolics UX Disk Partitions Go?".

How Big Should Symbolics UX Disk Partitions Be?

Genera measures disk space in blocks of 1,280 bytes. This odd measure corresponds to the page size used by the Ivory processor, which is 256 words of 40 bits each. UNIX usually measures disk space in blocks of 512 bytes, or in *kbytes* of 1,024 (not 1,000) bytes. In this section, we will use *block* to refer to a Genera block, and *kbyte* when we wish to discuss the size of a file in the UNIX filesystem.

Basic Genera development worlds tend to be about 45,000 blocks, and delivery worlds about 20,000 blocks. 50,000 blocks of paging space is a reasonable minimum, with typical configurations running around 100,000 blocks. About 10,000 additional blocks are needed for FEP software, incremental site-configured worlds, and other information such as namespace files for standalone sites. More space may be necessary for incremental worlds built with layered products or other large systems.

Ideally, all of this space should be allocated in one large partition file at system installation time. A 120,000 block (150,000 kbyte) file could contain a basic development Genera world, a site-configured incremental world, 50,000 blocks of paging space, and have room left for incidentals. If necessary, other partitions could be added later, in smaller increments, to increase paging space or store new worlds.

If you intend to completely fill a UNIX file system with a FEP partition file, you do not need to do any arithmetic to determine exactly how big to make the file. The `create-fep-partition` program provides information about the maximum size file that fits in the specified partition, and even suggests logging in as the UNIX superuser, when that would allow it to allocate more space.

Where Should Symbolics UX Disk Partitions Go?

UNIX systems tend to divide their disk drives into several partitions, and create separate file systems for each partition: one for `/` (the root file system) and one for `/usr` (the user file system), for example. These partitions are then mounted into a single hierarchy of file names at boot time. Using the NFS file protocol, partitions may be mounted from remote machines via the network.

A FEP partition file may be allocated within any UNIX file system, remote or local, but we strongly discourage the use of remote FEP partitions. We recommend allocating large FEP partition files within a UNIX disk partition reserved only for that purpose. This is mainly because allocating such a large file in a partition used for other purposes can be inconvenient (for example, some backup tools provide no mechanism for excluding a particular file), but also because UNIX file systems are by default tuned for storing many small files. Tuning a UNIX file system for storing a single large file improves performance (slightly) and space efficiency.

If you decide to allocate a dedicated UNIX file system to store a UX FEP partition file, you may need to rearrange the partitions on the disk drive in question, using the `format` program supplied with UNIX and perhaps restoring part of the disk's contents from tape. Once you have a UNIX partition, create a UNIX file system on it using the `newfs` program. If you intend to completely fill the UNIX file system with a single FEP partition file, you may tune the file system using the following options to `newfs`:

- b 8192 specifies a block size of 8192 bytes
- f 8192 disables the fragment feature
- c 32 increases the size of a cylinder group

-i 32768 reduces the amount of space wasted storing inodes

For more information on the `newfs` program, see section 8 of the *SunOS Reference Manual*. For more information on the `format` program, see the chapter "Maintaining Disks with `format`" in the Sun document *System and Network Administration*.

A FEP partition file may be accessed remotely using the NFS mounting abilities of the UNIX file system. However, performance in this configuration is poorer than for other configurations. Also, note that partition files may not be shared among multiple UX systems; there is no interlocking between the systems, and nothing to prevent them from using the same data blocks for two different purposes.

FEP partition files may be located anywhere in the UNIX file hierarchy. We recommend storing them in a top-level directory called `/ivory`, containing files named `fep0`, `fep1`, and so on. These files may be symbolic links to files contained in other file systems, or may be mount points for UNIX file systems containing the actual FEP partition files.

The Symbolics UX Installer Program

To install the UX UNIX software, place the tape labelled *Symbolics UX UNIX Software for SunOS 4.1, Genera 8.1* in a cartridge tape drive of any Sun on the network. Run the SunOS `extract_unbundled` program. This program asks a few questions about where the tape drive is, and then prints a copyright message from the tape. When you are ready, give the go-ahead, and the program loads an installer script from the tape and executes it.

The installer script is provided by Symbolics. It asks a few questions about which portions of the UX UNIX software you wish to install, and computes the required disk space. If the space available is not sufficient, the program asks for a different place to load the software. This is only a temporary resting place until the installation program can properly install the software. When the requested files have been restored, the installer script runs an installer program.

The installer program operates in two passes. In the first pass, it queries the user for configuration options, and determines what actions should be performed. In the second pass, it performs the tasks requested by the user, proceeding from task to task until it finishes or gets an error. The second pass does not require user interaction.

The program is divided into five phases, each of which deals with a different part of the installation. Phases can be skipped if the software necessary to perform them was not restored from tape.

- Phase 1 (UX Software Installation) installs files in the appropriate directories in the UNIX filesystem. These files include architecture-dependent files (object libraries and executable programs), C include files, source files, and IFEP prompts. See the section "Phase 1 (UX Software Installation) of Symbolics UX Installer Program".

- Phase 2 (X Font Installation) compiles and installs the Genera X fonts, for either (or both) of the MIT and Sun Open Windows X servers. See the section "Phase 2 (X Font Installation) of Symbolics UX Installer Program".
- Phase 3 (UNIX Kernel Configuration) configures and compiles a UNIX kernel that can support the desired number of UX boards. This includes updating various source files and copying (architecture-dependent) kernel binaries to the right place.
- Phase 4 (FEP Partition Preconfiguration) creates FEP partition files for the desired number of UX boards, and initializes those partitions from the distribution tape. Phase 4 should be skipped when upgrading life support for a new release of Genera.
- Phase 5 (UNIX Startup Configuration) deals with configuring UNIX, especially its startup procedures, for UX life support. It is usually skipped when upgrading life support for a new release of Genera.

First Pass of Symbolics UX Installer Program

The following sections present a sample dialogue between the user and the installer program. You should run the program as super-user; the UNIX prompt for super-user is #. User input is shown in **bold**. Where the questions below have no visible answer, a RETURN was entered, which accepts the default. Computer output is shown in typewriter font. Explanations about the questions are shown in *italic*.

Note that the dialogue varies from one installation to another, because the answers to some of the questions control which additional questions are asked.

The installation tool offers reasonable defaults for all the questions it asks, and creates directories that don't exist on the fly. You can always ignore the defaults and put things anywhere in the filesystem that seems convenient.

Running extract-unbundled

To install the UX UNIX software, place the tape labelled *Symbolics UX UNIX Software for SunOS 4.1, Genera 8.1* into a cartridge tape drive of any Sun on the network. Run the SunOS `extract_unbundled` program. This program asks a few questions about where the tape drive is, and then prints a copyright message from the tape. When you are ready, give the go-ahead, and the program loads an installer script from the tape and executes it.

Running `extract_unbundled` looks like this. What you type is indicated in bold. Explanations of the dialogue are in italics.

```
% su
Password:
# extract_unbundled
```

Enter media drive location [local | remote]: **local**

Enter Device Name (e.g. rst0, rmt0, rfd0c) : /dev/**rst0**

****Please mount the release media if you haven't done so already.****

Press return when ready:

The following product will be installed:

2+0 records in

2+0 records out

Symbolics UX400S/UX1200S UNIX Software for SunOS 4.1, Genera 8.1

Copyright (c) 1991 Symbolics, Inc. All rights reserved.

Portions of font library Copyright (c) 1984 Bitstream, Inc.

All Rights Reserved.

TRADEMARK NOTICE

Symbolics, Ivory, Symbolics UX400S, and Symbolics UX1200S are trademarks and Genera is a registered trademark of Symbolics, Inc.

UNIX(R) is a registered trademark of AT&T. X Window System is a trademark of the Massachusetts Institute of Technology. Open Windows, Sun, Sun-3, Sun-4, and SunOS are trademarks, and Sun Microsystems is a registered trademark of Sun Microsystems, Inc.

Use the Genera Show Legal Notice command to see important legal notices.

Do you want to continue [y|n]? **y**

/usr/etc/extract_unbundled : Extracting Installation Scripts

tar: blocksize = 20

x ./install_unbundled, 5629 bytes, 11 tape blocks

x ./unbundled_sizes, 82 bytes, 1 tape blocks

/usr/etc/extract_unbundled : Begin Install Script Execution

[At this point, the Symbolics-supplied installation script is run]

Do you wish to restore UX-family sun3 life support? **y**

[On Sun-4s, the script offers to install sun4 life support. "Life support" includes architecture-dependent files such as executable binaries and kernel object files, and architecture-independent files such as IFEP device PROMs.]

Do you wish to restore UX-family life support sources? **y**

[Life support sources are completely optional, except with UX ELS Update #1, where restoring life support sources is required.]

Do you wish to restore the Genera fonts? **y**

[Genera fonts are optional and large, but provide a significant performance

improvement. They need to be installed only once for each X server per version of Genera, assuming that the fonts are shared among all UX400-supporting Suns at the site.]

```
/usr/tmp/unbundled/install_unbundled: Restoring files
x ./sun3/etc/test-ivory, 139264 bytes, 272 tape blocks
x ./ifep/solstice-device.prom, 130717 bytes, 256 tape blocks
x ./src/Makefile, 58127 bytes, 114 tape blocks
x ./src/autoboot.c, 4193 bytes, 9 tape blocks
[many other files here ...]
```

```
/usr/tmp/unbundled/install_unbundled: Running installation tool
```

Phase 1 (UX Software Installation) of Symbolics UX Installer Program

The sample dialogue for Phase 1 is:

```
Do you wish to install the UX400S/UX1200S software (default Yes)?
The ivory-life program has to be run by root, the super-user.
Do you wish ivory-life to be installed setuid to root (default Yes)?
Certain services provided by the ivory-servicesd program require that it run as root.
Do you wish ivory-servicesd to be installed setuid to root (default Yes)?
Several of the UX400S/UX1200S life-support programs want to be able to read /dev/kmem.
Do you wish these programs to be installed setgid to kmem (default Yes)?
Where should application programs go (default /usr/bin)?
[You will need to know the answer to this question for a later installation
step, so make a note of it. If you get a warning message, see the section
"Setting the UNIX PATH Variable for a Symbolics UX Machine"
page !
.]
```

```
Where should maintenance and installation programs go (default /usr/etc)?
Where should include files go (default /usr/include/ivory)?
Where should libraries go (default /usr/lib)?
Do you wish to install sources to the UX400S/UX1200S software (default Yes)?
Where should sources go (default /usr/share/symbolics/src)?
Where should IFEP prompts go (default /usr/share/symbolics/ifep)?
```

These questions control the copying of UX400S software from tape into the appropriate places in the UNIX file system.

To proceed to the next installation step, see the section "Phase 2 (X Font Installation) of Symbolics UX Installer Program".

For general information about Ivory life support, See the section "Overview of Symbolics UX Life Support".

For information about permissions considerations of the ivory-life and ivory-servicesd programs, see the section "UNIX Permissions Considerations for Symbolics UX Life Support".

Phase 2 (X Font Installation) of Symbolics UX Installer Program

Do you wish to install the Genera fonts for the Sun Open Windows X server (default Yes)?
 Where do the Open Windows programs live (default /usr/openwin)?
 Where should the Open Windows Genera fonts go (default /usr/openwin/lib/genera-fonts)?

WARNING: directory /usr/openwin/lib/genera-fonts is not in the search path defined by environment variable FONTPATH. Make sure to update FONTPATH to include /usr/openwin/lib/genera-fonts.

[If you get this warning message, see the section "Setting the UNIX PATH Variable for a Symbolics UX Machine" page !
]

Do you wish to install the Genera fonts for the MIT X server (default No)?

These questions determine whether the fonts for Sun Open Windows X and/or the M.I.T. X server will be compiled. Installing fonts for the X server is optional but provides a significant performance improvement. If you omit this phase, text drawn by Genera may look peculiar.

Phase 3 (UNIX Kernel Configuration) of Symbolics UX Installer Program

Do you wish to build a UNIX kernel with support for the UX400S/UX1200S (default Yes)?
 What kernel architecture does this machine use (default sun4)? **sun4m**
 How many UX400S boards should this kernel support (default 1)? **0**
 How many UX1200S boards should this kernel support (default 1)?
 Where is your kernel build directory (default /usr/share/sys)?
 What kernel usually runs on this machine (default GENERIC)?

The correct answer to the question about kernel architecture for all Sun 4/6xx models is sun4m. For all other Sun 4/xxx models, the correct answer is sun4.

The installation program writes a UNIX kernel configuration file and compiles the resulting kernel based on the answers to these questions. It automatically updates the various files that need to be updated for UX kernel support.

Phase 4 (FEP Partition File Configuration) of Symbolics UX Installer Program

Do you wish to create FEP partition file(s) (default Yes)?
 Do you wish to create a FEP partition file for UX1200S device #16 (default Yes)?
 Where do you wish to create the FEP partition file (default /ivory/fep16)?
 There are 279685 kbytes available for this file.
 How big do you wish the partition to be (default 150000)?
 Do you wish to create a Hello.boot file (default Yes)?
 What is the UX400S/UX1200S's Internet address? **128.81.20.125**
 What is the UX400S/UX1200S's Chaosnet address? **12175**

You should replace the network addresses in bold with the network addresses you have chosen for the machine. For information on choosing network addresses, see the section "Choosing Machine Names and Network Addresses".

Note that if you are installing more than one UX board in a single UNIX, you will be asked these questions for each board.

For information about FEP partition files, see the section "Symbolics UX Disk Partitions".

To proceed to the next installation step, see the section "Phase 5 (UNIX Startup Configuration) of Symbolics UX Installer Program".

Phase 5 (UNIX Startup Configuration) of Symbolics UX Installer Program

Do you wish to install the UX400S/UX1200S kernel (default Yes)?
[Note that if no kernel was built in Phase 3, a question about which kernel to install is asked here.]

Do you wish to create the UX400S/UX1200S configuration file(s) (default Yes)?
 Do you wish UX1200S #0 to use network interface ie0 (default Yes)?
[This question is asked once for each board being installed.]

Do you wish to create/update the UX400S/UX1200S device files in /dev (default Yes)?
 Do you wish /etc/rc.local to start UX400S/UX1200S life support (default Yes)?
[We recommend answering yes to this question, see below.]

Do you wish to reboot UNIX after installation (default Yes)?

Each Ivory device driver has associated with it a major device number identifying a particular device file (such as /dev/ivory0) to the kernel. The kernel associates that file with the given Ivory device driver. Because Sun adds supported devices with each SunOS release, the major device numbers of non-Sun device drivers (such as the UX) change with every release. The expected major device number of the UX is 63 in SunOS 4.0.1, 68 in SunOS 4.0.3, 72 in SunOS 4.0.3c, 104 in SunOS 4.1 and 4.1.1, and 105 in SunOS 4.1.2.

If you are installing the software but not building a kernel (presumably you already built a kernel previously on some other machine at your site, and will ask the installer to copy it onto your machine), you need to know the major device

number of the Ivory device driver in that kernel. The major device number is printed out during the second pass of the installer. If you have forgotten the number, you can find it by editing the file `/sys/sun/conf.c`. Search for the entry that looks like:

```
{
    ivoryopen,    ivoryclose,    ivoryread,    ivorywrite, /*NN*/
    ivoryioctl,  nodev,        ivoryselect,  ivorymmmap,
    0,
}
```

The number *NN* in the comment is the major device number.

We recommend answering Yes to the question about whether `/etc/rc.local` should start UX life support. If life support is started automatically by `/etc/rc.local`, you should not also try to start it manually, because that would cause multiple copies of `ivory-serviced` and `ivory-life` to run. Running multiple copies of `ivory-life` for a single UX device is dangerous and can cause UNIX kernel crashes.

Second Pass of Symbolics UX Installer Program

During the second pass of the installer program, if you are installing from tape, you need to put in the first tape labeled *Genera 8.1 Distribution Worlds for UX Family*, or install from CD-ROM. If you are installing from CD-ROM, the UX installation tool does not ask if the Distribution World tape is mounted because tapes are not necessary with CD-ROM installation.

About to begin UX400S/UX1200S software installation, this will probably take a while.
Okay (default Yes)?

Is the Genera Distribution World tape mounted (default Yes)?

```
Updating application programs... Updated 3 files.
Updating maintenance and installation programs... Updated 10 files.
Updating include files... Updated 2 files.
Updating libraries... Updated 1 file.
% mkdir /usr/share/symbolics
% mkdir /usr/share/symbolics/src
Updating sources... Updated 78 files.
% mkdir /usr/share/symbolics/ifep
Updating IFEP proms... Updated 1 file.
```

```
Installing Genera fonts for Open Windows... Updated 224 files.
% cd /usr/openwin/lib/genera-fonts; /usr/openwin/bin/bldfamily -f 100
* genera-vt100      ./generavt100.ff (Encoding: unknown)
* genera-tvfontr   ./generatvfontr.ff (Encoding: unknown)
[many other fonts here ...]
```

```
Updating kernel binaries... Updated 11 files.
```

```

Configuring UX400S/UX1200S kernel...
  [Updated /usr/share/sys/sun/conf.c; Ivory major device number is 104]
  [Updated /usr/share/sys/sun/str_conf.c]
  [Updated /usr/share/sys/conf.common/files.cmn]
  [Created /usr/share/sys/sun3/conf/UX1200S from /usr/share/sys/sun3/conf/GENERIC]
% cd /usr/share/sys/sun3/conf; /etc/config UX1200S
Doing a "make depend"
  [Configured UX400S/UX1200S kernel]
% cd /usr/share/sys/sun3/UX1200S; make
cc -fsoft -c -O -Dsun3 -DUX1200S -DSUN3_E -DSUN3_260 -DSUN3_160 -DSUN3_110 -DSUN3_60 -DSUN3_50
WINSVJ -DOLDSCSI -DVDDRV -DASYNCHIO -DLWP -DVFSSTATS -DRFS -DCRYPT -DTCPCDEBUG -DIPCshmEM -DIPCSEMAPHORE -DIPCMESS
E -DSYSAUDIT -DSYSACCT -DHSFS -DTMPFS -DTFS -DLOFS -DNFSSERVER -DNFSCLIENT -DUFSS -DQUOTA -DINET -DKERNEL -I. -I..
I../.. ../netinet/in_proto.c
  [many other files here ...]
  [Compiled UX400S/UX1200S kernel]
% /var/tmp/unbundled/sun3/etc/create-fep-partition -automatic 153600000 yes 12175 128.81.20.125
ivory/fep16 /dev/rst0
Creating 146 megabyte file /ivory/fep16...
This will take a few minutes (about 9, I'd guess).
Created file /ivory/fep16, stored in 18761 Unix blocks (8192 bytes).

Creating initial FEP file system in partition file /ivory/fep16...
Creating paging file Paging-1.Page.1 (30000 blocks)...
Creating paging file Paging-2.Page.1 (20000 blocks)...

Restoring I325-KERNEL.FEP.351 (192 blocks)...
Installing I325-KERNEL.FEP.351 as the FEP kernel to boot on this partition...
Restoring I325-LOADERS.FLOD.222 (42 blocks)...
Restoring I325-LISP.FLOD.145 (17 blocks)...
Restoring I325-INFO.FLOD.157 (11 blocks)...
Restoring I325-DEBUG.FLOD.158 (81 blocks)...
Restoring I325-DISK.FLOD.193 (43 blocks)...
Restoring I325-TESTS.FLOD.144 (10 blocks)...
Restoring Genera-8-1.ilod.1 (40447 blocks)...
End of tape reached, file Genera-8-1.ilod.1 continues on next tape.

Is the next tape in place and are you ready to continue? (yes or no) yes

Still restoring Genera-8-0-2-Network.ilod.1 (18361 blocks remaining)...
Restoring Network-from-Genera-8-1.ilod.1 (10711 blocks)...
Restoring Genera-8-1-UX-Delivery.ilod.1 (18516 blocks)...
End of tape encountered.
Creating Hello.Boot and Boot.Boot files...

Installing kernel as /vmunix... Done.
Writing /etc/ivory16.conf...
Updating Ivory device files... Done.

```

Updating /etc/rc.local...

Rebooting...

Final UNIX Configuration Steps for the Symbolics UX Machine

Setting the UNIX PATH Variable for a Symbolics UX Machine

In Phase 1 (UX Software Installation) of the installation process, you specified a directory where UX400 application programs should be installed. (See the section "Phase 1 (UX Software Installation) of Symbolics UX Installer Program".) In order to use these programs easily, you should add this directory to your program search path, as defined by the UNIX environment variable PATH. The easiest way to do this is to add a line to the file `.login` in your home directory.

If you use the `cs` shell, add the following line to your `.login` file:

```
setenv PATH /usr/local/bin:$PATH
```

If you use the `sh` shell, add the following line to your `.login` file:

```
PATH=/usr/local/bin:$PATH; export PATH
```

Font Configuration for the X Servers

The fonts are compiled automatically; see the section "Phase 2 (X Font Installation) of Symbolics UX Installer Program". Once they are compiled, you must add the Genera fonts directory to your X server's font path.

Configuring Fonts for the SUN Open Windows X Server

The Sun Open Windows X server uses the font path specified in the UNIX environment variable `FONTPATH`. To make the Genera fonts accessible, you must add to this variable the name of the font directory you specified in Phase 2. (See the section "Phase 2 (X Font Installation) of Symbolics UX Installer Program".) The easiest way to do this is to add a line to the file `.login` in your home directory.

If you use the `cs` shell, add the following line to your `.login` file:

```
setenv FONTPATH /usr/openwin/lib/fonts:/usr/openwin/lib/genera-fonts
```

If you use the `sh` shell, add the following line to your `.login` file:

```
PATH=/usr/openwin/lib/fonts:/usr/opewin/lin/genera-fonts
```

Configuring Fonts for the M.I.T. X Server

For the M.I.T. X server, the font path is specified by the `-fp` command-line option. The M.I.T. X Server can be started via `xinit`, `xdm`, or by running the X server directly from the `/etc/rc` script. `xinit` and `/etc/rc` use one syntax, and `xdm` a slightly different syntax. For more information on starting the X server, see the X documentation, in particular the UNIX manual pages on `xdm`, `xinit`, and `xterm`.

Most installations will not already be specifying a font path, since the default includes all the distributed fonts. You will have to specify all the default font directories as well as the Genera font directory.

In the syntax used in `xinit`'s `.xserverrc` files and `/etc/rc`, this looks like:

```
/usr/bin/X11/X :0 -fp /usr/lib/X11/fonts/75dpi,
                /usr/lib/X11/fonts/100dpi,/usr/lib/X11/fonts/misc,
                /usr/lib/X11/fonts/genera
```

The `.xserverrc` file usually lives in each user's home directory.

In the syntax used in `xdm`'s `Xservers` file, this looks like:

```
:0 local /usr/bin/X11/X :0 -fp /usr/lib/X11/fonts/75dpi,
                /usr/lib/X11/fonts/100dpi,/usr/lib/X11/fonts/misc,
                /usr/lib/X11/fonts/genera
```

The `Xservers` file usually lives in the directory `/usr/lib/xdm`.

Note: In both examples, lines are broken for readability only.

Network Configuration for the Symbolics UX Machine

Network configuration for the Symbolics UX involves enabling access from the UX coprocessor to UNIX network services provided by the UNIX hosts and file servers.

Providing UNIX NFS File Access to a Symbolics Computer

In order for Genera to mount UNIX file systems through NFS, the file systems must be exported by a UNIX file server host. Genera must be able to mount user file systems in order to access user directories.

If you are not using Network Information Service at your site, Genera needs access to some files in the `/etc` directory of UNIX server hosts, so you need to export the root filesystem of these hosts. Network RPC tries to read the `/etc/passwd` and `/etc/group` files for the UNIX name lookup services it provides. The NFS system tries to read the `/etc/mtab` file on UNIX file servers in order to determine mount points.

Edit the `/etc/exports` file and use `exportfs` to allow NFS access to the UNIX filesystems. See the UNIX manual pages for the `exports` file and the `exportfs` program for more information.

Providing X Server Access to a Symbolics UX Coprocessor

This section applies only when the X server is not the UNIX in which the UX board is embedded.

Because Genera needs access to the X display for its console screen, the X server needs to be told to allow access from the UX coprocessor. Since X server access control only lasts for the duration of an X session, you will have to do this each time you log into your UNIX machine.

In the M.I.T. X Window System distribution, access can be enabled with the `xhost` UNIX program. Use the `xhost` UNIX program to tell the X server to allow the UX coprocessor to open connections to the display. (See the UNIX manual page for the `xhost` UNIX program for more information.) For example:

```
% xhost +hostname
```

In Sun Open Windows, the `newshost` program is used to enable access. (See the UNIX manual page for the `newshost` UNIX program for more information.) For example:

```
% newshost add hostname
```

Verifying the Symbolics UX Software Installation

This section describes how to verify that the software has been correctly installed by booting Genera. If you run into trouble, see the section "Troubleshooting Symbolics UX Startup". For more information on booting and halting the UX, see the section "Controlling the Symbolics UX".

Booting the Symbolics UX Machine

The `genera` program is used to boot the UX. See the section "Getting Started with the Symbolics UX". See the section "Controlling the Symbolics UX".

```
% genera
Starting Ivory on /dev/ivoryn
The Device PROM is loading on /dev/ivoryn
```

At this point, an X display appears, connected to the cold-load stream and the FEP. The `genera` program continues running until it prints:

```
The FEP is running on /dev/ivoryn
```

The cold-load X display looks something like this:

```
Type "Hello" to initialize the FEP's command databases, etc
FEP Command:
```

To boot Genera, type the following commands to the FEP:

```
FEP Command: Hello
FEP Command: Boot
```

See the section "FEP Commands".

Genera boots, and the `genera` program prints:

Lisp is running on `/dev/ivoryn`

An X display is created with the Genera console presented in it.

After Installing the Symbolics UX Software on UNIX

Once you have successfully booted Genera, you are finished with installing the UNIX part of the Symbolics UX software.

- **New machines:** If you are installing a new UX machine, you should now install Genera 8.1. See the section "Installing the Genera Software on a UX".
- **Existing machines:** If you are upgrading the software of an existing UX machine, you should have already installed Genera 8.1 world, and you are finished with the entire installation procedure.

Troubleshooting Symbolics UX Startup

This section provides some simple troubleshooting tips for use should your UX fail to operate correctly. If the problem persists, contact Symbolics Customer Service.

If the problem involves UX networking, see the section "Troubleshooting UX Network Problems".

- When you try to run `ivory-servicesd`, you get the following UNIX error:
`udp_agent_open: Address already in use.`

Analysis: This means that `ivory-servicesd` is already running. Probably in Phase 5 of the installer program (in the most recent installation run or in a previous installation run), you answered Yes to the question about whether `/etc/rc.local` should start UX life support. If life support is started automatically by `/etc/rc.local`, you should not also try to start it manually, because that would cause multiple copies of `ivory-servicesd` and `ivory-life` to run. Running multiple copies of `ivory-life` for a single Ivory device is dangerous and can cause UNIX kernel crashes.

Corrective actions: Don't try to run `ivory-servicesd` or `ivory-life` manually, if they are already started automatically, in an rc file.

- The `ivory-life` program (or one of the other life support utilities) fails with the error `/dev/ivoryn: No such device or address`. Run the `dmesg` program (see the *SunOS Reference Manual*, section 8), and look for the line `ivoryn at vme32d32 0xfxxx0000 vec 0xea` in the last boot sequence. If it's there, the kernel is finding your UX hardware; see the next bullet. If it's not there, the kernel is not finding your UX hardware.

Analysis: This problem indicates a problem with the UNIX kernel, or perhaps with the UX hardware.

Corrective actions:

1. Check that the UNIX kernel is configured with UX life support. Make sure that the kernel identification (printed in parentheses on the same line as the version of UNIX) is the same as the one you configured with UX life support. Check that the kernel configuration file includes a device line for `ivoryn`, and that the correct addresses are given. See the section "Phase 5 (UNIX Startup Configuration) of Symbolics UX Installer Program".
 2. Check that the UX hardware is present and securely seated in the VME backplane.
 3. Check that the UX hardware is operating correctly. Power-cycle the UNIX workstation while watching the LEDs on the UX board. The Fault and U0 LEDs should appear for a few seconds, and then the Fault and Run LEDs. See the section "Controls and LEDs on the UX Ivory Main Board".
- The `ivory-life` program fails with the error `/dev/ivoryn: No such device or address`, but `dmesg` shows that the kernel is finding the UX board.

Analysis: This problem occurs when the `/dev/ivoryn` file does not have major and minor device numbers corresponding to the UX device driver in the booted kernel.

Corrective action: Check that `/dev/ivory0` was created with the correct major and minor device numbers, using the `ls` program:

```
% ls -l /dev/ivoryn
crw-rw-rw- 1 root      m,   n Mar  7 16:54 /dev/ivoryn
```

`M` should be the number `NN` from the kernel configuration procedure. See the section "Phase 5 (UNIX Startup Configuration) of Symbolics UX Installer Program". If either number is wrong, you must delete the file and recreate it. See the section "Phase 3 (UNIX Kernel Configuration) of Symbolics UX Installer Program".

- The `ivory-life` program fails with a network interface error, and prints: *NIOCBIND: No such device or address.*

Analysis: The Symbolics UX configuration file has an incorrect value for the interface command. On Sun-3s and older Sun-4s, the network interface is usually `ie0`. On SPARCstations and SPARCservers, the network interface is usually `le0`.

Corrective action: Edit the UX configuration file (usually `/etc/ivoryn.conf`), fixing the line `interface`. Restart `ivory-life` by rebooting the Sun. See the section "Symbolics UX Configuration File".

- The UNIX kernel prints spurious level 2 interrupt on the console.

Analysis: This is due to a bug in the Sun-3 SCSI interface hardware.

Corrective action: Sun reports that the problem has been fixed in later revisions of the hardware.

- The genera program fails with error `/dev/ivoryn: Embedding communication area clobbered`.

Analysis: This occurs when the `ivory-life` program is not running.

Corrective action: Start `ivory-life` by rebooting the Sun.

- The genera program or the `halt-ivory` program prints out `WARNING: Can't verify kernel life-support version numbers`.

Analysis: This problem occurs when the program can't read data from the kernel in `/vmunix`, usually because the program is not `set-gid` to the `kmem` group.

Corrective action: Fix the protections on the program. See the section "UNIX Permissions Considerations for Symbolics UX Life Support".

- The genera program or the `halt-ivory` program prints out `WARNING: Kernel life-support version a.b.c.d does not match application life-support version e.f.g.h!`

Analysis: This problem occurs when you have mismatched life support software. The program comes from a different release of Genera than the life support code configured into the kernel. Other, more serious problems are likely to be caused by this mismatch.

Corrective action: Install a coherent, matched set of life support software.

- The genera program prints the warning "Unable to allocate a modifier for the Hyper key." (The warning may even be printed twice.)

Analysis: This warning usually occurs when running the Open Windows X server. The situation is harmless, although you will be unable to use the Hyper key in the cold load stream. (You can use the `halt-ivory` UNIX program in cases where you would normally press `Hyper-Control-Function` in the cold load stream.)

Corrective action: None necessary.

- Genera starts to boot, but the Open Windows X Server prints an error:

```

Process: 0x2517d8 (X Listener %Socket16000) Error: stackunderflow
Stack:
Executing: 'pop'
At: {packedarray{4} 'stepped' 'pop' *'pop' 'closefile' 'dup' (r) 'file'}
In: {... 4 'index' append 'put' packedarray{8} *'loop'}

```

The genera UNIX program also prints an error:

```

>>Attempt to get data from broken TCP connection to
  UNIX-HOST 6000 (:X-WINDOW-SYSTEM), local port 1025.

```

Analysis: This is a bug in the Open Windows X Server, that occurs when the UX's name and Internet address are not listed in the UNIX hosts database.

Corrective action: Update the UNIX hosts database (usually either the /etc/hosts file or the hosts NIS map) with the UX name and Internet address. Exit and restart the Open X-Windows Server.

- The genera program runs without error, and no notifications appear in the cold load stream, but the Genera console window doesn't appear.

Analysis I: This problem sometimes occurs because the ivory-life program is run by a user other than the super-user.

Corrective action I: Kill all four ivory-life processes, and restart ivory-life as the super-user.

```

% su
Password:
# /usr/share/local/etc/ivory-life

```

If ivory-life was run by the super-user, check that the program is not set-uid to some user other than the super-user.

Analysis II: This problem sometimes occurs because Genera doesn't know how to route Internet packets to UNIX.

Corrective action II: Genera and UNIX must have Internet addresses on the same subnet. Boot Genera with an appropriate Internet address.

- Genera starts to boot, but gets an error in the cold-load stream: You must have IP-TCP loaded to boot a UX. The Genera console window doesn't appear.

Analysis: This problem occurs when you try to boot a Genera world that doesn't have the IP-TCP system loaded.

Corrective action: Boot a world with IP-TCP loaded. If you are booting the Genera-8-1 distribution world, boot the Network-from-Genera-8-1 distribution IDS instead.

- Genera starts to boot, but gets an error in the cold-load stream: You must specify an Internet address to boot a UX. The Genera console window doesn't appear.

Analysis: This error occurs when you try to boot a UX without an Internet address.

Corrective action: Add an Internet address to the Set Network Address command in your Hello.Boot file. The Internet address should come after the Chaos address. To boot the UX with a network address different from the one specified in its Hello.Boot file, use the Set Network Address FEP Command:

```
FEP Command: Hello
[Output from Hello.Boot file, including Set Network Address command]
FEP Command: Set Network Address Chaos|nnnnn,Internet|a.b.c.d
FEP Command: Boot
```

See the section "FEP Commands". Once Genera has booted, you can edit the Hello.Boot file and correct its Set Network Address command.

- Genera starts to boot, but prints a notification in the cold-load stream about services missing from DIS-LOCAL-HOST. The Genera console window doesn't appear.

Analysis: This problem occurs when you have assigned the same Internet address to both the UNIX and Genera machines. Genera and UNIX must have different Internet addresses.

Corrective action: Assign a different Internet address to each host. To boot the UX with a network address different from the one specified in its Hello.Boot file, use the Set Network Address FEP Command:

```
FEP Command: Hello
[Output from Hello.Boot file, including Set Network Address command]
FEP Command: Set Network Address Chaos|nnnnn,Internet|a.b.c.d
FEP Command: Boot
```

See the section "FEP Commands". Once Genera has booted, you can edit the Hello.Boot file and correct its Set Network Address command.

- Genera starts to boot, but the genera UNIX program prints an error:

```
>> Connection failure to X11.0 server UNIX-HOST display 0:
Client is not authorized to connect to Server
```

The Genera console window doesn't appear.

Analysis: Genera is trying to create an X window, but access control is enabled on the X server and the UX is not listed in the access control list.

Corrective action: Disable access control, or add the UX host to the access list. See the section "Providing X Server Access to a Symbolics UX Coprocessor".

- Genera starts to boot, but the genera UNIX program prints an error:

```
>> X-SERVER-HOST (at Internet address e.f.g.h) did not
respond to a 1042->6000(:X-WINDOW-SYSTEM) request.
```

Either the destination does not support that protocol or some gateway could not determine a route to the destination.

The Genera console window doesn't appear.

Analysis: Genera is trying to create an X window, but doesn't know how to route Internet packets to X-SERVER-HOST.

Corrective action: Genera and X-SERVER-HOST must have Internet addresses on the same subnet. Boot Genera with an appropriate Internet address.

- Genera boots, but the herald shows SunOS (??) ?? rather than SunOS (*ident*) *version*, where *ident* will usually be UX and *version* will usually be 4.1.

Analysis: This problem occurs when the *ivory-servicesd* program is not running, or when that program can't read data from the UNIX kernel binary in */vmunix*.

Corrective action: As super-user, start the *ivory-servicesd* program, or grant it read access to */vmunix*.

```
% su
Password:
# /usr/share/local/etc/ivory-servicesd
```

- F1 on your keyboard does not operate as SELECT.

Analysis: This occurs when you are running under Open Windows, which reassigns the F1 key.

Corrective action: Probably your default *.xinitrc* contains:

```
xmodmap -e 'keysym F1 = Help'
```

You can comment out that line prior to running *openwin*.

The file in question is probably named */usr/openwin/lib/Xinitrc*, and/or the same thing could happen in the user's home directory: *~/.xinitrc*. Commenting out that line with a "#" fixes the problem.

```
# xmodmap -e 'keysym F1 = Help'
```

- Genera thinks you have the wrong keyboard, and just beeps when you press special keys such as the HELP key.

Analysis: SunOS 4.0, 4.0.1, and 4.0.3 kernels incorrectly report that the keyboard type is Type-3 when a Type-4 keyboard is attached. If you have a Sun Type-4 keyboard and Genera does not recognize it as such, it is because of this problem.

Corrective actions:

1. Upgrade to SunOS 4.1.
2. Make your X server work around the problem for all X applications, by altering the source code somehow (if you have the sources to your X server).
3. Use the KEYBOARD-LAYOUT-TYPE user property (described below) on the namespace host object for the Sun to identify the keyboard.

keyboard-layout-name *display-number layout-type*

A user property for host namespace objects informs Genera what the layout type for the keyboard for the specified display on the host. If this user property is present, it overrides the layout type inferred from the assigned X keysyms.

The valid layout types are SUN-TYPE-3 and SUN-TYPE-4.

- When Genera tries to use the UNIX file system, it fails with a Permission denied error.

Analysis: Genera is trying to mount a UNIX filesystem, but the UNIX NFS mount server isn't granting Genera access to the filesystem.

Corrective action: Edit the `/etc/exports` file to grant access to Genera. Run the `exportfs` program to apply the changes made to the `/etc/exports` file. See section 5 of the *SunOS Reference Manual* for more information on `/etc/exports`, and section 8 for more information on `exportfs`.

Troubleshooting UX Network Problems

This section presents an overview of how networking on the UX works, and some techniques for debugging network problems.

Overview of UX Networking

The UX network is implemented by a Network channel in the embedded communication area (shared memory between Genera and UNIX). All network packets Gen-

era wishes to transmit are placed in the network channel by Genera, and are removed and placed on the network by UX life support software running in the UNIX kernel. All network packets Genera receives are placed in the network channel by UX life support, and are removed and processed by Genera.

Life support arranges to receive all packets sent to the Sun's Ethernet address, via a SunOS facility known as the Network Interface Tap (NIT). IP packets with the Sun's IP address are filtered out, and all other packets are placed in the network channel. Note that Genera and UNIX are sharing an Ethernet address, but are using separate protocol (that is, IP) addresses.

Transmitting packets is trickier. When life support reads an outgoing packet (that is, a packet that Genera wishes to transmit) from the network channel, it examines the packet's Ethernet address. Packets bearing the Sun's Ethernet address are routed to a special UNIX network interface that is used to connect Genera and UNIX. Packets bearing other Ethernet addresses are placed on the Ethernet, using NIT. Packets bearing the Ethernet broadcast address are routed to the special interface and placed on the Ethernet.

Note that there are two different paths for UX network packets: one that connects Genera and its embedded host, and another that connects Genera with all other hosts on the network. Either one of these can potentially fail, and it's important to keep the two in mind when trying to debug UX network problems.

UX Network Debugging Techniques: ping and telnet

ping is a UNIX program that sends an ICMP echo packet to a specified host. If the host does not respond after several attempts, ping will report that the host is down; otherwise, ping will report that the host is up. ping can be used to determine whether Genera and its network system are up. Remember that if things are broken, ping can give a different answer when run on the UNIX machine in which the UX is embedded than when it is run on some other host.

Successful ping:

```
% ping coppertone
coppertone is alive
```

Unsuccessful ping:

```
% ping equinox
no answer from equinox
```

telnet is a UNIX program that opens a Telnet network connection to a specified host. telnet to a UX can fail even when ping works, for any of the following reasons:

1. Genera's network system is not completely enabled yet.
2. Genera doesn't trust the telnetting UNIX host, due to use of the Namespace secure subnets feature.

3. Genera's services are disabled.
4. Genera has remote login turned off. See the function **net:remote-login-on**.

Note that conditions 2, 3, and 4 all are true for distribution worlds.

Unsuccessful telnet:

```
% telnet coppertone
Trying 128.81.20.123 ...
telnet: connect: Connection refused
telnet>
```

Connection refused means that Genera is up, but would not allow the Telnet connection to be opened. Conditions 2, 3, or 4 above can cause this. telnet might also report *Connection timed out*: this means that Genera never responded to the telnet request. Condition 1 can be caused by Genera not running, some kind of network breakage, or the wrong IP address for the Genera host name.

UX Network Debugging Technique: netstat

netstat is a UNIX program for poking into the internals of the UNIX network system. It can be used to examine the special UNIX network interface mentioned above. The `-I` option to netstat is used to examine interface structures; its output looks like this:

```
% netstat -I emb0
Name Mtu Net/Dest Address Ipkts Ierrs Opkts Oerrs Collis Queue
emb0 1500 coppertone sage 1461556 0 1060318 61 0 0
```

coppertone is the name of the UX; *sage* is the name of the UNIX. If either of these shows up as an IP address rather than a name, something is wrong with the UNIX hosts database. The large number of packets, both Genera->UNIX (Ipkts) and UNIX->Genera (Opkts) can be explained by constant X traffic (blinking the cursor and the colon in the status line) for the one day that this interface has been up. The Oerrs are due to packets dropped, usually because the `recv guest->host` queue (see the section "UX Network Debugging Technique: describe-ivory") emptied temporarily during heavy network traffic.

Note that the embedding interface(s) are not created until the Genera network system comes up.

UX Network Debugging Technique: describe-ivory

The network channel can be examined via the `describe-ivory` program. `describe-ivory` dumps the complete, current state of the embedded communication area; only the network channel is described here. `describe-ivory` is distributed as part of UX life support, and is normally installed in `/usr/local/bin`.

The relevant portion of `describe-ivory`'s output looks like this:

```

@561: type 3 (Network), unit 0, emb0, status 2
  interface ie0, hw address 8:0:20:0:0:d6
  Host IP address 128.81.20.42, guest IP address 128.81.20.123
  323 crc/align errors, 13 aborts, 267 coll
  xmit guest->host queue @619: 20 4-byte elements (empty), signal h1
  xmit host->guest queue @593: 20 4-byte elements (1 element), no signal
  recv guest->host queue @671: 20 4-byte elements (10 elements), signal h2
  recv host->guest queue @645: 20 4-byte elements (1 element), signal g1

```

The channel header:

- 561 is the address of the network channel in the embedded communication area. Network is the type of the channel; other channel types have different formats. Unit is the unit number of the channel, emb0 is the name of the corresponding special UNIX interface, and status is a flag word. If status is 2, Genera is listening to this network channel.
- ie0 is the name of the UNIX network interface corresponding to this network channel. Packets transmitted by Genera through this channel (other than those bearing the Sun's Ethernet address) will be transmitted on UNIX's ie0 interface; packets received on UNIX's ie0 interface (other than IP packets bearing the Sun's IP address) will be placed in this network channel for eventual reception by Genera. 8:0:20:0:0:d6 is the Ethernet address of UNIX's ie0 interface.
- 128.81.20.42 is UNIX's IP address on this interface. 128.81.20.123 is Genera's IP address on this interface. The crc/align(ment) errors, aborts, and coll(isions) meters are copied from the UNIX ie0 interface.
- When Genera is not yet booted, or while it is booting but before its network system is up, the Network channel might look like this (differences in bold):

```

@561: type 3 (Network), unit 0, no emb if, status 0, uninitialized
  interface ie0, hw address 8:0:20:0:0:d6
  Host IP address 128.81.20.42, no guest address
  0 crc/align errors, 0 aborts, 0 coll
  xmit guest->host queue @619: 20 4-byte elements (empty), signal h1
  xmit host->guest queue @593: 20 4-byte elements (1 element), no signal
  recv guest->host queue @671: 20 4-byte elements (10 elements), signal h2
  recv host->guest queue @645: 20 4-byte elements (1 element), no signal

```

The queues:

- When Genera wants to transmit a packet, it places it in the xmit guest->host queue. UX life support processes the packet, then returns it in the xmit host->guest queue. Returning the packet acts as an acknowledgement that the packet was transmitted.
- If the xmit guest->host queue has more than a few elements in it, that is an indication that the network portion of life support is either backed up or dead. If

the condition persists (and Genera isn't doing something network-intensive, such as copying a world), network life support is probably dead. Check the `ivory-net` process (one of the children of `ivory-life`): if it is dead, probably you will have to crash Genera into the FEP, kill the `genera` program and all four `ivory-life` processes, restart `ivory-life` by rebooting UNIX, run `genera`, give the `Hello` command to the FEP, and finally warm boot Genera.

- When life support receives a packet and decides to route it to Genera, it removes one of the buffers waiting in the `recv guest->host`, copies the packet into it, and places the buffer into the `recv host->guest` queue. The Genera Ethernet driver reads the packet out of the buffer, and makes sure that there enough buffers available in the `recv guest->host` queue.
- If the `recv guest->host` queue remains empty for more than a short time, the Genera Ethernet driver has probably broken somehow (again, assuming that Genera is not doing something network-intensive). Your only hope of recovering is probably to warm boot Genera. Similarly, if the `recv host->guest` queue backs up under normal network load, the Genera Ethernet driver is not working, and your best hope of recovering is to warm boot Genera.

Symbolics UX-family UNIX Programs

Symbolics UX Life Support Programs

`ivory-servicesd` UNIX Program

`ivory-servicesd`

The `ivory-servicesd` program provides miscellaneous RPC services to Genera. `ivory-servicesd` is usually invoked during the UNIX system startup process. Note that `ivory-servicesd` may be started before the X server.

`ivory-servicesd` must run as root, so that it can provide certain functionality (via RPC) to Genera. However, in general, `ivory-servicesd` does no access-control checking, so the services it offers to Genera on a UX embedded within the UNIX on which `ivory-servicesd` is running, are also available (with some effort) to all hosts on the network. These services are:

- Setting the Sun clock
- Adding and deleting multicast masks to/from a Sun network interface

`ivory-servicesd` also provides a number of information-gathering capabilities:

- Looking up the SunOS kernel identifier

- Looking up the Sun's Network Information Service domain name
- Looking up the contents of the files /etc/fstab or /etc/mtab

ivory-servicesd also provides support for the embedded serial protocol, which is used by Genera to manipulate devices connected to the UNIX's serial ports. Access to UNIX serial ports is controlled by the usual UNIX idiom, based on the user logged in to the requesting Genera.

ivory-life UNIX Program

ivory-life *options device*

The ivory-life program is the primary life support program. It is responsible for initializing communication between UNIX and Genera, and for invoking the kernel support code to provide disk and network services. ivory-life is usually invoked during the UNIX system startup process. Note that ivory-life may be started before the X server.

ivory-life must run as root, since it performs privileged operations. ivory-life is also secure, in that it does not provide any dangerous capabilities or other ways to evade normal UNIX security.

When ivory-life is started, it reads the UX configuration file for *device*. Using the information from the configuration file, it initializes the shared memory through which UNIX will communicate with Genera. ivory-life then forks three times, and all four processes invoke portions of the UX kernel support. One of the child processes is responsible for managing Genera disk I/O, and will show up in process reports as ivory-disk. Another child process manages network I/O, and shows up as ivory-net. The ivory-misc process performs miscellaneous operations. The parent process performs internal housekeeping functions and retains the name ivory-life. Before invoking the kernel support, the parent writes out its process id number (PID) to the file /etc/*name*.pid, where *name* is the name portion of *device* (if *device* is /dev/ivory0, the PID will be written to /etc/ivory0.pid). These processes run in the kernel (except for ivory-misc, which runs in user mode) until they are killed or UNIX is shut down.

Note: If for some reason you need to restart UX life support, you must make sure to kill all four ivory-life processes before you run ivory-life again. Failure to do so can cause mysterious UNIX crashes.

If *device* is not specified, it defaults to /dev/ivory0, if present, or /dev/ivory/6.

Options:

- f *file* *file* is the UX configuration file to read. The default is to derive *file* from *device*'s name: if *device* is /dev/ivory0, *file* will default to /etc/ivory0.conf.
- f *facility*

Symbolics UX Configuration File

Phase 3 of the installation program automatically creates the UX configuration file. This file is consulted only by the `ivory-life` program. The UX configuration file consists of a series of configuration commands, one per line. Command arguments are separated from each other and from the command name by spaces or tabs. Blank lines and lines that begin with a semicolon (;) or sharp sign (#) are ignored. Unless otherwise noted, configuration commands may only be given once.

The configuration file is normally stored in the `/etc` directory, with a name derived from the name of the device special file that represents its UX. Thus, the configuration file for `/dev/ivory0` is usually `/etc/ivory0.conf`. If you wish to store your configuration file in a non-standard place, you will have to give the `-f` option to the `ivory-life` program. See the section "ivory-life UNIX Program".

There are four configuration file commands:

device The name of the UNIX device special file to which this configuration file applies. This command is optional. A typical use would be:

```
;; This configuration file is for /dev/ivory0
device /dev/ivory0
```

interface The name of the network interface Genera will use. Use the `-i` option to the `netstat` command to see a list of the network interfaces supported on your workstation (see section 8 of the *SunOS Reference Manual* for more information on `netstat`, and section 4 of the same manual for more information on network interfaces). A typical use would be:

```
;; This machine has an on-board ethernet interface
interface ie0
```

Note: On Sun-3s and older Sun-4s, the network interface is usually `ie0`. On SPARCstations and SPARCservers, the network interface is usually `le0`.

Note: Do not use `lo0`, the loopback interface, as an argument to the interface configuration command.

partition The name of a FEP partition file. This command is required, and must be used once for each FEP partition file you create. A typical use would be:

```
;; The FEP partition file is in its own disk
;; partition, which is mounted on /ivory.
partition /ivory/fep0
```

prom-directory The name of the directory where FEP PROMs are stored. This command is required. A typical use would be:

```
;; FEP device PROMs were restored here from the
;; UNIX software tape
prom-directory /usr/share/symbolics/ifep
```

In general, any time you change the configuration file, you also need to restart `ivory-life`. Usually the easiest way to do this is to reboot the Sun.

UNIX Programs for Manipulating Symbolics UX Disk Partitions

The UX software distribution includes a number of UNIX programs for creating and manipulating FEP partition files. The installation program automatically runs the `create-fep-partition` program, which creates partitions and restores distribution world tapes. The other programs are provided for system administration use.

These programs are intended for installation and system administration use only, they are not general-purpose tools. They have a number of implementation limitations:

- The FEP file system does not handle multipage datamaps, so if your files get too fragmented, they fail ungracefully.
- There is no way to delete, expunge, or shrink any files.
- You cannot create new files unless the directory representation has room for their description, that is, the representation cannot be extended.
- There is no pathname merging, version defaulting, or `.newest` lookup.

These limitations should not interfere with using these programs to install a UX system, and used properly these tools will not damage an existing FEP file system.

create-fep-file UNIX Program

```
create-fep-file partition-filename fep-filename #blocks
```

Creates the specified FEP file in the specified FEP partition file. This may be used to create another paging file should Genera run out of paging space.

create-fep-partition UNIX Program

```
create-fep-partition options partition-filename source-device
```

Creates a UNIX file to store a FEP partition in, creates a FEP file system in that partition, creates paging files, and loads FEP and Genera software into the partition from either a world tape or another partition file. It may be run on existing partitions to install new software without harm.

partition-filename is the UNIX file name of the FEP partition file to be created and/or initialized. It may be remote, mounted via NFS, but we strongly discourage

the use of remote FEP partitions. Note that a newly created partition file is not recognized by Genera Life Support until life support is restarted with the partition described in its configuration file.

source-device is either a QIC-11 tape device (/dev/rst0 on most Sun workstations), a remote tape device, or the UNIX filename of another FEP file partition. To access a tape device remotely, use the syntax *host:device*. Note that specifying a UNIX filename for *source-device* is a little awkward: it copies (or offers to copy) every FEP and world file in the source partition file, and does not filter out non-newest versions.

Note: Some UNIX NFS implementations disable checksumming of UDP/IP packets as a performance improvement. This dangerous practice can allow data errors to go undetected. Do not copy Genera world files using NFS unless both the source and destination machines checksum UDP/IP packets. Contact your UNIX vendor for the procedures involved.

Options:

- erase Makes the program erase the partition file and create a new FEP file system from scratch. This destroys the previous contents of the partition, and should be used with caution.
- paging {#blocks} Lets you specify the amount of paging space you want, which may be zero. Otherwise, the program allocates approximately half of the partition to paging space. The paging space is always split into two files named Paging-1 and Paging-2.
- query Makes the program ask before loading each file into the new partition. The program is somewhat interactive even without this option.

install-fep-kernel UNIX Program

`install-fep-kernel partition-filename fep-kernel-filename`

Installs the specified FEP kernel (which must already exist in the file system) in the disk label of the partition. This may be used in conjunction with `write-fep-file` to recover after accidentally destroying the FEP kernel of a UX.

read-fep-file UNIX Program

`read-fep-file options partition-filename fep-filename`

Reads the specified FEP file to stdout.

Options:

- characters Turns on Genera-to-UNIX character set translation.

show-fep-directory UNIX Program

show-fep-directory *partition-filename*

Displays a directory listing for the FEP file system in the specified FEP partition file.

write-fep-file UNIX Program

write-fep-file *options partition-filename fep-filename*

Writes from stdin to the specified FEP file.

Options:

-characters Turns on UNIX-to-Genera character set translation.

Installing the Genera Software on a UX**UNIX Requirements for Installing Genera on a Symbolics UX Machine**

There must be a login named lisp-machine on the SYS host. This user must have read and write access to the SYS directories. This user will be used by Genera to create the site files and to restore the Genera sources.

The tape or CD-ROM host and the SYS host must be using Network Information Service. They don't have to be Network Information Service (NIS) servers, but they must be NIS clients — Genera can find their NIS server by asking them. Genera uses NIS to look up host names for NFS mount points and to determine the UNIX uid information for the lisp-machine user. See the section "Providing UNIX NFS File Access to a Symbolics Computer".

The tape or CD-ROM host and the SYS host must be running the ivory-servicesd program. Genera uses the ivory-servicesd program to determine NFS mount points and the site NIS domain name.

Restoring the Genera Distribution Worlds From CD-ROM

Note: CD-ROM installation only works properly in Genera 8.1.1 or later versions. Be sure to install the Genera 8.3 worlds using a machine running Genera 8.1.1. or later.

Restore the Genera 8.3 world load files into your FEP file system, using the CD-ROM labeled GENERA 8.3 DISTRIBUTION WORLDS, SOURCES, AND BUNDLED PRODUCTS.

1. Insert the Genera 8.3 CD-ROM into the CD-ROM drive at your site. Type the CP command Select Activity FEP-Tape. In the FEP-Tape activity, type the command:

Read Image File

We recommend that you type this command, rather than choose it from the FEP Tape menu, because the typed command gives you more control over the tape specification. Supply the pathname of a tape image on the Genera 8.3 CD-ROM according to the following:

CD-ROM drive on a Lisp machine

```
Lisp-host|CDROMn:>DISTRIBUTION>BASIC_WORLD.TAPE
```

CD-ROM drive on a Sun

```
Sun-host:/cdrom/distribution/basic_world.tape
```

where "n" is the SCSI address of the ROM Drive.

2. On Ivory-based machines, you are asked if you want to restore the FEP kernel and flod files. Answer Y (for Yes) and accept the default pathname.

Note: Do not delete your old FEP kernel and flod files. We recommend that you always save the previous version of the kernel and flod files for a backup. See the section "Saving Previous FEP Kernels and FLOD Files".

3. You are asked if you want to restore the Genera 8.3 Netboot Core. Answer Y (for Yes) and accept the default pathname.
4. You are asked if you want to restore the Genera 8.3 world load file. Answer Y (for Yes) and accept the default pathname, or specify a FEP pathname that indicates an Ivory world load, such as:

```
FEP0:>Genera-8-3.ilod
```

The FEP-Tape loading program then loads the world from the CD-ROM.

5. Update the boot.boot file on your machine to load the new world. Your updated boot.boot file should look similar to this:

```
Load World
Start
```

This automatically loads the most recently created world on your disk (excluding netboot cores). **Note:** The next time you boot your machine, you must cold boot the FEP in order to use the new FEP kernel and FLOD files.

See the section "Contents of Hello.Boot Files and Boot.Boot Files".

6. For the remaining Genera 8.3 distribution worlds, use the Copy File command to copy the appropriate Genera worlds from the "worlds" directory on the CD-ROM to your local FEP filesystem.

If the CD-ROM drive is connected to an NXP1000, XL or a MacIvory, the world pathnames will look similar to this (where *host* is the name of the CD-ROM host, and *n* in *CDROMn* is the SCSI address of the CD-ROM drive):

```
cdrom-host|CDROMn:>WORLDS>GENERA_8_3.ILOD
```

If the CD-ROM drive is connected to a Sun workstation, the world pathnames will look similar to this (where *host* is the name of the Sun CD-ROM host, and */cdrom* is the directory the CD-ROM is mounted onto).

```
cdrom-host:/cdrom/worlds/genera_8_3.load
```

Note: The ISO9660 pathnames on the CD-ROM contain underscores which should be changed to hyphens when you copy the files to your FEP.

For example, when copying the Genera 8.3 world from CD-ROM to a 3600, with the CD-ROM drive connected to a Sun, you would type the following to a Lisp Listener (where *3600-host* is the name of the 3600):

```
Copy File cdrom-host:/cdrom/worlds/genera_8_3.load 3600-host|FEP0:>genera-8-3.load
```

Similarly, with the CD-ROM drive, SCSI address 3, connected to a MacIvory or XL, you would type the following:

```
Copy File CDROM-HOST|CDROM3:>WORLDS>GENERA_8_3.LOAD (to)  
3600-host|FEP0:>genera-8-3.load
```

The worlds directory on the CD-ROM contains the following worlds:

```
GENERA_8_3.ILOD  
GENERA_8_3.LOAD  
GENERA_8_3_MACI_DELIVERY.ILOD  
GENERA_8_3_UX_DELIVERY.ILOD  
NETBOOT_CORE_GENERA_8_3.ILOD  
NETBOOT_CORE_GENERA_8_3.LOAD
```

If your site is set up and you are not adding any new machines, you are ready to make a Genera 8.1 site-configured world. See the section "Making a Genera Site-Configured World".

If you are a new site with more than one Symbolics machine, you are ready to set up your site. See the section "Creating a New Site for Genera".

After Installing Genera on the Symbolics UX Machine

Once you have finished installing the Genera 8.1 software:

- **New machines:** If you are installing a new UX machine, then you have already installed the Symbolics UX software on UNIX, and you are finished with the entire installation procedure.

- **Existing machines:** If you are upgrading the software of an existing UX machine, you should now install the Symbolics UX software on UNIX. See the section "Installing and Configuring the Symbolics UX UNIX Software".

Common UX Installation Situations

Upgrading Genera on a Symbolics UX System

Upgrading a UX system to a new release of Genera (including upgrading the UX UNIX software) is subtly different from installing a new UX system. Many of the same steps are performed, but in a different order; some phases of the UNIX UX software installation may be skipped, but the Genera software installation (which is done automatically when installing a new UX system) must be done manually.

When upgrading a UX system to a new release of Genera, you must install the new Genera software first. Do this by using the Genera FEP-Tape activity; see the section Restoring the Genera 8.1 Distribution World on an Ivory Machine.

If you have multiple UXes in a single Sun, you probably want to install the new Genera software on all of them at this stage. More specifically, you must install the new Genera software on all UXes on a single system if the new version of UX UNIX software is incompatible with the version of Genera you are currently running on your UXes. If the new version of UX UNIX software is compatible with the version of Genera you are currently running on your UXes, then installing the new Genera software on all UXes at once is optional.

Once you have installed the new Genera software, you can install the new version of the UX UNIX software. See the section Upgrading the Symbolics UX UNIX Software.

Upgrading the Symbolics UX UNIX Software

Upgrading the Symbolics UX UNIX Software for a new release of Genera is subtly different from installing a new UX system. For more information on UX installation, use the online documentation for the *Genera 8.1 Software Installation Guide*.

If the Sun CD-ROM is not connected to the machine on which the UX UNIX software is to be installed, the CD-ROM must be mounted onto the UX machine's filesystem. On the machine with the CD-ROM drive, make sure that the directory the CD-ROM is mounted onto is exported (see the SunOS Reference Manual entry on `exportfs`). On the machine on which the UX UNIX software is to be installed, mount the CD-ROM onto the filesystem:

```
% su
# mkdir /cdrom
# mount OtherSun:/cdrom /cdrom
```

In order to upgrade the Symbolics UX UNIX Software, follow the installation procedure described in Installing and Configuring the Symbolics UX UNIX Software,

but answer No to the questions in "Phase 4 (FEP Partition File Configuration) of Symbolics UX Installer Program" and some of the questions in "Phase 5 (UNIX Startup Configuration) of Symbolics UX Installer Program".

More specifically, the usual "Phase 5 (UNIX Startup Configuration) of Symbolics UX Installer Program" answers for an upgrade will be as follows:

Do you wish to install the UX400S/UX1200S kernel?

Yes. *Upgrading the UX UNIX software requires building a new UNIX kernel which will include the new release of the UX device drivers. This question controls whether the new kernel will be installed (the kernel is actually built in Phase 3).*

Do you wish to create the UX400S/UX1200S configuration file(s)?

No. *The UX configuration files only need to be updated in order to add or remove network interfaces of FEP partition files, or to change the directory where IFEP PROM files are stored, or when a new UX board is being installed. Upgrading the UX UNIX software doesn't usually involve any of these situations.*

Do you wish to create/update the UX400S/UX1200S device files in /dev?

Yes. *The UX device files should be updated whenever a new UNIX kernel is built, in order that they specify the correct major device number for the UX device driver.*

Do you wish /etc/rc.local to start UX400S/UX1200S life support?

No. */etc/rc.local only needs to be updated when a new UX board is added to the system.*

Do you wish to reboot UNIX after installation?

Yes. *UNIX should be rebooted whenever a new kernel is installed.*

A sample upgrade dialogue would look something like the following.

```
% su
```

```
Password:
```

```
# cd /cdrom/sun
```

```
# extract_cdrom
```

The following product will be installed:

```
Symbolics UX400S/UX1200S UNIX Software for SunOS 4.1.2, Genera 8.3
```

Copyright (c) 1993 Symbolics, Inc. All rights reserved.
Portions of font library Copyright (c) 1984 Bitstream, Inc.
All Rights Reserved.

TRADEMARK NOTICE

Symbolics, Ivory, Symbolics UX400S, and Symbolics UX1200S are trademarks and Genera is a registered trademark of Symbolics, Inc.

UNIX(R) is a registered trademark of AT&T. X Window System is a trademark of the Massachusetts Institute of Technology. Open Windows, Sun, Sun-3, Sun-4, and SunOS are trademarks, and Sun Microsystems is a registered trademark of Sun Microsystems, Inc.

Use the Genera Show Legal Notice command to see important legal notices.

Do you want to continue [y|n]? **y**

Do you wish to restore UX-family sun3 life support? **y**

Do you wish to restore UX-family life support sources? **y**

Do you wish to restore the Genera fonts? **y**

extract_cdrom: Restoring files
x ./sun3/etc/test-ivory, 139264 bytes, 272 tape blocks
x ./ifep/solstice-device.prom, 130717 bytes, 256 tape blocks
x ./src/Makefile, 58127 bytes, 114 tape blocks
x ./src/autoboot.c, 4193 bytes, 9 tape blocks
[many other files here ...]

extract_cdrom: Running installation tool

Do you wish to install the UX400S/UX1200S software (default Yes)?

The ivory-life program has to be run by root, the super-user.

Do you wish ivory-life to be installed setuid to root (default Yes)?

Certain services provided by the ivory-servicesd program require that it run as root.

Do you wish ivory-servicesd to be installed setuid to root (default Yes)?

Several of the UX400S/UX1200S life-support programs want to be able to read /dev/kmem.

Do you wish these programs to be installed setgid to kmem (default Yes)?

Where should application programs go (default /usr/bin)?

Where should maintenance and installation programs go (default /usr/etc)?

Where should include files go (default /usr/include/ivory)?

Where should libraries go (default /usr/lib)?

Do you wish to install sources to the UX400S/UX1200S software (default Yes)?

Where should sources go (default /usr/share/symbolics/src)?

Where should IFEP prompts go (default /usr/share/symbolics/iefep)?

Do you wish to install the Genera fonts for the Sun Open Windows X server (default Yes)?

Where do the Open Windows programs live (default /usr/openwin)?

Where should the Open Windows Genera fonts go (default /usr/openwin/lib/genera-fonts)?

WARNING: directory /usr/openwin/lib/genera-fonts is not in the search path defined by environment variable FONTPATH. Make sure to update FONTPATH to include /usr/openwin/lib/genera-fonts.

Do you wish to install the Genera fonts for the MIT X server (default No)?

Do you wish to build a UNIX kernel with support for the UX400S/UX1200S (default Yes)?

How many UX400S boards should this kernel support (default 1)? **0**

How many UX1200S boards should this kernel support (default 1)?

Where is your kernel build directory (default /usr/share/sys)?

What kernel architecture does this machine use (default sun3)?

What kernel usually runs on this machine (default GENERIC)?

Do you wish to create FEP partition file(s) (default Yes)? **no**

Do you wish to install the UX400S/UX1200S kernel (default Yes)? **yes**

Do you wish to create the UX400S/UX1200S configuration file(s) (default Yes)? **no**

Do you wish to create/update the UX400S/UX1200S device files in /dev (default Yes)? **yes**

Do you wish /etc/rc.local to start UX400S/UX1200S life support (default Yes)? **no**

Do you wish to reboot UNIX after installation (default Yes)? **yes**

About to begin UX400S/UX1200S software installation, this will probably take a while.
Okay (default Yes)?

Upgrading Genera Without Upgrading Symbolics UX UNIX Software

Occasionally, you may wish to upgrade to a new Genera release without upgrading to the corresponding release of the Symbolics UX UNIX software. One scenario for this kind of split-release operation is when the new version of the Symbolics UX UNIX software requires a newer version of the UNIX operating system than is being used at your site. Split-release operation is only supported when the new Genera release is documented as being compatible with the version of the UX UNIX software you intend to run. See the section "Symbolics UX UNIX Software Compatibility Considerations".

If the new Genera software will be compatible with the version of the UX UNIX software you intend to run, you may install the new Genera software by using the Genera FEP-Tape activity. See the section "Restoring the Genera Distribution Worlds From CD-ROM".

Upgrading Symbolics UX UNIX Software Without Upgrading Genera

Occasionally, you may wish to upgrade to a new release of the Symbolics UX UNIX software without upgrading to the corresponding Genera software. One scenario for this kind of split-release operation is when the new version of the Symbolics UX UNIX software supports a new version of the UNIX operating system. Split-release operation is only supported when the new Genera release is documented as being compatible with the version of the UX UNIX software you intend to run. See the section "Symbolics UX UNIX Software Compatibility Considerations".

If the new UX UNIX software will be compatible with the version of the Genera software you intend to run, you may install the new UX UNIX software by running the UX installation tool. See the section "Upgrading Genera on a Symbolics UX System".

Running Multiple Releases of Genera in a Symbolics UX System

Multiple releases of Genera software may be freely intermixed in a Symbolics UX system, as long as all the Genera software is compatible with the release of the Symbolics UX UNIX software you intend to run. See the section "Symbolics UX UNIX Software Compatibility Considerations".

Symbolics UX UNIX Software Compatibility Considerations

The version of the Symbolics UX UNIX Software that is distributed with Genera 8.1 is compatible with all versions of Genera 8.0 and Genera 8.1.

The version of the Symbolics UX UNIX Software that was distributed with Genera 8.0 ECO#2 is compatible with all versions of Genera 8.0 and Genera 8.1, although some major features are not supported. Among these unsupported features is Ivory Netboot.

The versions of the Symbolics UX UNIX Software that were distributed with Genera 8.0 and Genera 8.0 ECO#1 are compatible with all versions of Genera 8.0 and Genera 8.1, although some major features are not supported. Among these unsupported features are SunOS 4.1, the Symbolics UX1200S, and Ivory Netboot.

The UX UNIX software that was distributed with Genera 7.4 Ivory ECO#3 is compatible only with Genera 7.4 Ivory ECO#3 software.

Running Multiple Releases of Symbolics UX UNIX Software in a Symbolics UX System

Symbolics does not support running multiple releases of the UX UNIX software on a single Sun. Forwards- and backwards-compatibility is always a goal while developing new releases of the UX UNIX software, but because critical functionality is imbedded in device drivers configured into the UNIX kernel, there is no way to run multiple releases of this software at once.

Installing Symbolics UX UNIX Software At Sites with a Central UNIX File-server

Sites with a central UNIX fileserver (accessed via NFS or a similar protocol) can avoid repeating certain steps of the Symbolics UX UNIX installation procedure by storing sharable files on the central fileserver. These sharable files include executable programs, libraries, sources, and X fonts.

Most common UNIX fileserver layouts (such as those recommended by Sun for Sun file servers) arrange to share executable programs and other binary files between all machines of the same architecture (for instance, Sun-3 or Sun-4), and to share architecture-independent files such as X fonts between all machines. In order to take advantage of the shortcuts described below, your fileserver must be configured in this manner.

At sites with a properly-laid-out central fileserver, only one copy of the X fonts needs to be installed. After that, each subsequent installation of the same Genera release can answer No to the questions in "Phase 2 (X Font Installation) of Symbolics UX Installer Program". Similarly, only one copy of the executable programs and libraries need to be installed for each UNIX architecture. After the first installation for each UNIX architecture, each subsequent installation of the same Genera release for the same UNIX architecture can answer No to the questions in "Phase 1 (UX Software Installation) of Symbolics UX Installer Program".

Differences Between Symbolics UX400S and Symbolics UX1200S

There are very few differences between Symbolics UX400S installations and Symbolics UX1200S installations. In both cases, hardware installation is performed by Symbolics Field Service, and software installation is performed by the customer. With one exception, all the Symbolics UX software operates in the same manner on UX1200Ses as on UX400Ses.

The major difference between the UX400S and the UX1200S is, of course, processor performance. The UX400S is based on the same processor design as the XL400, while the UX1200S is based on the newer, higher-performance XL1200 processor. Performance of the XL1200/UX1200S processor is rated at three times that of the XL400/UX400S processor.

Because the test-ivory program must be able to operate without support from the UX device driver, it is sometimes necessary to specify the model of UX board being tested. Use the `-t` switch to do this. For instance, the following invocation will test a UX1200S:

```
% test-ivory -t UX1200S other switches
```

One other user-visible difference between Symbolics UX400Ses and Symbolics UX1200Ses is in the device numbering. Symbolics UX400Ses are numbered starting from 0: `/dev/ivory0` specifies a UX400S, `/ivory/fep0` is usually a FEP file partition for a UX400S, and so on. Symbolics UX1200Ses are numbered starting from 16: `/dev/ivory16` specifies a UX1200S, `/ivory/fep16` is usually a FEP file partition for a UX1200S, and so on.

Support for the differing UX models is installed in the UNIX kernel as part of "Phase 3 (UNIX Kernel Configuration) of Symbolics UX Installer Program". Answer the questions

How many UX400S boards should this kernel support?

and

How many UX1200S boards should this kernel support?

with the maximum number of each model of UX that will be installed in a Sun running this kernel. (To add more boards of either model to an existing system, see the section "Adding a Symbolics UX to an Existing System".)

Mixing Symbolics UX400Ses and Symbolics UX1200Ses

Symbolics UX400Ses and Symbolics UX1200Ses may be intermixed freely in a single Sun system, up to the limits qualified by Symbolics. See the section "Differences Between Symbolics UX400S and Symbolics UX1200S" for more information on differences between the two models, and on how to configure a UNIX kernel to support both models of UX. See the section "Adding a Symbolics UX to an Existing System" See the section for information on how to add UX boards to an existing UX system.

Adding a Symbolics UX to an Existing System

Several phases of the installation procedure may be skipped when adding a new Symbolics UX board to a Sun system with an existing UX already installed. As always, the first step is to have the UX board installed by Symbolics Field Service. Once the board has been installed and tested (using the test-ivory program), the install-solstice program is run. This is the same program as is run off the Symbolics UX UNIX Software distribution tape. For convenience in upgrading UX systems, it is installed on-line during the software installation process. Normally it will be installed in /usr/etc, but it may be in a different directory depending on your answer to the "Phase 1 (UX Software Installation) of Symbolics UX Installer Program" question

Where should maintenance and installation programs go?

Because it is being run manually, and not off the distribution medium, the installation tool will automatically skip "Phase 1 (UX Software Installation) of Symbolics UX Installer Program" and "Phase 2 (X Font Installation) of Symbolics UX Installer Program", and begin with "Phase 3 (UNIX Kernel Configuration) of Symbolics UX Installer Program". Specify the total number of UX boards of each model (UX400S and UX1200S) that are installed in the system. For example, if a UX1200S were being added to a system that already contained a UX400S, the dialogue might look like this:

```
% su
Password:
# /usr/etc/install-solstice
```

Do you wish to build a UNIX kernel with support for the UX400S/UX1200S (default Yes)?
 How many UX400S boards should this kernel support (default 1)? **1**
 How many UX1200S boards should this kernel support (default 1)? **1**
 Where is your kernel build directory (default /usr/share/sys)?
 What kernel usually runs on this machine (default UX400S)?

In "Phase 4 (FEP Partition File Configuration) of Symbolics UX Installer Program", use the installation tool to create a FEP partition file only for the new board. Note that the installation tool asks what kind of distribution medium you have, and where it will be mounted: this is because it has been run manually, and has no way to tell whether your distribution came on tape or CD-ROM.

In continuing the example installation, note the No answer to the question

Do you wish to create a FEP partition file for UX400S device #0?

Do you wish to create FEP partition file(s) (default Yes)?
 Is your Symbolics distribution on CD-ROM or Tape (default Tape)?
 Is the tape device on the local host (default Yes)?
 What is the tape device (default st0)?
 Where do UX400S/UX1200S maintenance and installation programs live (default /usr/local/etc)?
 Do you wish to create a FEP partition file for UX400S device #0 (default Yes)? **No**
 Do you wish to create a FEP partition file for UX1200S device #16 (default Yes)? **Yes**
 Where do you wish to create the FEP partition file (default /ivory/fep16)?
 Unable to figure out how much space is available for this file
 How big do you wish the partition to be (default 150000)?
 Do you wish to create a Hello.boot file (default Yes)?
 What is the UX400S/UX1200S's Internet address? **128.81.20.125**
 What is the UX400S/UX1200S's Chaosnet address? **12175**

In "Phase 5 (UNIX Startup Configuration) of Symbolics UX Installer Program", re-configure the UNIX startup procedures. Note that (due to limitations of the Genera 8.1 UX installation tool) the configuration files for any pre-existing UX boards will be rewritten, and questions will be asked about the information that should go in them. Note also that the answer to the question

Do you wish /etc/rc.local to start UX400S/UX1200S life support?

should be Yes even if you already answered it Yes in a previous installation (with fewer UX boards). This is because the previous installation will have updated /etc/rc.local to start life-support only for the UX boards that were previously installed; to have /etc/rc.local start life-support for the newly-installed boards, /etc/rc.local must be updated again.

Do you wish to install the UX400S/UX1200S kernel (default Yes)?
 Do you wish to create the UX400S/UX1200S configuration file(s) (default Yes)?
 Where do IFEP proms live (default /usr/share/symbolics/ifep)?
 Do you wish UX400S #0 to use network interface ie0 (default Yes)?
 Do you wish UX1200S #0 to use network interface ie0 (default Yes)?
 Where is the partition file for UX400S #0 (default /ivory/fep0)?
 Do you wish to create/update the UX400S/UX1200S device files in /dev (default Yes)?

Do you wish /etc/rc.local to start UX400S/UX1200S life support (default Yes)?

Do you wish to reboot UNIX after installation (default Yes)?

About to begin UX400S/UX1200S software installation, this will probably take a while.
Okay (default Yes)?

Is the Genera Distribution World tape mounted (default Yes)?

Upgrading a Symbolics UX400S to a Symbolics UX1200S

Upgrading a Symbolics UX400S to a Symbolics UX1200S requires three steps:

1. Upgrading Genera on the UX400S (if necessary)
2. The actual hardware upgrade
3. Reconfiguring the UX UNIX software to support the UX1200S instead of the UX400S

If the UX400S that is going to be upgraded is running Genera 8.0 ECO#2 or later (including the Symbolics UX UNIX software), then there is no need to upgrade Genera before upgrading the UX400S to a UX1200S. If the UX400S is running Genera 8.0 ECO#1 or earlier, then you must upgrade to a release of Genera that supports the UX1200S **before** performing the actual hardware upgrade. This is because the UX400S's FEP Partition File will eventually be used by the UX1200S, so the software installed in it must support the UX1200S; similarly, the UNIX kernel and other UX UNIX software must support the UX1200S. To upgrade Genera, see the section "Upgrading Genera on a Symbolics UX System".

The hardware upgrade (swapping a UX400S for a UX1200S) is performed by Symbolics Field Service.

Once the UX1200S has been installed and tested (using the test-ivory program), the UX UNIX software is reconfigured by running the install-solstice program. This is the same program as is run off the Symbolics UX UNIX Software distribution tape. For convenience in upgrading UX systems, it is installed on-line during the software installation process. Normally it will be installed in /usr/etc, but it may be in a different directory depending on your answer to the "Phase 1 (UX Software Installation) of Symbolics UX Installer Program" installation question

Where should maintenance and installation programs go?

Because it is being run manually, and not off the distribution medium, the installation tool will automatically skip "Phase 1 (UX Software Installation) of Symbolics UX Installer Program" and "Phase 2 (X Font Installation) of Symbolics UX Installer Program", and begin with "Phase 3 (UNIX Kernel Configuration) of Symbolics UX Installer Program". In "Phase 3 (UNIX Kernel Configuration) of Symbolics UX Installer Program", answer the questions as appropriate, remembering that there will now be one fewer UX400S and one more UX1200S than the last time

you ran the program. Answer No to the questions in "Phase 4 (FEP Partition File Configuration) of Symbolics UX Installer Program"; rather than building and configuring a new FEP Partition file, you will reuse the UX400S's partition file for the UX1200S. In "Phase 5 (UNIX Startup Configuration) of Symbolics UX Installer Program" you must perform all the updates, in order to reconfigure UNIX startup for the new UX1200S.

The following example demonstrates reconfiguring the UX UNIX software for a UX400S to UX1200S upgrade. In the example, there was a pre-existing UX400S that has been upgraded to a UX1200S. There are no other UX-family boards in the system.

```
% su
```

```
Password:
```

```
# /usr/etc/install-solstice
```

```
Do you wish to build a UNIX kernel with support for the UX400S/UX1200S (default Yes)?
```

```
How many UX400S boards should this kernel support (default 1)? 0
```

```
How many UX1200S boards should this kernel support (default 1)? 1
```

```
Where is your kernel build directory (default /usr/share/sys)?
```

```
What kernel usually runs on this machine (default UX400S)?
```

```
Do you wish to create FEP partition file(s) (default Yes)? No
```

```
Do you wish to install the UX400S/UX1200S kernel (default Yes)?
```

```
Do you wish to create the UX400S/UX1200S configuration file(s) (default Yes)?
```

```
Where do IFEP prompts live (default /usr/share/symbolics/ifep)?
```

```
Do you wish UX1200S #0 to use network interface ie0 (default Yes)?
```

```
Where is the partition file for UX1200S #0 (default /ivory/fep16)? /ivory/fep0
```

```
Do you wish to create/update the UX400S/UX1200S device files in /dev (default Yes)?
```

```
Do you wish /etc/rc.local to start UX400S/UX1200S life support (default Yes)?
```

```
Do you wish to reboot UNIX after installation (default Yes)?
```

```
About to begin UX400S/UX1200S software installation, this will probably take a while.
```

```
Okay (default Yes)?
```

Overview of the Symbolics UX1200S

The UX1200S is a more powerful, higher-performance version of the UX400S. The UX1200S has the same processor architecture as the XL1200, whereas the UX400S has the same processor architecture as the XL400.

UX1200S Software Requirements

The UX1200S requires Genera 8.0.2. Note that the UNIX software shipped with Genera 8.0.2 to UX1200S customers is compatible with SunOS 4.1 only.

In a configuration where multiple UX boards are installed in a single Sun, each UX board runs its own copy of Genera. In a configuration that includes mixed UX400S and UX1200S boards in a single Sun, you can run the Genera 8.0.2 software on the UX1200S boards and run the Genera 8.0 or 8.0.1 software on the UX400S boards; the Genera versions are compatible.

Note that Genera 8.0.2 can also be run on UX400S boards, but Genera 8.0.1 and Genera 8.0 cannot be run on UX1200S boards.

UX1200S Documentation Update

The documentation on the UX400S pertains to the UX1200S, except as noted in this section.

On the UNIX side, /dev/ivory0 is the device for the first UX400S board. Any additional UX400S boards are in /dev/ivory1, /dev/ivory2, and so on. Similarly, /dev/ivory16 is the device for the first UX1200S board. Any additional UX1200S boards are in /dev/ivory17, /dev/ivory18, and so on.

UX Installation Example

This section illustrates the actual installation procedure for installing a single UX400S board into a SUN-3/260 under Genera 8.0.2. The following conditions apply, and in most cases, the default is used:

1. The installation of a single UX400S into a SUN 3/260 machine.
2. The Internet address of the Ivory machine is 128.81.20.127, and its assigned Chaosnet address 12177
3. The FEP partition file name for this Ivory machine is /b/ivory/fep0.
4. The Genera fonts are located in /usr/openwin/genera-fonts. We suggest that you this add this directory name to your FONTPATH.

```
root@edvac 2# extract_unbundled
```

```
Enter media drive location [local | remote]: local
Enter Device Name (e.g. rst0, rmt0, rfd0c) : /dev/rst0
**Please mount the release media if you haven't done so already.**
Press return when ready:
```

```
The following product will be installed:
```

```
2+0 records in
```

```
2+0 records out
```

```
Symbolics UX400S/UX1200S UNIX Software for SunOS 4.1, Genera 8.0.2
```

```
Copyright (c) 1990 Symbolics, Inc. All rights reserved.
```

Portions of font library Copyright (c) 1984 Bitstream, Inc.

All Rights Reserved.

TRADEMARK NOTICE

Symbolics, Ivory, Symbolics UX400S, and Symbolics UX1200S are trademarks and Genera is a registered trademark of Symbolics, Inc.

UNIX(R) is a registered trademark of AT&T. X Window System is a trademark of the Massachusetts Institute of Technology. Open Windows, Sun, Sun-3, Sun-4, and SunOS are trademarks, and Sun Microsystems is a registered trademark of Sun Microsystems, Inc.

Use the Genera Show Legal Notice command to see important legal notices.

Do you want to continue [y|n]? y

/usr/etc/extract_unbundled : Extracting Installation Scripts

tar: blocksize = 20

x ./install_unbundled, 4632 bytes, 10 tape blocks

x ./unbundled_sizes, 82 bytes, 1 tape blocks

/usr/etc/extract_unbundled : Begin Install Script Execution

Do you wish to restore UX400S sun3 life support? y

Do you wish to restore UX400S life support sources? y

Do you wish to restore the Genera fonts? y

/usr/tmp/unbundled/install_unbundled: Restoring files

x ./sun3/etc/test-ivory, 139264 bytes, 272 tape blocks

.

.

.

Long list of files being restored

.

.

.

x ./fonts/vt100.bdf, 9908 bytes, 20 tape blocks

/usr/tmp/unbundled/install_unbundled: Running installation tool

Do you wish to install the UX400S/UX1200S software (default Yes)?
The ivory-life program has to be run by root, the super-user.
Do you wish ivory-life to be installed setuid to root (default Yes)?
Certain services provided by the ivory-servicesd program require that it run as root.
Do you wish ivory-servicesd to be installed setuid to root (default Yes)?
Several of the UX400S/UX1200S life-support programs want to be able to read /dev/kmem.
Do you wish these programs to be installed setgid to kmem (default Yes)?
Where should application programs go (default /usr/local/bin)?
Where should maintenance and installation programs go (default /usr/etc)?
Where should include files go (default /usr/include/ivory)?
Where should libraries go (default /usr/lib)?
Do you wish to install sources to the UX400S/UX1200S software (default Yes)?
Where should sources go (default /usr/share/symbolics/src)?
Where should IFEP proms go (default /usr/share/symbolics/ifep)?

Do you wish to install the Genera fonts for the Sun Open Windows X server (default Yes)?
Where do the Open Windows programs live (default /usr/openwin)?
Where should the Open Windows Genera fonts
go (default /usr-genera-fonts)? /usr/openwin/genera-fonts
Do you wish to install the Genera fonts for the MIT X server (default No)?

Do you wish to build a UNIX kernel with support for the UX400S/UX1200S (default Yes)?
How many UX400S boards should this kernel support (default 1)?
How many UX1200S boards should this kernel support (default 1)? 0
Where is your kernel build directory (default /usr/share/sys)?
What kernel usually runs on this machine (default GENERIC)?

Do you wish to create FEP partition file(s) (default Yes)?
Do you wish to create a FEP partition file for UX400S board #0 (default Yes)?
Where do you wish to create the FEP partition file (default /ivory/fep0)? /b/ivory/fep0
There are 317544 kbytes available for this file.
How big do you wish the partition to be (default 150000)? 317544
Do you wish to create a Hello.boot file (default Yes)?
What is the UX400S/UX1200S's Internet address? 128.81.20.127
What is the UX400S/UX1200S's Chaosnet address? 12177

Do you wish to install the UX400S/UX1200S kernel (default Yes)?
Do you wish to create the UX400S/UX1200S configuration file(s) (default Yes)?
Do you wish UX400S #0 to use network interface ie0 (default Yes)?
Do you wish to create/update the UX400S/UX1200S device files in /dev (default Yes)?
Do you wish /etc/rc.local to start UX400S/UX1200S life support (default Yes)?
Do you wish to reboot UNIX after installation (default Yes)? no

About to begin UX400S/UX1200S software installation, this will probably take a while.
Okay (default Yes)?

Is the Genera Distribution World tape mounted (default Yes)?

```
% mkdir /usr/etc
Updating application programs... Updated 3 files..
Updating maintenance and installation programs... Updated 10 files..
% mkdir /usr/include/ivory
Updating include files... Updated 2 files..
Updating libraries... Updated 1 file..
% mkdir /usr/share/symbolics
% mkdir /usr/share/symbolics/src
Updating sources... Updated 78 files..
% mkdir /usr/share/symbolics/ifep
Updating IFEP prompts... Updated 1 file..

Installing Genera fonts for Open Windows... Updated 224 files..
% cd /usr/openwin/genera-fonts; /usr/openwin/bin/bldfamily -f 100
* genera-vt100          ./generavt100.ff (Encoding: unknown)
.
.
.
    long list of fonts being installed
.
.
.
* genera-abacus        ./generaabacus.ff (Encoding: unknown)
cat: ./Compat.list: No such file or directory
Updating kernel binaries... Updated 11 files..
```

```

Configuring UX400S/UX1200S kernel...
[Updated /usr/share/sys/sun/conf.c; Ivory major device number is 104]
[Updated /usr/share/sys/sun/str_conf.c]
[Updated /usr/share/sys/conf.common/files.cmn]
[Created /usr/share/sys/sun3/conf/UX400S from /usr/share/sys/sun3/conf/GENERIC]
% cd /usr/share/sys/sun3/conf; /etc/config UX400S
Doing a "make depend"
Configured UX400S/UX1200S kernel]
% cd /usr/share/sys/sun3/UX400S; make

```

```

.
.
.
    commands to compile the new UNIX(tm) kernel
.
.
.

```

loading vmunix

rearranging symbols

text	data	bss	dec	hex
905344	191736	217944	1315024	1410d0

[Compiled UX400S/UX1200S kernel]

```
% /a/tmp/sun3/etc/create-fep-partition -automatic 325165056 yes 12177 128.81.20.127 /b/iv
```

/fep0 /dev/rst0

Creating 310 megabyte file /b/ivory/fep0...

This will take a few minutes (about 20, I'd guess).

Created file /b/ivory/fep0, stored in 39714 Unix blocks (8192 bytes).

Creating initial FEP file system in partition file /b/ivory/fep0...

Creating paging file Paging-1.Page.1 (60000 blocks)...

Creating paging file Paging-2.Page.1 (60000 blocks)...

Restoring I322-KERNEL.FEP.351 (192 blocks)...

Installing I322-KERNEL.FEP.351 as the FEP kernel to boot on this partition...

Restoring I322-LOADERS.FL0D.222 (42 blocks)...

Restoring I322-LISP.FL0D.145 (17 blocks)...

Restoring I322-INFO.FL0D.157 (11 blocks)...

Restoring I322-DEBUG.FL0D.158 (81 blocks)...

Restoring I322-DISK.FL0D.193 (43 blocks)...

Restoring I322-TESTS.FL0D.144 (10 blocks)...

Restoring Genera-8-0-2-UX-Delivery.ilod.1 (19590 blocks)...

Restoring Genera-8-0-2-Network.ilod.1 (42961 blocks)...

End of tape reached, file Genera-8-0-2-Network.ilod.1 continues on next tape.

Is the next tape in place and are you ready to continue? (yes or no) yes

```
Still restoring Genera-8-0-2-Network.ilod.1 (18361 blocks remaining)...
End of tape encountered.
Creating Hello.Boot and Boot.Boot files...
```

```
Installing kernel as /vmunix... Done.
Writing /etc/ivory0.conf...
Updating Ivory device files... Done.
Updating /etc/rc.local...
```

```
UX400S/UX1200S installation completed.
```

Genera 8.1 Software Installation Instructions for Installing From CD-ROM

Distribution of Genera 8.1 on CD-ROM

With Genera 8.1, customers have the alternative of receiving their Genera distribution on CD-ROM instead of on tape. CD-ROM (compact disc read-only memory) is a 650-megabyte disc that can hold the equivalent of 270,000 typed manuscript pages or 1500 standard floppy disks. CD-ROM also provides a convenient and compact method for loading software distributions.

For customers with MacIvory or XL systems, the drive can be plugged into a SCSI port on the system. Alternatively, we expect that customers with Sun systems, plus Sun's CD-ROM drive and Symbolics' IP/TCP product, will be able to access the CD-ROM distribution through FTP.

In Genera 8.1, you can access ISO9660 format CD-ROMs directly as a separate file system on XLs and MacIvories. For UXes, you can access ISO9660 discs in the Sun's CD-ROM drive using NFS or TCP-FTP.

For example, on an XL or MacIvory, if your CD-ROM drive is SCSI unit #3, you access its file system by saying something like

```
Show Directory CDR0M3:>
```

Of course, other Lisp machines can access these file systems over the network by saying something like

```
Show Directory SOUR-CREAM|CDROM3:>
```

ISO9660 is an international standard for storing file systems on CD-ROM. It provides a hierarchical file system. Files have names, types, and version numbers. Continuing the above example, a typical pathname might be

```
SOUR-CREAM|CDROM3:>IMAGES>NASA>MIRANDA>MIRANDA3.IMG;1
```

(Do not infer by the use of IMG as the type that there is a restriction of 3 characters on type names. The only restriction in the standard is that the name and type together not exceed 30 characters.)

IS9660 is the successor to the format known as High Sierra. However, Genera does not support High Sierra format discs.

it is possible to access sources and documentation directly from the CD-ROM if you setup your SYS host translations to point to the ISO9660 file system on the disc.

Overview of CD-ROM Installation

This section describes the general procedure for installing Genera from CD-ROM.

Note: In all cases, make sure that machine is powered off before connecting or removing the CD-ROM drive.

- Connect the CD-ROM drive to one of the following:
 - An NXP1000 or a Symbolics XL Family machine. See the section "Installing a SCSI Device". Symbolics has qualified the Apple CD-ROM drive (Sony model CDU-8001) for use with NXP1000s and XLs, however, other SCSI CD-ROM drives may also work.
 - A Symbolics MacIvory. Follow the manufacturer's instructions for configuring and connecting a SCSI CD-ROM drive.
 - A Sun workstation. Follow the manufacturer's instructions for configuring and connecting a SCSI CD-ROM device.
- When the CD-ROM drive is connected to a Sun, you must make the CD-ROM drive accessible to other machines on the network.

Setting up Network Access for a CD-ROM Drive on a Sun Workstation

1. Check the UNIX permissions on `/dev/sr0`. These permissions will be propagated by SunOS to all files on the Genera Distribution CD-ROM. At a minimum, the read and execute bits must be set for the user who will perform the installation (often root). For more information on setting UNIX file permissions, see the SunOS Reference Manual entry for `chmod`. For more information on the Sun CD-ROM driver, see the SunOS Reference Manual entry on `sr`.
2. Insert the CD-ROM in the Sun CD-ROM drive, and turn on the drive. Mount the CD-ROM into the Sun filesystem using the UNIX `mount` command (the `mount` command must be run by the super-user):

```
% su
# mount -o ro -t hsfs /dev/sr0 /cdrom
```

For more information on the UNIX `mount` command, see the SunOS Reference Manual entry on `mount`.

3. On the machine with the CD-ROM drive, make sure that the directory the CD-ROM is mounted onto is exported (see the SunOS Reference Manual entry on `exportfs`).

CD-ROM Installation Steps - an Overview

1. *(Sites with MacIvories only)*
Install the Genera 8.3 MacIvory Macintosh software on all MacIvories which will run Genera 8.3. Since installing this software from CD-ROM requires a CD-ROM drive on the local MacIvory, we suggest that you install this from the Genera 8.3 MacIvory floppies. See the section "Installing the MacIvory Diskettes".
2. *(Sites with UX Family machines only)*
Install the Genera 8.3 UX UNIX software on all UX Family machines which will run Genera 8.3. Run the UX UNIX software installation tool, to install the Genera 8.3 UX UNIX software from the tape. See the section "Upgrading the Symbolics UX UNIX Software".
3. Restore the Genera 8.3 Distribution worlds from CD-ROM, see the section "Restoring the Genera Distribution Worlds From CD-ROM".
4. Make a site-configured Genera 8.3 world for each machine-type. See the section "Making a Genera Site-Configured World".

If your CD-ROM drive is connected to a MacIvory or an XL running Genera 8.3, it is possible to use the CD-ROM as a `SYS` host. If you wish to store any portion of the Genera 8.3 documentation, sources, examples, fonts, and non-loaded systems on CD-ROM instead of in a LMFS, See the section "Using a CD-ROM as a `SYS` Host" and use that section as a guide when updating your `sys.translations` file.

5. If there are any Genera 8.3 documentation, sources, examples, fonts, or non-loaded systems that you do *not* plan to access from CD-ROM, you must restore them into a Lisp Machine File System (LMFS) or a UNIX File System. For instructions on how to do this, see the section "Restoring the Genera Sources".
6. On each machine that is now running a site-configured Genera 8.3 world, copy the new flod files. See the section "Copying New Flod Files" for each machine type. **Note:** The Flod files for 3600-family machines are the same as in previous Genera releases. Therefore, this step is not necessary on 3600-family machines.
7. Cold boot each machine with the new flods.

8. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
9. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

Initial CD-ROM Installation of Genera on an XL

Note: CD-ROM installation only works properly in Genera 8.1.1 or later versions. Be sure to install on a machine running Genera 8.1.1. or later CD-ROM.

1. Make a site-configured Genera 8.3 world for your site. See the section "Making a Genera Site-Configured World".

If your CD-ROM drive is connected to a MacIvory or an XL running Genera 8.1.1, it is possible to use the CD-ROM as a SYS host. If you wish to store any portion of the Genera 8.1.1 documentation, sources, examples, fonts, and non-loaded systems on CD-ROM instead of in a LMFS, see the section "Using a CD-ROM as a SYS Host" and use that section as a guide when updating your `sys.translations` file.

2. Boot a site-configured Genera 8.1.1 world on the initial 8.1 machine, and use the FEP-Tape activity to restore the remaining Ivory worlds from the CD-ROM. Type

```
Select Activity FEP-Tape
```

In the FEP-Tape frame, type (Where "n" in "CDROMn" is the SCSI address of the local CD-ROM drive.)

```
Read Image File CDROMn:>DISTRIBUTION>OTHER_IVORY_WORLDS.TAPE
```

Do not use the [Read Image File] menu item, because it will not prompt you for a world image pathname.

The OTHER_IVORY_WORLDS.TAPE image contains the worlds:

```
Netboot-Core-from-Genera-8-1.ilod
Network-from-Genera-8-1.ilod
Genera-8-1-MacIvory-Delivery.ilod
Genera-8-1-UX-Delivery.ilod.1
```

The FEP-Tape activity queries you before restoring each world. Type Y to restore any worlds which you need at your site. Type N to skip restoring any worlds which you do not need.

If you plan to use Ivory netboot at your site, you need to restore

```
Netboot-Core-from-Genera-8-1.ilod
```

If you have any UX Family machines at your site, you need to restore

`Network-from-Genera-8-1.ilod`

If you have any MacIvories at your site, and you plan to run the Genera 8.1.1 Delivery world on them, you need to restore

`Genera-8-1-MacIvory-Delivery.ilod`.

If you have any UX Family machines at your site, and you plan to run the Genera 8.1 Delivery world on them, you need to restore

`Genera-8-1-UX-Delivery.ilod`

Initial CD-ROM Installation of Genera on a MacIvory

Genera 8.1 introduces support for ISO9660 pathnames, which is needed to access much of the software on the Genera 8.1 CD-ROM. This section describes how to get Genera 8.1 running in the absence of ISO9660 pathname support.

Note: Genera 8.1.1 contains important patches for CD-ROM users. If you do not already have the Genera 8.1.1 CD-ROM, contact Symbolics Software Support at 1-800-966-LISP.

1. Turn on the CD-ROM drive, and insert the Symbolics Genera 8.1 CD in the drive.
2. Install the Genera 8.1 MacIvory Macintosh software from the CD-ROM. See the section "Installing Macintosh Software From a CD-ROM". See the section "Installing the MacIvory Diskettes".
3. Open the Ivory Breath of Life application, which is located in the MacIvory Applications folder. Use the Disk Restore menu to restore the Genera 8.1 Distribution world from CD-ROM. In the Disk Restore menu, specify the following parameters:

Select the operation [Restore Files Only]. This will preserve all existing information on the disk. Note: The other possible operation, [Create New Filesystem], assumes that you are building an Ivory partition from scratch, and will therefore erase all existing information in the partition.)

Specify the SCSI address of the CD-ROM drive. The SCSI addresses of available CD-ROM drives appear in boldface.

Select the Ivory partition you wish to install the Genera 8.1 world into.

Click on [Restore] to begin the restoration. Disk Restore will automatically find the Genera 8.1 flod files and distribution world load on the CD-ROM, and restore them into the the Ivory partition you specified. It does not automati-

cally update your hello.boot and boot.boot files, but it creates new hello.boot and boot.boot files if you do not have any. In any case, it creates the file I325.boot. See the section "Contents of Hello.Boot Files and Boot.Boot Files".

4. Once Disk Restore is finished, Quit the Ivory Breath of Life program. Restart the Macintosh, then double click on the Genera icon to boot Genera.

If your boot files were not updated, transfer to the Ivory FEP and type

```
Hello I325
```

Then type the rest of the commands in your old hello.boot file by hand. Then type

```
Load World
Start
```

Go to the Ivory menu and click on "Quit". Click on "Ok". Then Double click on the Genera icon to boot Genera. When Genera comes up, update your hello.boot and boot.boot files. See the section "Contents of Hello.Boot Files and Boot.Boot Files".

5. Make a site-configured Genera 8.1 world for your site. See the section "Making a Genera Site-Configured World".

If your CD-ROM drive is connected to a MacIvory or an XL running Genera 8.1, it is possible to use the CD-ROM as a SYS host. If you wish to store any portion of the Genera 8.1 documentation, sources, examples, fonts, and non-loaded systems on CD-ROM instead of in a LMFS, see the section "Using a CD-ROM as a SYS Host" and use that section as a guide when updating your sys.translations file.

6. Boot a site-configured Genera 8.1 world on the initial 8.1 machine, and use the FEP-Tape activity to restore the remaining Ivory worlds from the CD-ROM. Type

```
Select Activity FEP-Tape
```

In the FEP-Tape frame, type (Where "n" in "CDROMn" is the SCSI address of the local CD-ROM drive.)

```
Read Image File CDROMn:>DISTRIBUTION>OTHER_IVORY_WORLDS.TAPE
```

Do not use the [Read Image File] menu item, because it will not prompt you for a world image pathname.

The OTHER_IVORY_WORLDS.TAPE image contains the worlds:

```
Netboot-Core-from-Genera-8-1.ilod
Network-from-Genera-8-1.ilod
Genera-8-1-MacIvory-Delivery.ilod
Genera-8-1-UX-Delivery.ilod
```

The FEP-Tape activity will query you before restoring each world. Type Y to restore any worlds which you will need at your site. Type N to skip restoring any worlds which you will not need.

If you plan to use Ivory netboot at your site, you need to restore

```
Netboot-Core-from-Genera-8-1.ilod
```

If you have any UX Family machines at your site, you need to restore

```
Network-from-Genera-8-1.ilod
```

If you have any MacIvories at your site, and you plan to run the Genera 8.1 Delivery world on them, you need to restore

```
Genera-8-1-MacIvory-Delivery.ilod
```

If you have any UX Family machines at your site, and you plan to run the Genera 8.1 Delivery world on them, you need to restore

```
Genera-8-1-UX-Delivery.ilod
```

Initial CD-ROM Installation of Genera From a CD-ROM on a Sun

When a CD-ROM drive is connected to a Sun workstation, the Sun provides the ISO9660 pathname support. Note that in order to connect to the Sun, your Symbolics machine must be running either IP-TCP or NFS.

When the CD-ROM drive is connected to a Sun, you must make the CD-ROM drive accessible to other machines on the network.

Setting up Network Access for a CD-ROM Drive on a Sun Workstation

1. Check the UNIX permissions on `/dev/sr0`. These permissions will be propagated by SunOS to all files on the Genera Distribution CD-ROM. At a minimum, the read and execute bits must be set for the user who will perform the installation (often root). For more information on setting UNIX file permissions, see the SunOS Reference Manual entry for `chmod`. For more information on the Sun CD-ROM driver, see the SunOS Reference Manual entry on `sr`.
2. Insert the CD-ROM in the Sun CD-ROM drive, and turn on the drive. Mount the CD-ROM into the Sun filesystem using the UNIX `mount` command (the `mount` command must be run by the super-user):

```
% su
# mount -o ro -t hsfs /dev/sr0 /cdrom
```

For more information on the UNIX `mount` command, see the SunOS Reference Manual entry on `mount`.

3. On the machine with the CD-ROM drive, make sure that the directory the CD-ROM is mounted onto is exported (see the SunOS Reference Manual entry on `exportfs`).

Remote CD-ROM Installation Procedure for 3600, MacIvory, and XL

1. *(For MacIvory only)* If your initial 8.1 Symbolics machine is a MacIvory, you must first install the Genera 8.1 MacIvory Macintosh software. Since the Sun cannot read the MacFS on the CD-ROM, you must install the MacIvory Macintosh software from diskettes. See the section "Installing the MacIvory Diskettes".
2. Use the Copy File command to copy the appropriate Genera 8.1 Distribution world(s) to your local FEP filesystem. You may do this from a Symbolics machine running Genera 8.0, because the Sun provides ISO9660 pathname support. Note that because the CD-ROM drive is connected to a Sun, you must specify the world load files in lowercase. Also note that the ISO9660 pathnames on the CD-ROM contain underscores which should be changed to hyphens when you copy the files.

- On a 3600:

```
Copy File cdrom-host:/cdrom/worlds/netboot_core_from_genera_8_1.load (to)
FEPn:>netboot-core-from-genera-8-1.load
Copy File cdrom-host:/cdrom/worlds/genera_8_1.load FEPn:>genera-8-1.load
```

- On a MacIvory:

```
Copy File cdrom-host:/cdrom/worlds/netboot_core_from_genera_8_1.ilod (to)
FEPn:>netboot-core-from-genera-8-1.ilod
Copy File cdrom-host:/cdrom/worlds/genera_8_1.ilod FEPn:>genera-8-1.ilod
```

Optionally,

```
Copy File cdrom-host:/cdrom/worlds/genera_8_1_macivory_delivery.ilod (to)
FEPn:>genera-8-1-macivory-delivery.ilod
```

- On an XL:

```
Copy File cdrom-host:/cdrom/worlds/netboot_core_from_genera_8_1.ilod (to)
FEPn:>netboot-core-from-genera-8-1
Copy File cdrom-host:/cdrom/worlds/genera_8_1.ilod FEPn:>genera-8-1.ilod
```

3. Boot the Genera 8.1 distribution world on the Symbolics machine.
4. Make a site-configured Genera 8.1 world for your site. See the section "Making a Genera Site-Configured World".

Remote CD-ROM Installation Procedure for UX

Customers who receive their Genera distribution on CD-ROM must follow a slightly different installation procedure from those who receive their Genera distribution on tape. Genera sources, world loads, and FEP files are distributed on a single CD-ROM labelled *Genera 8.3 CD-ROM*. UX UNIX software is distributed on a tape labelled *UX UNIX Software*.

Before beginning, see the section "UNIX Requirements for Installing Genera on a Symbolics UX Machine".

1. If the Sun CD-ROM is not connected to the machine on which the UX UNIX software is to be installed, the CD-ROM must be mounted onto the UX machine's filesystem. On the machine with the CD-ROM drive, make sure that the directory the CD-ROM is mounted onto is exported (see the SunOS Reference Manual entry on `exportfs`). On the machine on which the UX UNIX software is to be installed, mount the CD-ROM onto the filesystem:

```
% su
# mount OtherSun:/cdrom /cdrom
```

2. Use the Copy File CP command to copy the appropriate Genera 8.1 Distribution Worlds from CD-ROM to your local FEP filesystem. Note that because the CD-ROM drive is connected to a Sun, you must specify the world load files in lowercase. Also note that the ISO9660 pathnames on the CD-ROM contain underscores which should be changed to hyphens when you copy the files.

```
Copy File cdrom-host:/cdrom/worlds/netboot_core_from_genera_8_1.ilod (to)
FEPn:>netboot-core-from-genera-8-1.ilod
Copy File cdrom-host:/cdrom/worlds/genera_8_1.ilod.1 (to) FEPn:>genera-8-1.ilod
Copy File cdrom-host:/cdrom/worlds/network_from_genera_8_1.ilod (to)
FEPn:>network-from-genera-8-1.ilod
```

Optionally,

```
Copy File cdrom-host:/cdrom/worlds/genera_8_1_ux_delivery.ilod (to)
FEPn:>genera-8-1-ux-delivery.ilod
```

3. Run the UX UNIX software installation tool, to install the Genera 8.1 UX UNIX software from the tape.

```
# extract_unbundled
```

Run phases 1, 2, and 3 of the UX installer program, skipping phase 4 and parts of 5. See the section "The Symbolics UX Installer Program" and see the section "First Pass of Symbolics UX Installer Program". See the section "Upgrading Genera on a Symbolics UX System".

4. Boot the Genera 8.1 distribution world on the Symbolics machine.
5. Make a site-configured Genera 8.1 world for your site. See the section "Making a Genera Site-Configured World".

CD-ROM Installation Steps - an Overview

1. *(Sites with MacIvories only)*
Install the Genera 8.3 MacIvory Macintosh software on all MacIvories which will run Genera 8.3. Since installing this software from CD-ROM requires a CD-ROM drive on the local MacIvory, we suggest that you install this from the Genera 8.3 MacIvory floppies. See the section "Installing the MacIvory Diskettes".
2. *(Sites with UX Family machines only)*
Install the Genera 8.3 UX UNIX software on all UX Family machines which will run Genera 8.3. Run the UX UNIX software installation tool, to install the Genera 8.3 UX UNIX software from the tape. See the section "Upgrading the Symbolics UX UNIX Software".
3. Restore the Genera 8.3 Distribution worlds from CD-ROM, see the section "Restoring the Genera Distribution Worlds From CD-ROM".
4. Make a site-configured Genera 8.3 world for each machine-type. See the section "Making a Genera Site-Configured World".

If your CD-ROM drive is connected to a MacIvory or an XL running Genera 8.3, it is possible to use the CD-ROM as a SYS host. If you wish to store any portion of the Genera 8.3 documentation, sources, examples, fonts, and non-loaded systems on CD-ROM instead of in a LMFS, See the section "Using a CD-ROM as a SYS Host" and use that section as a guide when updating your sys.translations file.

5. If there are any Genera 8.3 documentation, sources, examples, fonts, or non-loaded systems that you do *not* plan to access from CD-ROM, you must restore them into a Lisp Machine File System (LMFS) or a UNIX File System. For instructions on how to do this, see the section "Restoring the Genera Sources".
6. On each machine that is now running a site-configured Genera 8.3 world, copy the new flod files. See the section "Copying New Flod Files" for each machine type. **Note:** The Flod files for 3600-family machines are the same as in previous Genera releases. Therefore, this step is not necessary on 3600-family machines.
7. Cold boot each machine with the new flods.

8. Make customized user and server worlds for your site. See the section "Making Customized General User and Server Worlds".
9. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

Miscellaneous CD-ROM Information

Recommended Care and Handling of a 3M CD-ROM

The CD-ROM itself is a piece of plastic worth several dollars. The value of the CD-ROM is in the information which it stores. The CD-ROM is resistant to damage from scratches, fingerprints, and dust. It is not, however, indestructible. To assure the information's integrity, 3M recommends observing the following:

1. Handle the disc by the edges to prevent fingerprints on either surface of the disc. When removing the disc from the jewel box, be sure to press down on the middle hub of the jewel box to release the disc. Do not pull up hard enough on the disc to bend it more than a half (.5) inch before it releases. Place the disc carefully into the CD-ROM drive, making sure not to scratch the disc against the drive or any other obstruction.
2. When the disc is not in use, place it in its jewel box. This prevents scratching of the disc's surfaces. The information is covered by protective layers, but it is possible to scratch through the label and damage the information. Placing an unprotected disc flat on a desk or table is inviting trouble.
3. Dust is attracted to the surface of a CD-ROM. This dust can affect the readability of the disc. Remove dust from the unlabeled side of the CD-ROM before loading it into the drive. Dust carried into a drive can also be deposited on the lens assembly, which can affect readability of subsequent discs on the system.
4. If the disc becomes soiled with fingerprints or spots, wipe it with a soft dry cloth. If the cloth alone does not remove the spot, attempt to fog the disc with your breath and rewipe. For severe spots resisting removal by this method, a small amount of ethyl alcohol can be placed on the cloth to dampen it. Using the dampened cloth, wipe the area. This method should only be used if the disc is unusable in its present state.
5. No abrasive or solvent cleaners should be used on the disc. Do not use conventional vinyl record cleaning solutions because they contain chemicals that can damage a CD-ROM.

Ejecting the CD From the Drive

- On an NXP or an XL, press the eject button on the front of the CD-ROM drive.

- On a MacIvory, use the Eject menu item in the File menu of the Macintosh Finder.
- On a Sun, use the *umount* command to unmount the CD-ROM drive, then press the eject button on the front of the CD-ROM drive. For more information on the UNIX *umount* command, see the SunOS Reference Manual entry on *umount*.

Site Configuration: A Conceptual Overview

Site configuration enables your Symbolics computers to describe and access the resources available to them. Once your site has been configured, all of its Symbolics computers can find out about the other hosts, printers, and users there.

Namespace service is at the heart of site configuration. In order for one machine (a local host) to use any of the resources provided by other machines (remote hosts), namespace service — managed and provided by a namespace server — is required.

A local host depends on the namespace server for answers to these questions:

- How is the remote host connected to the local host (what is the remote host's network address)?
- What network protocol must be used to obtain the desired service?

By configuring your site, you give each Symbolics machine sufficient information to know where and how to obtain namespace service. You also give your namespace server sufficient information to provide that service.

More information is available about the terms we use when discussing site configuration. See the section "Glossary of Terms for Site Configuration".

Site Configuration and Namespace Service

The namespace server uses namespace files to describe each resource at a particular site. The namespace server might not store the namespace files locally, but it knows where to locate them.

If your site is large, with many (over ten) user machines to make demands on the namespace server, Symbolics recommends that you create a dedicated namespace server; use one Symbolics machine that's unavailable for user applications. If your site is small (under ten user machines), designate one of the user machines as the namespace server.

The namespace server's purpose is to collect and maintain information for a site. All of the information known about a site's network(s) and each host, printer, and user is stored in the namespace database. More information is available about namespace objects and the namespace system. See the section "Setting Up and Maintaining the Namespace Database".

There can be more than one namespace server at a site. One server is the primary namespace server; the others are secondary namespace servers. More information is available about the differences between server machines. See the section "Machines and Worlds".

A typical Symbolics site uses a namespace server to store the namespace database, and a file server to store system sources and online documentation. It is possible for one machine to perform both these services, provided it has enough disk space.

There are some restrictions pertaining to servers. The namespace server must be a Symbolics computer. The system sources and online documentation must reside on one of the following:

- A Symbolics computer.
- Any UNIX host running NFS, in a .sct directory (see the section "Using SCT with a UNIX File System").

More information is available about system sources and online documentation. See the section "System Sources and Online Documentation".

Choosing a Site Name

As part of the site configuration process, you'll select a site name that designates the physical locale of your machines. (This is different from a host name. Host names designate particular machines at a given location.)

When you are ready to configure your site, choose a site name that — in all probability — will not be duplicated by another site. For example, if you are setting up a site at ACE Computers in Nutley, New Jersey, and your department works on robotics, you might call your site ACE-Robotics or ACE-Nutley rather than just ACE.

Note: You cannot use a dot [.] as part of the site name that you choose.

Choosing Machine Names and Network Addresses

It is easier to refer to a computer by its name than by its Chaosnet (the Symbolics network) or Internet address. It is possible to refer to a host by its Chaosnet address, but this can become tedious.

Most customers choose a theme by which to name their machines. For example, each Symbolics computer at your site might be named after different mountains, or famous scientists.

Most sites have all of their machines on the same Ethernet cable. When you create a site, you select a subnet number for this cable. You also select a unique number for each machine. If you have machines on more than one Ethernet cable, you select a unique number for each cable.

If you use networks such as NFS, DNA or IP-TCP along with Chaosnet, we suggest that when you configure your site, you employ an addressing scheme that will encompass all of these. More information is available about network addressing schemes, see the section "Network Addressing".

Note: XL-family, 3600-family, UX-family and MacIvory machines come from Symbolics with Chaosnet address 52525. This is almost never what you want. In fact, if a site already has one machine using that address, the new one will be confused with the old one. NXP1000 machines come pre-configured with the Internet address that you specify at the time of purchase. If you did not specify one, you must boot your machine by hand when you install it, see the section "Caveats About NXP Installation".

Format of Chaosnet Addresses

A Chaos address is a 16-bit quantity, in which the high-order 8 bits represent the subnet number, and the low-order 8 bits represent the host number on that subnet. Neither the subnet number nor the host number can be zero. Chaos addresses are expressed in octal.

Example: Chaos Address 401

```

      15 14 13 12 11 10 9  8  7  6  5  4  3  2  1  0
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|<-----Subnet number----->|<-----Host number----->|

```

The subnet number is 1.

The host number is 1.

The Chaos address is 401 octal.

For technical details on how the Chaosnet address is used: See the section "Chaosnet Addresses and Indices".

Chaosnet addresses are displayed in octal, and each Chaosnet address at a site must be unique. For example, if you have a machine whose cable (subnet) number is 1 and whose machine (host) number is 2, the resulting Chaosnet address is 402.

The lowest Chaosnet address that you can have is 401. Symbolics suggests that you use this number (401) as the Chaosnet address for your first machine, and that you number subsequent machines in ascending order.

Format of Internet Addresses

Internet addresses are expressed in decimal, in four octets separated by periods. Each octet is 8 bits long. There are three kinds of Internet addresses: Class A, Class B, and Class C.

Examples of Internet addresses:

- 10.2.0.7 is host 2.0.7 on Class A network 10.
- 139.41.0.3 is host 0.3 on Class B network 139.41.
- 192.10.0.200 is host 200 on Class C network 192.10.0.

Note that the host number cannot be zero or 255, because those are considered broadcast addresses.

Interpreting Internet Addresses

Internet addresses consist of network and host fields. The network field identifies the network, and the host field identifies the host on that network. This size of the Internet address depends on the address and the configuration of the network.

You can use a subnet field for networks containing subnets. Using a subnet field divides the address into three fields. A subnet mask determines the bits used for selecting a subnet. Note that the rules for determining a subnet field vary for each network.

Note: You cannot fill a field (network, subnet, or host field) with all zeros or ones for representing a network, subnet, or host.

Class A Addresses

A Class A Internet address is a 32-bit number, in which the high-order octet (8-bits) represents the network number and the following three octets represent the host number. The first octet is less than 128.

Example of Class A Internet Address: 10.2.0.7

```

+-----+-----+-----+-----+
|00001010|00000010|00000000|00000111|
+-----+-----+-----+-----+

|<-net-->|<-----host----->|

```

Class B Addresses

A Class B Internet address is a 32-bit number, in which the two high-order octets represent the network number and the following two octets represent the host number. The first octet of a Class B network is greater than or equal to 128 and less than 192.

Example of Class B Internet Address: 139.41.0.3

```

+-----+-----+-----+-----+

```

```

|10001011|00101001|00000000|00000011|
+-----+-----+-----+-----+
|<---network----->|<-----host----->|

```

Class C Addresses

A Class C Internet address is a 32-bit number, in which the three high-order octets represent the network number and the low-order octet represents the host number. The first octet of a Class C network is greater than or equal to 192, and less than 224.

Example of Class C Internet Address: 192.10.0.200

```

+-----+-----+-----+-----+
|11000000|00001010|00000000|11001000|
+-----+-----+-----+-----+
|<-----network----->|<-host->|

```

Internet Subnet Number

The Internet subnet number is the Internet address resulting from replacing the host field of an Internet address with zeros. You can determine the Internet subnet number of a network by determining the class of a host address and replacing the host portion of the address with zeros. For example, the class B address 128.81.38.232 corresponds to the Internet Subnet number 128.81.0.0.

Subnet Masks

A subnet mask determines the field of the Internet address specifying the subnet on the network. A subnet mask is a 32-bit quantity containing one in every bit corresponding to the official Internet subnet number. Additionally, the subnet mask contains a zero in every bit selecting a host on a subnet. For example, A class B network (128.81.0.0) is broken into many subnets, using the third octet of the address for selecting a subnet. The Class B default mask is 255.255.0.0; since the third octet determines a subnet, you have to fill it with ones. The resulting subnet mask is 255.255.255.0.

Mapping an Internet Address into a Chaos Address

Once you have an Internet address for a host, you can map that address into a Chaos address. You can then assign sequential Chaos addresses for all Chaos hosts on the network. If you are on the Internet, you can use each host's Internet address to derive a Chaos address.


```
|<-----Subnet number----->|<-----Host number----->|
```

The resulting Chaos address is 4403 octal.

Logical Pathnames and the SYS Host

A logical pathname is one that does not correspond to any particular physical file system on a host. Logical pathnames make it easy to keep software on more than one type of file system.

For example, the set of files containing the Symbolics system sources and online documentation system is stored at each site. Some sites store these files on a Lisp Machine File System (LMFS), others store them on a VAX/Berkeley UNIX host with the Chaosnet package, and still others use a UNIX host running NFS in a .sct directory (see the section "Using SCT with a UNIX File System"). It is also possible to use a VAX/VMS file server running DNA. More information is available about using VAX hosts. See the section "Site Configuration and Namespace Service".

Symbolics software uses logical pathnames. All sites create a logical host (called *SYS*). Logical pathnames and the logical *SYS* host allow software to work correctly (and the same way) at every site. All pathnames for system software files are logical, and all begin with the logical host *SYS*. Only the translation of each logical pathname to a physical pathname differs at each site.

The translation of logical to physical pathnames depends on the translations files loaded into the current world. For more information about the translations files, see the section "Pathname Translation".

A site that stores the system software on a UNIX system translates logical pathnames into UNIX pathnames. A site that stores the system software on a LMFS translates logical pathnames into LMFS pathnames.

The flexibility of logical pathnames enables sites to split their logical *SYS* host across several physical hosts. A given physical host might contain some of the system software, but the logical entity called a *SYS* host contains all of it.

Machines and Worlds

When you configure your site, you set up different machines to perform specific functions. Each type of machine (user, server, or namespace server) requires a different kind of "world".

User machines

User machines may or may not have special software loaded into their worlds. The distinguishing characteristic of user machines is that they rely on other machines for specific services, such as file service (including access to the online documentation), printer service, and mail delivery.

You can build special worlds for user machines if you want them to run special software (software that is not part of the distribution world). For example, you may load Symbolics non-loaded systems such as Conversion-Tools or Metering into the worlds that you run on the user machines.

Server Machines

Server machines are those machines designated to provide network services to many — or all — of the machines at a site. Typically, server machines provide the following services:

- **File storage:** They have a large file system (the LMFS) where many users store files.
- **Print spooling:** They receive requests to hardcopy information, queue the information, and print it on one or more printers.
- **Store-and-Forward Mail Service:** They receive mail messages and store them locally in mail files for users, or forward them over the network.

Any machine can offer these services. However, Symbolics recommends that a machine upon which many (more than ten) user machines depend for services be used only as a server, and not also as a user machine.

Often, servers run software that is not part of the distribution world. In particular, the Print and Mailer systems have to be loaded in order to provide printer and store and forward mail services, respectively.

To avoid loading these systems every time a server is booted — and to improve the server's performance — you will want to save separate server Incremental Disk Save (IDS) Worlds. You can then Optimize and Save the server worlds that contain whatever non-standard software systems your servers run.

For information about IDS Worlds, the Optimize World command, and the Save World command, see the section "Making, Distributing, and Using Worlds".

Namespace Servers

Primary namespace servers read and write the permanent files of the namespace database. These files can be stored on any file server on the network. However, namespace update operations will run much faster if you store the namespace files on the primary namespace server's local LMFS. (This is only possible if the machine has enough disk space to store the files.)

Secondary servers are not namespace authorities, but they can serve as backups in case the primary server is temporarily unavailable. These servers attempt to keep a copy of the namespace information current by querying the primary server more often than a non-server machine would.

Since namespace servers load namespace changes into virtual memory at boot time, booting on a namespace server will take progressively longer as your namespace grows.

To save time, Symbolics recommends that you periodically create updated namespace server worlds. If you boot your most current site-configured world on a namespace server, all of the newest namespace information will (automatically) be loaded. When you save the newly booted world, you'll get a permanent copy of the up-to-date namespace information.

Remember that each namespace server loads the namespace information only about (and for) those machines that it serves. If you have multiple namespace servers — and they serve different namespaces — you'll need to boot and save a different, updated world on each.

The Distribution World

Core software, containing the Lisp language and all of the programming tools including the Zmacs editor, the Lisp Machine File System, the Zmail electronic mail facility, Document Examiner and others, is contained within the distribution world. This world is formally known as some version of *Genera*.

When you receive a distribution world from Symbolics, it is without any customizations. Once you have loaded the distribution world onto a machine, you can make modifications to it (such as loading special systems or software) to customize it for your site. More information is available about customizing worlds for user, server, and namespace server machines. See the section "Machines and Worlds".

Non-Loaded Systems

Systems are collections of files that — when loaded together — affect the Lisp environment in useful ways. For example, if you load the Print system into the Lisp environment of your machine, your machine will then have the capabilities of a print spooler. It will take hardcopy requests from other hosts and print them, one at a time, on a local printer.

Some systems come already loaded into your world by Symbolics; these loaded systems make up what is called the distribution (or base) world. Document Examiner is an example of a system that comes to you as part of the distribution world.

Other systems aren't pre-loaded into the distribution world. These non-loaded systems provide additional facilities that might only be needed by some Symbolics computers. An example of a non-loaded system is the Store-and-Forward Mailer.

Most systems that Symbolics ships contain several files, and optionally include some patches. The patches are modifications that are too small to warrant compiling the entire system.

Each system has a major and minor version number. The major version number indicates how many revisions and recompilations the system has had. The minor version number indicates how many patches were made to the system since the last revision or recompilation was made to it.

More information is available about loading systems. See the section "Load System Command".

System Sources and Online Documentation

Symbolics distributes a portion of the sources from which the Genera environment was developed, including all of the documentation database.

If you set up one or more machine(s) as server(s) for these files, then you have access to a portion of the system sources. This greatly enhances your development environment and facilitates your development of applications.

You can access the Genera sources from anywhere on the network, without using any disk space other than that on the server machine. For more information: See the section "Logical Pathnames and the SYS Host".

In the Genera environment, all of the printed documentation is available online. You need only to select the Symbolics online documentation system, called Document Examiner, or to execute a few simple commands, in order to view any part of the documentation for the system.

To use Document Examiner, press `SELECT D`. You can get a list of Document Examiner commands by clicking Left on Help. You can get a list of topics to look up by using the Show Candidates command with several words of the general subject you want to find. You can display documentation by using the Show Documentation command on a known topic name or by clicking Left on a topic name in the candidates pane.

Pruning the Documentation Database

The documentation database consists of many files in a large directory on a file server. At some sites, file server space is limited and not all of the documentation files can be kept online. If this is the case at your site, you can selectively remove some of the documentation while still leaving users with online access to commonly used documentation.

Pruning affects only the file space used on the documentation server; it does not affect the part of the document database that resides in the worlds on individual machines. Document Examiner commands (such as Show Candidates and Show Overview) still find the names of sections that are in documents whose files have been deleted.

When a user tries to look up a topic from a removed book, Document Examiner displays a notice that the topic appears in a book that has been removed from the online set. (The notice shows the book name and the name of the file that contains that particular section.)

If users request it, you can restore files from tape to the documentation directory. The material in those files will be available for lookup as soon as they have been restored.

Documentation Database Directories

The directory `SYS:DOC:INSTALLED-N`; contains the set of directories that hold the documentation files. Because of the hypertext nature of our documentation, there is

not a complete correspondance between a given set of files and a book (some files may contain records for more than one book, or some records may be used by several books). However, as a rough cut to help prune the documentation files, here is a list of the directories as they relate, for the most part, to particular books. This should allow you to remove the directories for those books that you do not use. Simply use `Dired` in `Zmacs` or the `Command Processor` command `Delete Directory` to delete the directories for those books you want to remove. **Note:** When using the `Delete Directory` command be sure that the pathname you specify is a directory, not a file in a directory. In the latter case, the directory one level up is the one that gets deleted. See the section "`Delete Directory Command`".

Books 1, 2, 3, and 4:
Genera Concepts
Genera Workbook
Genera User's Guide
Genera Handbook

cp; hard; lms;
miscf; miscu; miscui
nota; sage; user;

Book 5:
Editing and Mail

conv; fed; zmacs;
zmailc; zmailt;

Book 6:
Program Development Tutorial

clyde; tools; workstyles;

Books 7, 8, and 9:
*Symbolics Common Lisp -
Language Concepts
Programming Constructs
Dictionary*

ansi-cl; arr; char;
clcp; clos; cond;
conversion-tools; data-types; defs;
dictionary; eval; flav;
flow; func; io;
mac; miscf; pkg;
prim; scope; strings;
str;

Books 10 and 11: <i>Programming the User Interface</i> <i>User Interface Dictionary</i>	
	ined; menus; new-uims; uims; scroll; windoc;

Book 12: <i>Program Development Utilities</i>	
	comp; debug; maint; meter; meter-int; miset;

Book 13: <i>Networks</i>	
	netio; nfile; prot;

Book 14: <i>Internals</i>	
	audio; fep; instr; init; int; iprim; proc; sched; stor; tape;

<i>Site Administration</i>	
	fep; file; fsed; pig; site;

Ivory Documentation:	
	ivory; macivory; rpe; ux400; xl400;

Installation Guides:	
	sig;

<i>Genera 8.0 Release Notes</i>	
	rn8-0;

Setting and Defining Sites

This section describes the criteria for using the Set Site and Define Site Commands, and shows the Set and Define Site dialogues. See the section "Set Site Command".

See the section "Define Site Command".

Here are the criteria for determining whether to use the Set Site or Define Site command on a Symbolics 3600-family or Ivory-based machine:

Set Site	If the namespace files have already been created, use the Set Site command. This gives your machine access to the already configured site's resources.
Define Site	If the namespace files do not yet exist, use the Define Site command. This creates a new namespace that you can expand later to include other hosts, sites, users, printers, and networks.

Set Site Dialogue

Issue the Set Site command. If you accept the default ("Get from network"), your machine broadcasts a query to other network machines to learn their site name and the identity of the local machine.

```
Command: Set Site (site name [default Get from network])RETURN
```

If you wish to change your site, or if information about your site is unavailable, enter a site name manually. In the examples that follow, *Downunder* is the name of a sample site.

1. When the local host is the namespace server for your site:

```
Command: Set Site (site name [default Get from network]) downunder
Parameters for site DOWNUNDER
Namespace Name: Downunder
Namespace Server Name: Local
Namespace Descriptor File: HATCH:>sys>site>downunder-namespace.text
<ABORT> aborts, <END> uses these values
```

Figure 6. Local host is the namespace server

If the name of the namespace server provided is "Local" or the name of the local host, the menu adjusts to ask for the name of the namespace descriptor file, since you are saying that the local host is the primary namespace server for the site, as in Figure 6 .

2. If the name of the namespace server is different from the local host's name, the menu adjusts to ask for the network address of the Primary Namespace Server, as in Figure ! .
3. The network must be a primary network for the local machine. The address must be a valid address for the network. Figure ! shows the complete dialog for setting the site when the Primary Namespace Server is not the local host: These are all the parameters needed for setting site when the local host is

```

Command: Set Site (site name [default Get from network]) downunder
Parameters for site DOWNUNDER
Namespace Name: Downunder
Namespace Server Name: Wombat
Namespace Server Address: a pair of a network and an address
<ABORT> aborts, <END> uses these values

```

Figure 7. Namespace server is not the local host

```

Command: Set Site (site name [default Get from network]) downunder
Parameters for site DOWNUNDER
Namespace Name: Downunder
Namespace Server Name: Wombat
Namespace Server Address: CHAOS 12345
<ABORT> aborts, <END> uses these values

```

Figure 8. Complete menu for setting the site when the Primary Namespace Server is not the local host.

not the primary namespace server. Of course, you could just say Set Site with no parameters, since the same server specified explicitly in this last example would presumably respond to the broadcast request from the local host.

4. When the local host is the primary namespace server, some additional parameters may be necessary. Figure 9 shows the dialog when the local host is the Primary Namespace Server. The defaults are set up for the local LMFS.

```

Command: Set Site (site name [default Get from network]) downunder
Parameters for site DOWNUNDER
Namespace Name: Downunder
Namespace Server Name: Local
Namespace Descriptor File: HATCH:>sys>site>downunder-namespace.text
<ABORT> aborts, <END> uses these values

```

Figure 9. Additional parameters needed when the local host is the Primary Namespace Server

If the user provides a name for the local host that is not the current name or "Local". The address of the server will be requested. If the address is that of the local machine, the menu looks like this:

```

Command: Set Site (site name [default Get from network]) Downunder
Parameters for site DOWNUNDER
Namespace Name: Downunder
Namespace Server Name: Wombat
Namespace Server Address: CHAOS 24503
Namespace Descriptor File: HATCH:>sys>site>downunder-namespace.text
<ABORT> aborts, <END> uses these values

```

5. If the descriptor file is not on the local host, the minimal parameters necessary to access the host holding the descriptor file will be required. Note: Due to a bug, descriptor files cannot be stored on UNIX hosts.

See the section "Customizing and Saving Worlds".

Define Site Dialogue

In a Lisp Listener, issue the Define Site command and give a site name. This might be the name of your company, or it can be related to the machine names you have chosen.

```

Command: Define Site (site name) Downunder

```

A menu comes up with with the necessary fields to be filled in, as in Figure ! :

```

Command: Define Site (site name) Downunder
Defining site DOWNUNDER with the local host as the Primary Namespace Server
Namespace Server Name: the name of the primary namespace server
<ABORT> aborts, <END> uses these values

```

Figure 10. The basic Define Site menu

Enter the name of your namespace server. The Define Site dialogue expects an actual name here, so do not enter Local for the Namespace Server name.

After you have entered the name of your namespace server, additional menu slots are displayed, with the appropriate default information filled in. You can, of course, change them.

Namespace Server Name

The name of the local host in the new site.

System File Directory

The physical directory that will correspond to the logical directory SYS:SITE; in the new site. For more information, see the section "Logical Pathnames and the SYS Host".

Namespace Descriptor File

The name of the file that will hold the namespace description that is used to find all the files that make up the namespace database.

```

Command: Define Site (site name) downunder
Defining site DOWNUNDER with the local host as the Primary Namespace Server
Namespace Server Name: wombat
System File Directory: wombat:>sys>site>
Namespace Descriptor File: wombat:>sys>site>DOWNUNDER-namespace.text
Default Login: Lisp-Machine
Host for Bug Reports: wombat
Local Timezone: EDT
Standalone Site: Yes No
<ABORT> aborts, <END> uses these values

```

Figure 11. An expanded Define Site menu

Default Login	The "Lisp name" of the user object "Lisp-Machine".
Host for Bug Reports	The name of the host that will accept bug-report mail. For information about the mailer, see the section "Installing and Configuring the Mailer".
Local Timezone	The abbreviation of the local timezone or an integer indicating the hour offset from GMT. For more information about time zones, see the section "Specifying a Time Zone for Your Site".
Standalone Site	Whether the local site consists of just the local machine or not. A broadcast is used to verify the identity of the local host and get the current time when booting at non-standalone sites. Saying "Yes" means that these broadcasts will not be performed.

Press END to define your site.

If a name other than the Namespace Server Name is used in the Namespace Descriptor File, System File Directory, or Host for Bug Reports fields, the dialog will ask for additional parameters to define these hosts in the new site. This information consists of system-types, addresses, and file protocols used to access files on the hosts.

You are now ready to log in and use the newly configured software. The local host is automatically registered at the newly defined site. You may want to save the new, site-specific version of your world. For more information, see the section "Customizing and Saving Worlds".

Specifying a Time Zone for Your Site

All sites are in some time zone. The time zone translates user-specified time into universal time. Users specify their site's time zone by modifying the time zone at-

tribute of their site object. This can be done during the Define Site dialogue and/or with the Namespace Editor.

When you boot a distribution world, if the machine is unable to get the time from the network and does not trust its calendar clock (typically because the FEP has been reset, or the board has been replaced, or the machine is new) you are first prompted for a local time zone and then for the local time.

It is possible to set your time zone with mnemonic symbols. Besides Eastern Standard Time (EST), Central Standard Time (CST), and Pacific Standard Time (PST), you can use other commonly accepted mnemonics for time zones for all over the world. You can specify a time zone as an offset from Greenwich Mean Time (GMT) as well as by mnemonic name.

Specify time zones west of GMT by their mnemonic if one is defined, or by a four-digit number preceded by a - (for west of GMT) or a + (for east of GMT). Four-digit numbers must be in the range -1200 to 1200 and can end either in 00 or 30.

For example, -0500 means five hours west of Greenwich, and is equivalent to EST. It is also possible to allow time zones on the half hour, like South Australian Standard time, which is 9.5 hours east of GMT. You specify this by either SAST or +0930. See the function **time:timezone-string**.

Daylight Savings Time

Every area of the world has its own rules that control when daylight savings time is in effect. The system "knows about" daylight savings time for time zones in the United States (but not for European countries or other time zones).

If you are in a time zone outside of the United States, you must manually tell the system when daylight savings time is in effect. Do this by using the Namespace Editor to edit your site object.

Edit your site object with the Edit Namespace Object command, using a namespace class of **Site**. Use the mouse to replace the time zone with a new one of your choice.

Note: When you create a site, you are queried for a time zone (this happens during the Define Site dialogue). If your site is in the United States, enter the standard version of your local time zone (like EST), regardless of whether daylight savings is in effect. If you enter your time zone using the Namespace Editor, enter just the standard time zone as well.

Here is a complete list of the time zones that adjust automatically during daylight savings:

<i>Time Zone</i>	<i>Abbreviation</i>
Atlantic Standard Time	AST
Central Standard Time	CST
Eastern Standard Time	EST
Mountain Standard Time	MST
Pacific Standard Time	PST

Yukon Standard Time	YST
Alaska-Hawaii Standard Time	AHST
Hawaii Standard Time	HST

<i>Time Zone</i>	<i>Abbreviation for Daylight Savings</i>
Atlantic Standard Time	ADT
Eastern Standard Time	EDT
Central Standard Time	CDT
Mountain Standard Time	MDT
Pacific Standard Time	PDT
Yukon Standard Time	YDT
Alaska-Hawaii Standard Time	AHDT
New Zealand Daylight Time	NZD

Glossary of Terms for Site Configuration

This section defines some terms related to site configuration.

namespace Every site has a *namespace* associated with it. The namespace is a database in which names of users, hosts, and printers are resolved into their representative objects. (See *namespace database*, below.)

namespace server A *namespace server* maintains the namespace database files (see *namespace*, above). A namespace server also processes queries and updates about the mappings and the objects to which those mappings refer. All hosts chosen as namespace servers must be Symbolics computers.

primary namespace server A namespace has a single *primary namespace server*. This server is responsible for maintaining the files that make up the namespace database, and for controlling the access of users and secondary namespace servers to this information.

secondary namespace server A namespace can have one or more *secondary namespace servers*. These servers answer queries when their information is up-to-date, and they can ask the primary server for up-to-date information. Secondary namespace servers are most useful for large sites (more than 25 machines).

namespace database The namespace database consists of a number of files containing information about objects of various classes. You can create, modify, or delete objects in the namespace database by using the Namespace Editor.

To run the Namespace Editor, click on [Namespace Editor] in the System menu or use the Edit Namespace Object command.

The namespace database usually resides in files located in the SYS:SITE; logical directory.

The default location for the namespace database is on the primary namespace server. The default name for the file that contains the names of the other files making up the database (the namespace description file) is as follows (where SITENAME is the registered name of the site):

```
>sys>site>SITENAME-namespace.text
```

Other files constituting the database would have names such as:

```
>sys>site>SITENAME-namespace-log.text
>sys>site>SITENAME-namespace-changes.text
>sys>site>SITENAME-hosts.text
>sys>site>SITENAME-users.text
>sys>site>SITENAME-objects.text
```

The Edit Namespace Object command can be run on any host at the site. To update a database, both the primary namespace server and the host on which the namespace database resides must be available.

objects and classes The namespace database contains *objects*. Objects can fall within different *classes*. The set of classes in the namespace database is as follows:

```
namespace
site
host
user
network
printer
```

For example, suppose you have a user, CHARLIE, logged in to Symbolics machine TUNA at site FISHERY. The user, host, and site are all registered in the namespace called FISHERY.

- FISHERY is an object of class namespace (because it's the name of the namespace that contains the site FISHERY, the user CHARLIE, and the host TUNA).
- FISHERY (a second object) is a object of class site (because it's also the name of the site).
- TUNA is a host object.
- CHARLIE is a user object.

- namespace object* A *namespace object* is a resource (a namespace, site, host, user, network, or printer) that's registered in a namespace database.
- standalone site* The most common *standalone site* is a single Symbolics machine that is not connected to any network. However, because only Symbolics machines can provide namespace service, a site is also considered standalone if it consists of one Symbolics machine and any number of non-Symbolics-machine hosts.
- An organization with several Symbolics machines, none of which is on a network, constitutes several independent standalone sites; that is, each machine appears as the single Symbolics machine host at a standalone site.
- who-am-i* The *who-am-i* request is part of the procedure Symbolics machines use at boot time to determine if the saved Lisp environment matches the current site.

Special Installation Situations

During site configuration and your software installation, you may encounter special situations. Such situations include:

- Adding non-Symbolics computers to your site.
- Installing software from a CD-ROM drive that is on a non-Symbolics machine (a Sun workstation).
- Storing documentation, sources, examples, fonts, and non-loaded systems on a non-Symbolics computer.

This chapter contains information about these special situations.

Adding Non-Symbolics Computers to Your Site

Once the non-Symbolics hardware and software have been correctly installed (and verified) at your site, use the Namespace Editor (specifically, the Edit Namespace Object command) to add non-Symbolics machines to your namespace database. More information is available about the Namespace Editor.

See the section "Using the Namespace Editor".

- If another host at your site is already running the same operating system as the new host, use the [Copy Object] menu item in the Namespace Editor window. Be sure to change the network address(es) and name(s), though.
- If no other host at your site is already running the same operating system as the new host, use the [Create Object] menu item in the Namespace Editor win-

dow. For information about the attributes of a new host, see the section "Attributes for Objects of Class "Host"".

Using a CD-ROM as a SYS Host

In order to use a CD-ROM as a SYS host, the CD-ROM drive must be connected to one of:

- an NXP1000
- an XL Family machine running Genera 8.1.1 or later
- a MacIvory running Genera 8.1.1 or later
- a Sun-4

Note: Genera 8.1 is not sufficient for a Symbolics machine to serve as a CD-ROM SYS host.

The procedure for setting up your site to use a CD-ROM SYS host is as follows:

1. Copy everything from the >SYS>SITE> directory on the CD-ROM to the SYS:SITE; directory for your site. The SYS:SITE; directory for a site is specified in the Site namespace object.

```
Copy File HOST|CDROMn:>SYS>SITE>*.;*;* SYS:SITE;*.;*.*
```

Note: If your CD-ROM drive is on a Sun, this step must be executed in a Genera 8.3 world as follows:

```
Copy File sun-host:/cdrom/sys.sct/site/* sys:site;*.;*.*
```

2. Edit your SYS:SITE;SYS.TRANSLATIONS file to translate logical pathnames to physical pathnames of files on the CD-ROM. After making the necessary changes, save and load the new sys.translations file, or build it into your site-configured Genera 8.3 worlds. Once the updated translations are loaded, you should be able to access files on CD-ROM as easily as you would access files on a LMFS.

In the following examples, *HOST* is the name of the NXP, XL, or MacIvory CD-ROM host, and *n* in CDROM*n* is the SCSI address of the CD-ROM drive.

If all machines at your site are running Genera 8.3, and if your CD-ROM drive is connected to an NXP, XL, or MacIvory, use the following example translations as a guide:

```
;;; -*- Mode: LISP; Syntax: Common-lisp; Package: USER -*-
(fs:set-logical-pathname-host "SYS" :translations
 '(("sys:**;*.;*.*" "HOST|CDROMn:>SYS>**>*.;*.*")))
```

If your CD-ROM drive is connected to a Sun, use the following example translations as a guide:

```
;;; -*- Mode: LISP; Syntax: Common-lisp; Package: USER -*-
(fs:set-logical-pathname-host "SYS" :translations
  '(("sys:**;*. *.*" "sun-host:/cdrom/sys.sct/**/*")))
```

If some machines at your site are running a release older than Genera 8.3, use the following example translations as a guide:

```
;;; -*- Mode: LISP; Syntax: Common-lisp; Package: USER -*-
(multiple-value-bind (major-version minor-version)
  (sct:get-release-version)
  (select major-version
    ;; Translations for Genera 8.1, 8.1.1
    (8 (selector minor-version equalp
      ("1.1")
      (fs:set-logical-pathname-host "SYS" :translations
        '(("sys:**;*. *.*" "ACME-SMBX:>rel-8-1>sys>**>*. *.*")))))
      ("2 NXP")
      (fs:set-logical-pathname-host "SYS" :translations
        '(("sys:**;*. *.*" "HOST|CDROMn:>sys>**>*. *.*;*"))))
      (3)
      (fs:set-logical-pathname-host "SYS" :translations
        '(("sys:**;*. *.*" "HOST|CDROMn:>sys>**>*. *.*;*"))))
      (otherwise (error "~A ~A unknown release or system version.~2T
        Edit SYS:SITE;SYS.TRANSLATIONS to add this system."
          major-version minor-version))))
    (otherwise (error "~A ~A unknown release or system version.~2T
      Edit SYS:SITE;SYS.TRANSLATIONS to add this system."
        major-version minor-version))))))
```

If you wish to use the CD-ROM as the SYS host for only specific parts of the Genera 8.3 documentation, sources, examples, fonts, and non-loaded systems, use the following example translations as a guide (This example assumes that all Symbolics machines at your site are running Genera 8.3):

```
;;; -*- Mode: LISP; Syntax: Common-lisp; Package: USER -*-
(fs:set-logical-pathname-host "SYS" :translations
  ;; Access documenatation and fonts from the CD
  '(("sys:doc;**;*. *.*" "HOST|CDROMn:>SYS>DOC>**>*. *.*;*")
    ("sys:fonts;**;*. *.*" "HOST|CDROMn:>SYS>FONTS>**>*. *.*;*")
    ;; Access everything else from a LMFS on a file server
    ("sys:**;*. *.*" "SMBX:>rel-8-3>sys>**>*. *.*")))
```

Known limitations and problems with using a CD-ROM SYS host

1. The ISO9660 filesystem does not understand unusual characters such as +, \$, and & in pathnames. There are a few pathnames in the Genera distribution which contain such characters. If you run into one, simply take the retry op-

tion to specify another pathname, and respecify the same pathname, but drop the offending character. For example,

```
SYS:EMBEDDING;MACIVORY;TOOLBOX;LISP+C-SCT-SUPPORT.BIN appears as
SYS:EMBEDDING;MACIVORY;TOOLBOX;LISPC-SCT-SUPPORT.BIN
```

in the CD-ROM filesystem.

2. The ISO9660 filesystem limits the length of file and directory names to 31 characters, including periods and file types. File names longer than this are truncated. There are many truncated pathnames in the Genera distribution, however only three files produced ambiguous pathnames when truncated. On the Genera CD-ROM, we renamed these three files to avoid truncation problems:

```
SYS:EXAMPLES;CONSTRAINT-FRAME-LANGUAGE-1.LISP
SYS:EXAMPLES;CONSTRAINT-FRAME-LANGUAGE-2.LISP
SYS:EXAMPLES;CONSTRAINT-FRAME-LANGUAGE-3.LISP
```

are named

```
SYS:EXAMPLES;CONSTRAINT_FRAME_LANGUAGE_.LISP
SYS:EXAMPLES;CONSTRAINT_FRAME_LANGUAG_2.LISP
SYS:EXAMPLES;CONSTRAINT_FRAME_LANGUAG_3.LISP
```

Note that the ISO9660 translations rules automatically do the correct thing for all other truncated pathnames on the CD-ROM. For example, Genera automatically translates

```
SYS:METERING;INTERFACE;STATISTICAL-CALL-TREE-METERING-RUN.IBIN
```

to the correct CD-ROM pathname:

```
CDROMn:>sys>metering>interfaces>statistical_call_tree_mete.ibin
```

3. When using a CD-ROM as a SYS host, you must reset access paths whenever you change CD disks.

Storing Genera Sources on a Non-Symbolics Machine

You can use a UNIX machine running NFS as a file server to store your Symbolics sources.

Use this example for your SYS.TRANSLATIONS file:

```
;;; -*- Mode: LISP; Syntax: Common-lisp; Package: USER -*-
(fs:set-logical-pathname-host "SYS" :translations
'(("sys:**;" "ACME-UNIX:/usr/share/symbolics/re1-8-4/sys.sct/**/")))
```

If you are running multiple releases (for example, some combination of Genera 8.3 and Genera 8.4), use the following example:

```

;;; -*- Mode: LISP; Syntax: Common-lisp; Package: USER -*-
(multiple-value-bind (major-version minor-version)
  (sct:get-release-version)
  (select major-version
    ;; Translations for Genera 8.1.1, 8.2, and 8.3
    (8 (selector minor-version equalp
      (("3") (fs:set-logical-pathname-host "SYS" :translations
        '("sys:**;"
          "ACME-UNIX:/usr/share/symbolics/rel-8-3/sys.sct/**/")))))
      ((4)
        (fs:set-logical-pathname-host "SYS" :translations
          '("sys:**;"
            "ACME-UNIX:/usr/share/symbolics/rel-8-4/sys.sct/**/")))))
      (otherwise (error "~A ~A unknown release or system version.~2T
        Edit SYS:SITE;SYS.TRANSLATIONS to add this system."
          major-version minor-version))))))
  (otherwise (error "~A ~A unknown release or system version.~2T
    Edit SYS:SITE;SYS.TRANSLATIONS to add this system."
      major-version minor-version)))

```

Using SCT with a UNIX File System

The System Construction Tool (SCT) presents a special case in using UNIX backup files. SCT requires a version number for the files it uses. For directories maintained by SCT, files are maintained so that the newest versions of the files have an explicit version number. (See the section "System Construction Tool" for background information on SCT).

Use the name of a UNIX directory to specify that files contained in it are to be maintained by SCT and should all end with explicit version numbers. If the name of a UNIX directory ends with the .sct file extension, Symbolics NFS creates all new files under that directory with explicit version numbers.

Here's an example. Suppose a sys.translations file looked like this:

```

;;; -*- Mode: LISP; Syntax: Common-lisp; Package: USER -*-

(fs:set-logical-pathname-host "SYS" :translations
  '("**;" "ACME-UNIX:/usr/share/symbolics/rel-8-0/sys.sct/**/"))

```

In this case, all new files created with Symbolics NFS under /usr/share/symbolics/rel-8-0/sys.sct/ are created with explicit version numbers in the newest versions of files, meeting SCT requirements. This is true for all sub-directories of a directory whose name ends in .sct.

The double-asterisk (**) wildcard can be used only in UNIX pathnames that are accessed by means of NFS. Other UNIX file protocols, such as FTP, do not recognize this wildcard. For further information on the use of the double-asterisk wildcard, see the section "LMFS Pathnames".

Genera considers UNIX pathnames with any directory components ending in the string ".sct" to be for SCT-maintained files, and hence to contain version numbers. The syntax that Genera uses for backup file version numbers in UNIX pathnames is a number surrounded by tildes separated by a dot from the file type.

This pathname is considered to have a version number:

```
ENIAC:/u0/mwra/visi-brain.sct/foo.lisp.~23~
```

So merging with it as a default behaves like this:

<i>Default</i>	<i>Input</i>	<i>Merged</i>
ENIAC:/u0/dpw.sct/code/foo.lisp.~23~	bar.⌘	ENIAC:/u0/dpw.sct/code/bar.lisp
ENIAC:/u0/dpw.sct/code/foo.lisp.~23~	⌘.bin	ENIAC:/u0/dpw.sct/code/foo.bin

Note that backup file version numbers are not defaulted as part of the merging operation.

You can set the number of backup versions that are retained by Genera with the "NFS-GENERATION-RETENTION-COUNT User Property" for host namespace objects.

Contents of Hello.Boot Files and Boot.Boot Files

This section briefly describes the philosophy of which FEP commands belong in the Hello.boot file and which belong in the Boot.boot file on both 3600-family and Ivory-based machines.

Hello.boot File

This file should contain anything that holds constant for this machine, regardless of the world you are running. For example, this includes your flod files, your paging files, your network address, and so on.

First, this file must scan all the flods from which you want to use commands. This includes the lisp, loaders, info, and debug flods.

Next, the file should include the Initialize Hardware Tables command, so that any errors it encounters will be seen early in the booting process. (This command is completely optional in Ivory-based machines. It is automatically invoked by the Ivory FEP when needed, but errors it signals may be less clear when automatically invoked.)

Then, the Hello.boot file should declare any paging files you always want to use; this saves you having to manually Add Paging Files, or to say the same in your boot file. Do not declare any "spare" paging you want to save for an emergency, because Load World automatically uses all declared paging files, and thus the spare paging file would not be available in case of an emergency. It is harmless to declare paging files that do not exist (for example, one you might delete every now and then to make room for Copy World); nonexistent ones give a warning, but are not used.

Ivory machines only:

Starting with Genera 8.1, the `hello.boot` file for an Ivory machine should contain two lines:

```
Hello Innn
Hello Local (or hostname)
```

Each of these corresponds to a boot file. The *nnn* refers to the IFEP number, which is 328 for Genera 8.3.

Hello Innn Boot File

The `Innn.boot` file (where *nnn* is the IFep version number, which is 328 for Genera 8.3) should contain the commands to scan the flod files and initialize things.

```
Scan I328-lisp.flod
Scan I328-loaders.flod
Scan I328-info.flod
Scan I328-debug.flod
Initialize Hardware Tables
```

Hello Local Boot File

The `local.boot` file should contain those commands that set up this specific machine, declaring paging files, setting the network address, and any other boot options.

```
Declare Paging Files FEP0:>Paging-1.page
Declare More Paging Files FEP0:>Paging-2.page,Paging-3.page
Set Boot Options :Network Address Chaos|52525 :IDS Enable
```

Boot options are required for standalone sites. They are optional for other sites, but they do save manually invoking them or having them duplicated or inconsistent in boot files (if you have multiple boot files). See the section "Set Boot Options FEP Command".

The boot options replace other FEP commands. For compatibility, some of the boot options have corresponding commands. Note that some of these commands (such as Enable/Disable IDS on 3600-family machines), if present, must appear in the `Boot.boot` file, since they must follow Load World. (In the Ivory-based FEP, these commands warn if you try to use them at the wrong time, while in the 3600-family FEP they silently perform no operation if used at the wrong time). We recommend using the Set Boot Options command instead of the corresponding commands, so you can set all the boot options in one spot.

Normally, you set your network address with Set Boot Options, but you could use the Set Network Address command instead. Whichever way you choose to set the network address, it should be done in the `Local.boot` file, and should not be duplicated in your `Boot.boot` files.

3600 and Ivory Machines

Note that, in previous releases, you had to set the network address after loading the world, but this restriction no longer holds on either architecture.

Here is a sample Hello.boot file for a 3600-family machine:

Sample 3600-Family Machine Hello.Boot File

```
Scan v127-info.flod
Scan V127-loaders.flod
Scan v127-lisp.flod
Scan v127-debug.flod
Initialize Hardware Tables
Declare Paging-Files FEP0:>Paging-1.page
Declare More Paging-Files FEP0:>Paging-2.page,Paging-3.page
Set Chaos-Address 52525
```

(For a sample Hello.boot file for an Ivory-based machine, see the section "Sample Ivory-Based Machine Hello.Boot File".)

Boot.boot Files

These files should contain any setting that pertains only to the particular world (and microcode, on a 3600-family machine) this file loads and starts. (Note that this excludes the address of the machine.) There is a slim possibility that you might want to have some explicit paging file commands here (for example, a boot file to boot with just one paging file explicitly added for when you want to do world copying).

Sample 3600-Family Boot.Boot Files

Here is the sequence of commands for a Symbolics 3600-family machine that is cold booting a world from the local disk:

```
Clear Machine
Load Microcode FEP0:>3640-fpa-mic.mic.430
Load World FEP0:>genera-8-3.load
Enable IDS
Start
```

Here is the sequence of commands for a Symbolics 3600-family machine that is cold booting a world from a remote disk (netbooting):

```
Clear Machine
Load Microcode FEP0:>3640-fpa-mic.mic.430
Netboot inc-site-genera-8-3
Enable IDS
Start
```

The information in the rest of this section applies to Ivory-based machines only.

Boot.boot files for Ivory-based machines should contain only the following:

```
Load World
Start
```

Note that you can set your default world to boot in boot options. If you choose to load the default world, there is no need to give an explicit pathname argument to Load World.

It is unnecessary to do a Clear Machine before the Load World since this command does nothing on Ivory-based machines. On 3600 family machines, it serves a purpose. To avoid confusion and make boot files easier to understand and maintain, if Clear Machine is left in the boot file of an Ivory machine, it is ignored.

It is possible, although not recommended, to give the Enable/Disable IDS command here, but the recommended way is to do all that in your Hello.boot file, via Set Boot Options :IDS [Enable/Disable]. You should also use the Set Network Address commands in your Hello.boot file. (Note that if someone copies a boot file with an address in it to another machine, confusion will result.)

Note also that Disable IDS saves only an inconsequential amount of memory and overhead, so there is usually no advantage in doing it.

Here is the recommended sequence of commands for a Symbolics Ivory-based machine that is cold booting a world from the local disk:

```
Load World
Start
```

This loads the most recent world on your local disk, which is usually the one you want to boot.

For a netbooted machine, the recommended sequence is:

```
Netboot inc-site-genera-8-3
Start
```