

Genera 8.1 Software Installation Guide for 3600 Family Machines

CD-ROM Distribution Media for All Sites

All sites who requested CD-ROM distribution media receive the following:

- *Symbolics Genera 8.1, all platforms*, 1 CD, P.N. 995745 (This CD contains Genera 8.1 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.)

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 3600 Family machines also receive:

- *Genera 8.1 Distribution World for 3600-Family Machines*, one QIC-11 tape, IFS Format, P.N. 995730 (This tape contains microcodes and the 8.1 World and Netboot Core for Symbolics 3600 Family machines.)

Overview of CD-ROM Installation

This is 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:
 - 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.

- See the section "Setting up Network Access for a CD-ROM Drive on a Sun Workstation".
- Install the Genera 8.1 Distribution World from CD-ROM onto one Symbolics machine at your site.

When choosing a machine on which to do the initial installation of Genera 8.1, you probably will choose a faster machine for the best response. We recommend the following configurations, in order of preference:

- A Symbolics XL Family machine with a local CD-ROM drive. The XL must also have a local QIC-100 tape drive.
- A Symbolics MacIvory machine with a local CD-ROM drive.
- Any other Symbolics machine, with access to a CD-ROM drive connected to a Sun workstation. In this case, the Symbolics machine must be running IP-TCP or NFS.

If you choose an XL machine, see the section "Initial CD-ROM Installation of Genera on an XL"; for a MacIvory, see the section "Initial CD-ROM Installation of Genera on a MacIvory" ; for other machines, see the section "Initial CD-ROM Installation of Genera From a CD-ROM on a Sun".

- Install Genera 8.1 worlds on the remaining Symbolics machines at your site, and install the Genera 8.1 flods, documentation, sources, examples, fonts, and non-loaded systems. See the section "CD-ROM Installation of Genera on Subsequent Machines".

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 of Genera on Subsequent Machines

1. (*Sites with MacIvories only*) Install the Genera 8.1 MacIvory Macintosh software on all MacIvories which will run Genera 8.1. 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 8.1 MacIvory floppies. See the section "Installing the MacIvory Diskettes".
2. Use the Copy World command to copy the appropriate Genera 8.1 worlds from the initial Genera 8.1 machine to as many other machines as possible. If your initial 8.1 machine is a 3600, copy your 3600 site-configured Genera 8.1 world (and any other 8.1 3600 worlds you wish) to all the 3600 Family machines at your site. Similarly, if your initial 8.1 machine is a MacIvory, XL, or UX, copy your Ivory site-configured Genera 8.1 world (and any other Ivory worlds you wish) to all the Ivory-based machines at your site. Remember that you cannot copy Ivory (.ilod) worlds to 3600 Family machines, and you cannot copy 3600 (.load) worlds to Ivory-based machines.

Note: If your initial Genera 8.1 machine is a 3600 netboot server, you may wish to just copy the appropriate netboot core to the other 3600 Family ma-

chines at the site. Remember that the 3600 netboot core will only work on 3600 Family machines. You cannot netboot Ivory-based machines at this point, because Ivory netboot will not be enabled until you install the Genera 8.1 IFEP later in this installation process.

3. Boot a site-configured Genera 8.1 world on each machine which is now capable of booting Genera 8.1.
4. For the remaining machines at your site, use the Copy File command to copy the appropriate Genera 8.1 worlds from the "worlds" directory on the CD-ROM to your local FEP filesystem. If the CD-ROM drive is connected to an XL or a MacIvory, the Copy File must be done from a machine running Genera 8.1, because ISO9660 pathname support is required. If the CD-ROM drive is connected to a Sun, you can use Copy File from a Symbolics machine running Genera 8.0, since the Sun provides the ISO pathname support. **Note:** A Symbolics machine must be running IP-TCP or NFS in order to access a CD-ROM on a Sun.

If the CD-ROM drive is connected to an 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_1.1LOD
```

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_1.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 8.1 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_1.load 3600-host|FEP0:>genera-8-1.load
```

For a similar scenario, but with the CD-ROM drive, SCSI address 3, connected to a MacIvory or XL, you would type the following on an 8.1 machine:

```
Copy File CDROM-HOST|CDROM3:>WORLDS>GENERA_8_1.LOAD (to)  
3600-host|FEP0:>genera-8-1.load
```

3600 worlds:

```
netboot_core_from_genera_8_1.load  
genera_8_1.load
```

MacIvory worlds:

```
netboot_core_from_genera_8_1.ilod
genera_8_1.ilod
```

Optionally,

```
genera_8_1_macivory_delivery.ilod
```

XL worlds:

```
netboot_core_from_genera_8_1.ilod
genera_8_1.ilod
```

UX worlds:

```
netboot_core_from_genera_8_1.ilod
genera_8_1.ilod
network_from_genera_8_1.ilod
```

Optionally,

```
genera_8_1_ux_delivery.ilod
```

5. Update your boot.boot files to boot a Genera 8.1 distribution world.
6. (*Sites with UX Family machines only*) Install the Genera 8.1 UX UNIX software on all UX Family machines which will run Genera 8.1. Run the UX UNIX software installation tool, to install the Genera 8.1 UX UNIX software from the tape. See the section "Upgrading the Symbolics UX UNIX Software".
7. Make a site-configured Genera 8.1 world for each remaining 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 8.1, 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.

8. If there are any Genera 8.1 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). To do this, type

```
Restore Distribution :Use Disk Yes :Menu Yes
```

In the Restore Distribution frame, specify a disk pathname of one of the distribution images on the CD-ROM. The possible distribution images are:

If the CD-ROM drive is connected to a MacIvory or XL:

```

HOST|CDROMn:>DISTRIBUTION>GENERA_8_1_SOURCE_DIST_1.TAPE
  (Genera sources and documentation)
HOST|CDROMn:>DISTRIBUTION>GENERA_8_1_SOURCE_DIST_2.TAPE
  (Genera non-loaded systems)
HOST|CDROMn:>DISTRIBUTION>GENERA_8_1_FPC_DIST.TAPE
  (Fortran, Pascal, and C layered products)

```

If the CD-ROM drive is connected to a Sun, and you are running 8.1 with IP-TCP or NFS Client loaded:

```

host:/cdrom/distribution/genera_8_1_source_dist_1.tape
  (Genera sources and documentation)
host:/cdrom/distribution/genera_8_1_source_dist_2.tape
  (Genera non-loaded systems)
host:/cdrom/distribution/genera_8_1_fpc_dist.tape
  (Fortran, Pascal, and C layered products)

```

If the CD-ROM is connected to a Sun, and you are running 8.1 with IP-TCP or NFS Client loaded, specify a pathname that is accessible over Chaos, where you copied the distribution images in "Initial CD-ROM Installation of Genera From a CD-ROM on a Sun".

For each distribution image, click on [Initialize Restoration], use the mouse to deselect any systems you do not want to restore, then click on [Perform Restoration] to restore the selected files from CD-ROM to LMFS. For a list of systems on the Source distribution tapes, and advice about which systems you might need to restore, see the section "Restoring the Genera Sources". Restoring the contents of the FPC distribution image is completely optional.

9. On each machine that is now running a site-configured Genera 8.1 world, copy the new flod files. See the section "Copying New Flod Files" for each machine type.
10. Cold boot each machine with the new flods.
11. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
12. Back up your world(s) to tape. See the section "Backing Up Worlds to Tape".

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:

- (Users upgrading from Genera 8.0 to Genera 8.1 *only*) Restore the Distribution World tape. See the section "Restoring the Genera Distribution World From Tape on a 3600-Family Machine".
- (New Sites *only*) Make a site-configured Genera 8.1 world for your site. See the section "Making a Genera Site-Configured World".

All 3600 sites should do the following:

1. Restore the Genera 8.1 documentation, sources, examples, fonts, and non-loaded systems into your Lisp Machine File System (LMFS). See the section "Restoring the Genera Sources".
2. Copy the new flod files. See the section "Copying New Flod Files to a 3600".
3. Make customized user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".
4. 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 8.1 Distribution World on a 3600 Family Machine

Restore the Genera 8.1 world load files into your FEP file system, using the tape labeled *Symbolics Genera 8.1 Distribution World for 3600 Family Machines*. (Sites with new 3600 family machines shipped from Symbolics should not have to perform this step. The Genera 8.1 world loads should already be on the disk of the new machine.)

1. Insert the *Symbolics Genera 8.1 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, give the command:

Read Tape

The Read Tape command is also available in the FEP-Tape menu, but we rec-

ommend 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-8-1.load
```

The FEP-Tape loading program then loads the world from tape onto disk. For releases where the world spans multiple tapes the system prompts you to insert a new tape when the end of each tape has been reached.

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-8-1.load
Enable IDS
Start
```

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

6. Boot the Genera 8.1 Distribution world. In a Lisp Listener, type

```
Halt Machine
```

In the FEP, type

```
Boot
```

If your site is already set up, after you have restored the world, netboot core (see the section "Netbooting"), and microcode, 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, 3600, or MacIvory Machine".

Completing the Installation

Creating a New Site for Genera

The steps in this section should be performed at new sites that have multiple Symbolics (3600-family, MacIvory, XL, NXP, or UX) machines.

Choosing Names, Addresses, and Servers for the New Site

1. Select a name for the site. For information about choosing site names, see the section "Choosing a Site Name".
2. Select a name for each machine at the site, including non-Symbolics machines.
3. Select a network address for each Symbolics computer. Remember that each machine must have a unique network address. Machines that are not Symbolics computers might already have Internet addresses for IP-TCP networks. If not, select the appropriate network addresses for them. For information about network addresses, see the section "Choosing Machine Names and Network Addresses".

Note: All preconfigured machines come from Symbolics with Chaosnet address 52525. *Unless you have a standalone machine, this is not a desirable Chaosnet address.*

4. Choose a Symbolics computer to be your namespace server. You cannot use a non-Symbolics machine for a namespace server. For information about server machines, see the section "Machines and Worlds".
5. Choose a machine to be the file server and, if necessary, the print server, mail server, and Domain server. Symbolics recommends that you use your namespace server machine for these, provided it has enough disk space. More information is available about choosing server machines. See the section "Machines and Worlds".

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 325 for Genera 8.1.

Hello Innn Boot File

The *Innn*.boot file (where *nnn* is the IFep version number, which is 325 for Genera 8.1) should contain the commands to scan the flod files and initialize things.

```
Scan I325-lisp.flod
Scan I325-loaders.flod
Scan I325-info.flod
Scan I325-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 inconsis-

tent 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-8-1.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-8-1
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-8-1
Start
```

At this point you are ready to boot Genera.

Defining the New Site

This step should be performed at new sites with multiple machines only. Sites with a standalone XL or a standalone MacIvory do not need to define a site.

On the namespace server machine, use the Define Site command to create the initial namespace database. Once you have entered data into this database, the namespace files are stored permanently.

Define Site enables the local machine to access the permanent namespace database files, and at the same time, it creates a host namespace object for the local machine.

Define Site Dialogue for XL, 3600, and MacIvory Machines

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 132:

```
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 134. 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

```

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 135. An expanded Define Site menu

that is used to find all the files that make up the namespace database.

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.

Now you are ready to add your machines to your site.

Registering a New Symbolics XL, 3600, or MacIvory Machine

Choosing a Name and Network Address for the New Machine

Select a name and network address(es) for the new machine.

For information about network addresses, see the section "Choosing Machine Names and Network Addresses".

Note: All preconfigured machines come from Symbolics with Chaosnet address 52525. Unless you have a standalone machine, this is not a desirable Chaosnet address.

Creating a New Host Object for a 3600, XL, or MacIvory Machine

1. Use the Namespace Editor on the namespace server to set up the namespace. For information about using the Namespace Editor, see the section "Using the Namespace Editor".
2. Create a host object for each new 3600-family, XL, and MacIvory machine at the site. The system type for all machines is "LispM". The machine type should be one of "3600", "XL400", "XL1200", or "MacIvory". Supply the appropriate Chaosnet address for each machine. Here is a sample namespace database entry for a 3600-family host object:

```

Site: SCRC
Nickname: SCRC-NUTHATCH
Short Name: HATCH
Machine Type: 3600
System Type: LISPM
Address: CHAOS 24503
Pretty Name: Nuthatch
Console Location: SCRC 2 Ellen's office, 246
Printer: AMERICAN
Bitmap Printer: AMERICAN
Service: CHAOS-STATUS CHAOS-SIMPLE CHAOS-STATUS
Service: CONFIGURATION CHAOS CONFIGURATION
Service: FILE CHAOS NFILE
Service: LISPM-FINGER CHAOS-SIMPLE LISPM-FINGER
Service: LOGIN CHAOS SUPDUP
Service: LOGIN CHAOS 3600-LOGIN
Service: LOGIN CHAOS TELNET
Service: NOTIFY CHAOS-SIMPLE NOTIFY
Service: SEND CHAOS SEND
Service: SEND CHAOS CONVERSE
Service: SHOW-USERS CHAOS NAME
Service: TIME CHAOS-SIMPLE TIME-SIMPLE
Service: UPTIME CHAOS-SIMPLE UPTIME-SIMPLE

```

3. Create namespace objects that represent users, other computers and printers at the site (whether Symbolics or non-Symbolics machines).

Making a Genera Site-Configured World

Note: You must build a site-configured world for each machine type (3600-Family or Ivory) at your site, with the exception of standalone sites that consist of only one MacIvory or one XL machine. We do not recommend saving a site-configured world in which the Define Site command has been issued. When building a site-configured world, use the Set Site command on a freshly booted machine.

Perform these steps to make a site-configured world:

1. Edit (update) the SYS:SITE;SYS.TRANSLATIONS file. For instructions on how to do this, see the section "Updating the SYS.TRANSLATIONS File for Genera 8.1". Once you have updated the translations file, return to this section and finish making the Genera 8.1 site-configured world.
2. Cold boot the Genera 8.1 distribution world on your machine.
3. Use the Set Site command to configure the machine for your site. For information about the Set Site dialogue, see the section "Set Site Dialogue".
4. Save the new world, using either the Save World Complete command or the Save World Incremental command, depending on your circumstances. Information is available to help you make this decision. See the section "Save World Command".

This example shows how to save an incremental version of the world for Genera 8.1 on Ivory-based machines:

```
Save World Incremental FEP0:>Inc-Site-from-Genera-8-1.load
```

This site-configured world will serve as the base world for all Ivory worlds you build at your site, except for the namespace server's world. Similarly, a site-configured 3600 world will serve for all 3600-Family worlds you build at your site (except for namespace server worlds). Information is available about making user and server worlds for your site. See the section "Making Customized Genera User and Server Worlds".

Updating the SYS.TRANSLATIONS File for Genera 8.1

Before making a site-configured world, edit the file SYS:SITE;SYS.TRANSLATIONS so that Genera 8.1 will "know" where its sources reside. Once you have finished editing the SYS.TRANSLATIONS file for your site, save and load the file.

For sites with Symbolics 3600-series machines, MacIvories, and/or XL machines, Symbolics suggests that you use one of those machines to hold the distribution sources. Alternatively, you can use a non-Symbolics machine to store sources. For further information, see the section "Storing Genera Sources on a Non-Symbolics Machine".

For sites that use MacIvories and XL or 3600-series machines, it is preferable to store sources on XL or 3600-series machines, rather than MacIvories. This is be-

cause networks are more reliable and more disk space is generally available with these machines.

If the machines at your site will be running only Genera 8.1, use this example as a model for your SYS.TRANSLATIONS file:

```
;;; -*- Mode: LISP; Syntax: Common-lisp; Package: USER -*-
(fs:set-logical-pathname-host "SYS" :translations
 '(("sys:standalone-site;**/*.*)" "ACME-SMBX:>sys>standalone-site>**/*.*)"
   ("sys:**/*.*)" "ACME-SMBX:>rel-8-1>sys>**/*.*")))
```

If the machines at your site will be running multiple versions of Genera,

see the section "A Sample SYS.TRANSLATIONS File That Stores Genera Sources for Multiple Releases".

A Sample SYS.TRANSLATIONS File That Stores Genera Sources for Multiple Releases

This example shows a SYS.TRANSLATIONS file that has been conditionalized to allow Genera 8.1, and Genera 8.1.1 to coexist with Genera 8.2 NXP. In this example, ACME-SMBX is the name of the Symbolics computer on whose Lisp Machine File System (LMFS) the sources will be stored.

```
;;; -*- 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, and 8.2 NXP
    (8 (selector minor-version equalp
      (("1" "1.1") (fs:set-logical-pathname-host "SYS" :translations
        '(("sys:standalone-site;**/*.*)"
          "ACME-SMBX:>sys>standalone-site>**/*.*)"
          ("sys:**/*.*)" "ACME-SMBX:>rel-8-1>sys>**/*.*"))))
      (("2 NXP" "2 NXP ECO#1") (fs:set-logical-pathname-host "SYS" :translations
        '(("sys:**/*.*)" "ACME-SMBX:>rel-8-2>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))))))
```

Managing System Source Files When Running Multiple Releases

If you are planning to run more than one release at the same time, we recommend keeping sources for all of them. Keeping all sources available may not be possible because of disk space limitations on your SYS host (or hosts), but the lack of certain source files may cause problems. This means that if you wish to delete some of the Genera 8.1 sources, you may have to employ a workaround.

The workaround is to change the `SYS.TRANSLATIONS` file so that Genera 8.1 can find files in the Genera 8.2 directory. *This workaround is limited to Genera fonts, and FEP flod file directories. Do not use the workaround on documentation, X-fonts, Lisp sources, or binaries.*

For example, if you have to delete the Genera 8.1 font files from your `SYS` host, modify the translations file as shown in bold (this translation allows users of Genera 8.1 systems to find screen fonts in the `>REL-8-2>` directory).

```
;;; -*- 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, and 8.2
    (8 (selector minor-version equalp
      (("1" "1.1") (fs:set-logical-pathname-host "SYS" :translations
        '("sys:fonts;*;*.*)"
        "ACME-SMBX:>rel-8-2>sys>fonts>*>*.*)"
        ("sys:standalone-site;*;*.*)"
        "ACME-SMBX:>sys>standalone-site>*>*.*)"
        ("sys:*;*.*" "ACME-SMBX:>rel-8-0>sys>*>*.*)"))))
      (("2 NXP" "2 NXP ECO#1") (fs:set-logical-pathname-host "SYS" :translations
        '("sys:*;*.*" "ACME-SMBX:>rel-8-2>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 are using a non-Symbolics machine to store your system documentation, sources, examples, fonts, and non-loaded systems, you would make such a modification in the appropriate file system syntax.

Once you have made a site-configured world, you are ready to restore the Genera 8.1 documentation, sources, examples, fonts, and non-loaded systems.

Restoring the Genera Sources

This section describes how to restore the Genera documentation, sources, examples, fonts, and non-loaded systems. This step is not necessary for sites that use the CD-ROM as a `SYS` host. See the section "Using a CD-ROM as a `SYS` Host".

Use the CD-ROM labeled *Genera 8.2 NXP World and Sources*.

Restoring the Genera sources in their entirety requires approximately 26,000 LMFS records (equivalent to approximately 104,000 Ivory FEP blocks or 130,000 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 new site-configured Genera world and put the Genera 8.2 NXP CD-ROM into the CD-ROM drive.
2. Use the following command to restore the files from the distribution image on the CD-ROM:

```
Restore Distribution :Menu yes :Use Disk yes
```

The system responds by displaying a menu, as shown in Figure 127.

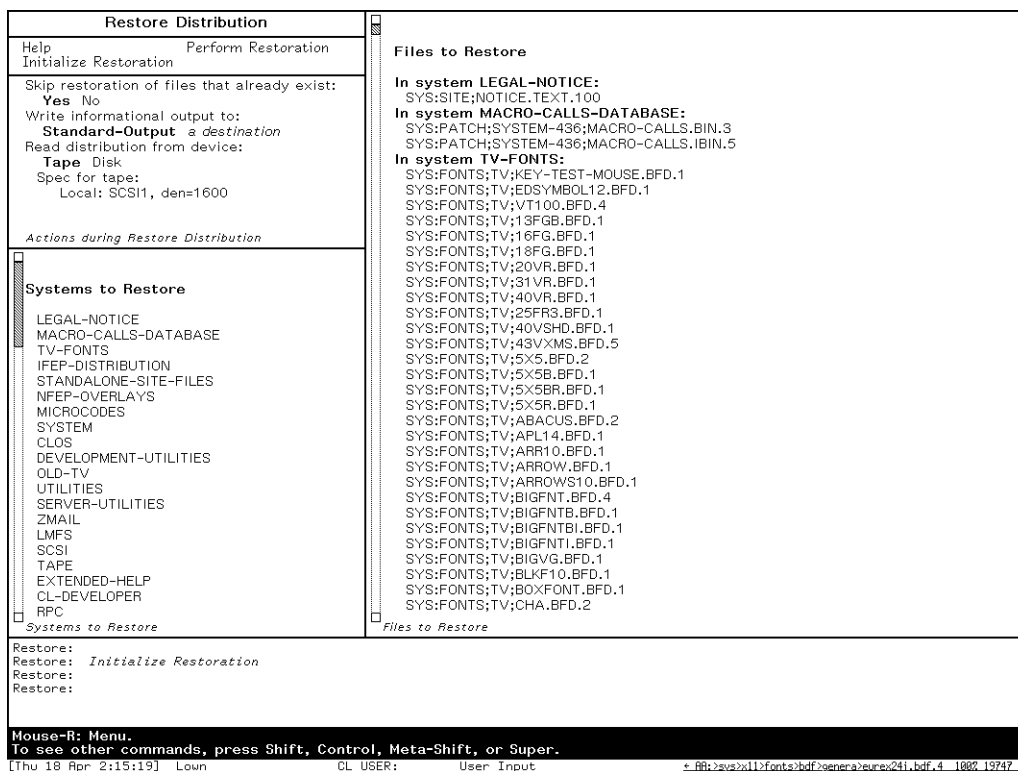


Figure 136. System Selection Menu for the Restore Distribution Command

At the menu, specify the appropriate pathname for the distribution image on the CD-ROM

```
Lisp Machine: LISP-MACHINE|CDROMn:>DISTRIBUTION>GENERA_8_2_NXP_SOURCES.TAPE
Sun: sun-host:/cdrom/distribution/genera_8_2_nxp_sources.tape
```

3. 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 distribution image.

All 8.2 NXP sites must restore the following systems:

LEGAL-NOTICE
 MACRO-CALLS-DATABASE
 TV-FONTS

All sites that include an Ivory-based machine (NXP, MacIvory, XL, or Symbolics UX) must restore the following systems:

IFEP-DISTRIBUTION (Ivory FEP Kernel and IFEP Overlays)

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

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

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

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

MAILER
DOMAIN-NAME-SERVER

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

CLIM (Common Lisp Interface Manager)
CLIM-DEMO
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
X-CLIENT-FONTS
STATIC-RUNTIME
DBFS-UTILITIES (Static backup utilities)
UNSUPPORTED (systems that are no longer maintained)

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

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

Sites that use CLIM with Color X-Terminals must restore the following system:

IVORY-COLOR-SUPPORT

Copying New Flod Files

New flod files are available for Ivory machines upgrading from Genera 8.0 software to Genera 8.1. Copying new flod files is required for Ivory-based machines (including the XL, MacIvory, and Symbolics UX) even if you have restored the new flod files and FEP kernel from the distribution world tape.

Copy the newest FEP software to the FEP file system on each of your Symbolics machines. For Ivory-based machines this includes FEP kernel and flod files.

Give the following command at a Lisp Listener:

```
Copy Flod Files :version 1325
```

Answer "Yes" to the query about updating the hello.boot file. By doing this, your hello.boot file will automatically be updated to scan the new flod files.

For information about the Copy Flod Files command, see the section "Copy Flod Files Command".

To use the new flods, you must cold boot the machine's FEP. Follow the instructions for your machine type:

Copying New Flod Files to a 3600

There was a bug in the NFEP overlays shipped with Genera 8.0.1 that caused bus errors when the Clear Machine FEP command was issued on some machines. Regrettably, the bug is extremely elusive and could not be diagnosed in time for Genera 8.1. In order to correct this, we are reverting to the NFEP overlay versions shipped with Genera 8.0, which did not exhibit this problem. To keep NFEP overlay installation simple, we have increased the version numbers on the Genera 8.0 overlays before releasing them for Genera 8.1. Other than the changed version numbers, the Genera 8.1 NFEP overlays are identical to the Genera 8.0 overlays.

Copy the newest NFEP overlays to the FEP file system on each of your Symbolics 3600 Family machines.

Give the following command at a Lisp Listener:

```
Copy Flod Files :Version (one of "V127", "G206", or "G208")
```

For information about the Copy Flod Files command, see the section "Copy Flod Files Command".

Note: If you plan to use Internet-based namespace services at your site, you must copy the rel7 flod by hand to each 3600 Family machine at your site. Use the Copy File command to copy the appropriate rel7 flod file from the SYS:N-FEP; directory to the FEP filesystem of each 3600 Family machine.

For information about Internet-based namespace services, see the section "Using the Internet as the Primary Network".

After you have restored the documentation, sources, examples, fonts, and non-loaded systems, you are ready to make customized Genera 8.1 user and server worlds. (For further information on this topic, see the section "Customizing and Saving Worlds".)

Making Customized Genera User and Server Worlds

Follow these instructions to make customized worlds containing the software you will be using at your site. For information about netbooting, See the section "Net-

booting". For more information about Loadable Systems, see the section "Loadable Systems for All Machine Types".

Making Worlds for User Machines

We recommend the use of Incremental Disk Save (IDS) for customized worlds. (This makes it easier to distribute newer worlds to machines with small disk configurations.)

1. Boot the site-configured world with IDS enabled. (On Ivory-based machines, check to see that your `hello.boot` file contains a Set Boot Options command line with the keyword and value `:IDS Enable`. On 3600-series machines, check that your `boot.boot` file contains an Enable IDS command line). If you have multiple machines at your site, *do not* use your namespace server machine to boot this world.

2. Disable Services, using the Disable Services command, as shown:

```
Disable Services
```

3. Log in as the default system user. Type to the command prompt:

```
Login Lisp-Machine
```

This command line logs you in as the default user and is useful during installation procedures, when you must perform certain operations after cold booting.

4. Load any software you want to include in the world such as non-loaded systems or layered products. For example, to load the non-loaded system called Metering, type:

```
Load System Metering
```

For more information on loadable systems, see the section "Loadable Systems for All Machine Types".

For 3600 family machines, also see the section "Loadable Systems for 3600 Family Machines".

5. Optionally, use the Optimize World command:

```
Optimize World
```

The Optimize World command reorganizes the world to improve paging performance. It increases the size of an IDS world, however. More information is available about the Optimize World command. See the section "Optimize World Command".

6. Save the new world, using either the Save World Incremental command or the Save World Complete command. This world will serve as the base world

for all of the non-server machines at your site. For example, to save an incremental Metering world using Genera 8.1 software, and based on the site-configured world:

```
Save World Incremental FEP0:>Inc-Metering-8-1-from-Inc-Site.load
```

Information is available about which of the two Save World commands to use. See the section "Save World Command".

7. Once you have created these worlds, you can move them out to the appropriate machines, using the Copy World command.

Note: The `.ilod` file extension indicates world-load files for Ivory-based machines, just as the `.load` file extension indicates world-load files for Symbolics 3600-family machines. Files with the `.ilod` extension can be copied only between Ivory-based machines. Files with the `.load` extension can be copied only between Symbolics 3600-series machines.

Loadable Systems for All Machine Types

CLIM

CLIM stands for Common Lisp Interface Manager. It is a portable, powerful, high-level user interface management system toolkit intended for Common Lisp software developers.

For more information about CLIM, see the section "Common Lisp Interface Manager (CLIM): Release 1.0".

Certain CLIM patches depend on having the CLX system loaded. Before loading CLIM, load the CLX system. CLX is a component system of X Remote Screen.

If you want online access to the CLIM tutorials, make sure that you have restored the CLIM-TUTORIAL files from the Genera source tape image.

To load CLIM, type

```
Load System CLIM
```

The CLIM Lisp Listener is bound to `SELECT-λ` (`SELECT-symbol-shift-L`).

CLIM-Demo

The CLIM Demos provide examples of small and medium-sized applications using CLIM.

For more information about the CLIM Demos, see the section "Using the CLOE CLIM Demos".

Before loading the CLIM Demo system, CLIM must be loaded.

To load the CLIM Demos, type

Load System CLIM-Demo

To run the CLIM Demos in Genera, type

```
(setf clim-demo::*demo-root* (clim:open-root-window :sheet))
(clim-demo::start-demo)
```

CLIM-Doc

CLIM Doc contains the online documentation for Symbolics CLIM.

CLIM Doc depends on packages defined in the CLIM system.

To load CLIM Doc, type

Load System CLIM-Doc

Print

The Print system defines the Genera Print Spooler. The Print Spooler manages multiple print requests sent to a printer, from any number of users. You must load the Print system on Symbolics hosts designated as print spoolers.

For more information about Printers and Print Spoolers, see the sections "Installing a Printer" and "Defining the Print Spooler" in *Site Operations*.

If you are using Symbolics LGP2 or LGP3 printers, make sure that you have restored the LGP2-SUPPORT files from the Genera source tapes. If you are using Symbolics DMP1 printers, make sure that you have restored the DMP1-SUPPORT files from the Genera source tapes. Note that the LGP2-SUPPORT and DMP1-SUPPORT systems are not loadable systems, they just contain fonts and metric files which the Print software expects to find on your disk.

To load the Print system, type

Load System Print

The Print Spooler activity is bound to SELECT-S.

Mailer

The Mailer system defines the Genera Store-and-Forward Mailer. The Mailer is a program that provides mail forwarding and delivery services to users at a site. The Mailer is distinct from Zmail, which is a user program for reading and composing mail. You must load the Mailer system on Symbolics hosts designated as mail servers.

For more information about the Mailer, see the sections "Symbolics Store-and-Forward Mailer" and "Installing and Configuring the Mailer" in Document Examiner (see the section "Reading Online Documentation").

The mailer does not depend on any other loadable systems. However, if you wish to send and receive mail from other sites on the ARPA Internet, you may need to

set up a Name server at your site. See the section "Loadable Systems on the CD-ROM - Domain Name Server".

To load the Mailer system, type

```
Load System Mailer
```

The Mailer Activity is bound to SELECT-0.

Domain Name Server

The Domain Name Server software enables a Symbolics computer to act as a Name server host. A Name server provides other hosts on the network with information about the hosts in your domain. Hosts inside and (optionally) outside your site consult your Name server about objects in your domain. You must load the Domain Name Server system on Symbolics hosts designated as Name servers. Note that it is not necessary to load the Domain Name Server system on hosts which serve only as central name resolvers for your site.

For more information about Domain Names, Name servers and central name resolvers, see the section "Internet Domain Names" and see the section "Installing the Internet Domain Names System".

The Domain Name Server system does not depend on any other loadable systems. However, a Symbolics host that is designated as a Name server for the ARPA Internet must run IP/TCP, because it must be capable of communicating with other hosts on the Internet using IP/TCP. See the "IP-TCP" description below.

To load the Domain Name Server system, type

```
Load System Domain-Name-Server
```

The Domain Name Server activity is on Select-a.

Metering

The Metering system defines a uniform interface to the various Genera metering tools. This interface makes it convenient to meter different aspects of program performance, such as time, paging, and consing.

For more information about Metering, see the section "Metering a Program's Performance".

The Metering system does not depend on any other loadable systems.

To load the Metering system, type

```
Load System Metering
```

The Metering Interface activity is bound to SELECT-2.

Conversion-Tools

The Conversion Tools system defines special-purpose Zmacs commands which partially automate the code conversion process. Tools are included for the following

types of conversions: DW to CLIM, Flavors to CLOS, Zetalisp to Common Lisp, Symbolics Common Lisp to portable Common Lisp, and Package Conversion.

For more information about Conversion Tools, see the section "Conversion Tools". For more information about converting from Dynamic Windows to CLIM, see the section "Converting an Application From DW to CLIM".

The Conversion Tools system does not depend on any other loadable systems.

To load the Conversion Tools system, type

```
Load System Conversion-Tools
```

Hacks

The Hacks system contains Genera Demos.

For more information about the Genera Demonstration facility, see the section "The Demonstrations Facility".

The Hacks system does not depend on any other loadable systems.

To load the Hacks system, type

```
Load System Hacks
```

IP-TCP

The IP/TCP software package enables a Symbolics computer to communicate with other systems supporting the ARPA Internet protocol family. Load IP/TCP on machines which will use IP/TCP protocols for remote login, file transfer, and/or electronic mail and messages.

For more information about IP-TCP, see the document *Symbolics IP/TCP Software Package*.

The IP-TCP system does not depend on any other loadable systems. Note that the Genera 8.2 NXP World, and the 8.1 Network IDS for Symbolics UX family machines already contains IP-TCP.

To load the IP-TCP system, type

```
Load System IP-TCP
```

IP-TCP-Doc

IP-TCP Doc contains the online documentation for Symbolics IP-TCP.

IP-TCP Doc depends on packages defined in the IP-TCP system. Note that the Genera 8.2 NXP World, and the 8.1 Network IDS for Symbolics UX family machines already contains IP-TCP Doc.

To load IP-TCP Doc, type

Load System IP-TCP-Doc

NFS-Server and NFS-Client

Symbolics NFS is a fully symmetrical implementation of the Sun Network File System protocol, as specified in the Sun Network File System Protocol Specification, Revision B of 17 February 1986. Symbolics NFS allows your Symbolics machine to act as an NFS client or server or both. Load the NFS Server system on Symbolics machines which will act as NFS servers (NFS-Client is a component system of NFS-Server, and will be loaded automatically when you load NFS-Server). Load the NFS Client system on Symbolics machines which will act as NFS clients.

For more information about Symbolics NFS, see the document *Symbolics Network File System (NFS) User's Guide*.

The NFS Server and NFS Client systems depend on RPC and IP-TCP.

Note that the Genera 8.2 NXP World, and the 8.1 Network IDS for Symbolics UX family machines already contains RPC, IP-TCP and NFS Client.

To load the NFS-Server system, type

```
Load System NFS-Server
```

NFS-Documentation

NFS Documentation contains the online documentation for Symbolics NFS.

NFS Documentation depends on packages defined in the NFS system. Note that the Genera 8.2 NXP World, and the 8.1 Network IDS for Symbolics UX family machines already contains NFS Documentation.

To load NFS Documentation, type

```
Load System NFS-Documentation
```

X-Server

The Symbolics X Server program handles X protocol output and input requests from other systems on the network and performs the requested operations on the local system's screen. It allows you to operate X client applications (such as "xterm") running on other systems from your Symbolics Computer's console. The Symbolics X Server only works on non-embedded machines. On embedded machines (MacIvory or UX), run an X Server on the embed host, such as MacX for Macintosh or OpenWindows for Sun.

For More information about the Symbolics X Server, see the document *Symbolics X Window System User's Guide*.

The X-Server system does not depend on any other loadable systems. The C Runtime system, which includes software to support the C language implementations

of the X Server, is a component system of X-Server. As long as you have restored C-Runtime from the Genera Source tape image, C-Runtime will automatically be loaded when you load the X-Server system. The standard set of X fonts are included in the X Server distribution, in the Server Native Format used by the X Server. To use these, make sure that you have restored the X-SERVER-FONTS files from the Genera Source tape image.

To load the X-Server system, type

```
Load System X-Server
```

The X11 Server activity is bound to SELECT-SQUARE.

X-Remote-Screen

X Remote Screen is the client side of Symbolics X Windows. In X Windows, the client is the application seeking to present a user interface, and the server is a virtual console capable of doing so. The Genera X client supports multiple user interfaces simultaneously, possibly displayed on different consoles.

For More information about Symbolics X Remote Screen, see the document *Symbolics X Window System User's Guide*.

The X-Remote-Screen system depends on RPC, Embedding-Support, and IP-TCP.

If you wish the Genera X Client to use Genera X fonts instead of the standard 75dpi X fonts, make sure that you have restored the X-CLIENT-FONTS files from the Genera Source tapes, and that you have done any necessary server-dependent work to get the Genera X Client to use the Genera fonts. Note that the Genera 8.2 NXP World, and the 8.1 Network IDS for NXP1000 and Symbolics UX family machines already contains RPC, Embedding-Support, IP-TCP and X-Remote-Screen.

To load the X-Remote-Screen system, type

```
Load System X-Remote-Screen
```

X-Documentation

X-Documentation contains the online documentation for Symbolics X Windows (both the Symbolics X Server and X Remote Screen).

X-Documentation does not depend on any other loadable systems. Note that the Genera 8.2 NXP World, and the 8.1 Network IDS for NXP1000 and Symbolics UX family machines already contains X-Documentation.

To load X Documentation, type

```
Load System X-Documentation
```

Stalice-Runtime

Stalice Runtime is a subset of Stalice, an object-oriented database system for the Genera programming environment. Stalice-Runtime enables Genera users to use,

access, and maintain a Staticce database. Staticce Runtime does not include tools for developing Staticce applications -- those tools are available separately, in the Staticce Developer layered product.

For more information about Staticce Runtime, see the section "Staticce Runtime".

The Staticce-Runtime system does not depend on any other loadable systems.

To load the Staticce Runtime system, type

```
Load System Staticce-Runtime
```

DBFS-Utilities

The DBFS-Utilities system defines utilities for making backup copies of databases stored in a Staticce File System. If a disk containing a Staticce File System is damaged or destroyed, a backup copy of the database can be restored onto a fresh disk.

For more information about Staticce backup utilities, see the section "Staticce File System Operations Program".

DBFS-Utilities depends on Staticce-Runtime. Before loading DBFS-Utilities, make sure that the Staticce-Runtime system is loaded on your machine.

To load the DBFS-Utilities system, type

```
Load System DBFS-Utilities
```

The Staticce File System Operations program is bound to `SELECT-δ` (`SELECT-symbol-shift-D`).

Mac-Dex

Mac-Dex is the Macintosh Document Examiner, a sample MacIvory application. Mac-Dex is provided as an example of how to write an application which presents a Macintosh style interface using Symbolics' standard Macintosh application and **zl-user:define-remote-program-framework**. The Mac-Dex system is the Lisp side of the Mac Dex application, which is our standard Macintosh application configured to run Mac-Dex. Loading Mac-Dex on a 3600-family machine (or other non-MacIvory Symbolics machines) would allow you to use Mac-Dex from a non-NuBus Macintosh by talking to your 3600.

For more information about using Symbolics' standard Macintosh application, see the section "Setting the Size of Application Icons That Use Ivory".

The Mac-Dex system depends on RPC, Embedding-Support, and MacIvory-Support. On 3600-family machines, load the RPC, Embedding-Support, and MacIvory-Support systems before loading Mac-Dex. Note that all Ivory distribution worlds already contain RPC, Embedding-Support, and MacIvory-Support.

To load the Mac-Dex system, type

```
Load System Mac-Dex
```

To run Mac Dex, do the following:

1. Load the system Mac-Dex into Genera.
2. Use [File / Quit] to get out of Genera.
3. Double-click on the Mac Dex icon, located in the MacIvory Applications folder.

You do not need to run the Genera icon when you run the Mac Dex application. The Mac Dex application will boot your Ivory if it is not booted. To access Genera documentation via Mac Dex, use the menu titled **Lookup** in the Apple menu bar.

HyperCard-MacIvory

The HyperCard-MacIvory system contains the MacIvory interface to HyperCard. It also includes a collection of demos that illustrates the close communication and integration possible between Ivory and a host HyperCard application.

For more information about the HyperCard-MacIvory system, see the section "MacIvory Interface to HyperCard".

The HyperCard-MacIvory system depends on Mac-Dex. Note that the HyperCard interface always uses an embedded channel, so loading HyperCard-MacIvory is only useful on a MacIvory.

To load the HyperCard-MacIvory system, type

```
Load System HyperCard-MacIvory
```

IVORY-COLOR-SUPPORT

You must load Ivory-Color-Support if you are using CLIM with a Color X-terminal with an NXP1000. On a Color MacIvory, loading Ivory-Color-Support allows you to configure a MacIvory application to use full Macintosh Native Color.

Ivory-Color-Support does not depend on any other loadable systems.

To load Ivory-Color-Support, type

```
Load System Ivory-Color-Support
```

Loadable Systems for 3600 Family Machines

RPC

Remote Procedure Call (RPC) is a facility that allows a function executing on one processor to call a function executing on another processor. The two functions can be written in the same language or in different languages. The two processors can be of the same type or of different types. Symbolics RPC is an implementation of industry-standard RPC that underlies Sun Microsystems' NFS and other programs. Loading RPC on a 3600-family machine allows you to run applications which use

RPC, such as an application which uses RPC to request service (or provide service) to a Sun.

For more information about Symbolics RPC, see the section "The Remote Procedure Call Facility".

RPC does not depend on any other loadable systems. However, if you wish to have online access to RPC examples, make sure that you have restored the ADDITIONAL-RPC-EXAMPLES and UX-RPC-AND-FRIENDS files from the Genera Source tapes. Note that all Ivory distribution worlds already contain RPC, so loading RPC is only relevant on 3600 family machines.

To load the RPC system, type

```
Load System RPC
```

Embedding-Support

The Embedding Support system contains support for various features of embedded hosts. For example, Apple and Sun keyboard support are defined in the Embedding Support system. Load Embedding Support on a 3600 family machine if you wish to load other systems which depend on Embedding Support (such as X Remote Screen, Mac Dex, MacIvory Support, MacIvory Development, UX Support, or UX Development).

For more information about embedded hosts, see the documents "MacIvory User's Guide" and "User's Guide to the Symbolics UX".

The Embedding-Support system depends on RPC. On 3600-family machines, load the RPC system before loading Embedding-Support. Note that all Ivory distribution worlds already contain Embedding-Support, so loading Embedding-Support is only relevant on 3600 family machines.

To load the Embedding-Support system, type Load System Embedding-Support

MacIvory-Support

The MacIvory Support system defines the Macintosh user interface for MacIvory, and also provides the software necessary to access a remote MacIvory's Macintosh file system. The main reason for loading MacIvory Support on a 3600-family machine is to enable file access to a MacIvory's Macintosh file system.

For more information about accessing a Macintosh file system, see the section "Accessing the Macintosh File System".

The MacIvory Support system depends on RPC and Embedding-Support. On 3600-family machines, load the RPC and Embedding-Support systems before loading MacIvory-Support. Note that all Ivory distribution worlds already contain MacIvory-Support, so loading MacIvory-Support is only relevant on 3600 family machines.

To load the MacIvory-Support system, type

```
Load System MacIvory-Support
```

UX-Support

The UX-Support system enables various interactions between Genera and UNIX. Loading UX-Support on a 3600 family machine allows you to access a Sun tape drive from your 36XX, using the UNIX-REXEC rmt tape server. Loading UX-Support on a 3600 family machine also enables you to use a UNIX printer via UNIX-LPD.

For more information about the Symbolics UX machine series, see the document *User's Guide to the Symbolics UX*.

The UX Support system depends on RPC and Embedding-Support. On 3600-family machines, load the RPC and Embedding-Support systems before loading UX-Support. Note that all Ivory distribution worlds already contain UX-Support, so loading UX-Support is only relevant on 3600 family machines.

To load the UX-Support system, type

```
Load System UX-Support
```

RPC-Development

The RPC Development system defines utilities which enable you to write your own RPC-based programs. Loading RPC-Development on a 3600-family machine allows you to do RPC development on a 3600.

For more information about Symbolics RPC, see the section "The Remote Procedure Call Facility".

The RPC-Development system depends on RPC. On 3600-family machines, load the RPC system before loading RPC-Development. Additionally, if you wish to have on-line access to RPC examples, make sure that you have restored the ADDITIONAL-RPC-EXAMPLES and UX-RPC-AND-FRIENDS files from the Genera Source tapes. Note that all Ivory distribution worlds already contain RPC-Development, so loading RPC-Development is only relevant on 3600 family machines.

To load the RPC-Development system, type

```
Load System RPC-Development
```

MacIvory-Development

The MacIvory Development system contains tools for developing MacIvory user interfaces. Loading MacIvory-Development on a 3600-family machine enables you to do Macintosh interface development on a 36XX.

For more information about developing MacIvory user interfaces, see the section "Developing User Interfaces with MacIvory".

The MacIvory-Development system depends on RPC, RPC-Development, Embedding-Support, and MacIvory-Support. On 3600-family machines, load the RPC, RPC-Development, Embedding-Support, and MacIvory-Support systems before loading MacIvory-Development. Note that all Ivory distribution worlds already contain

MacIvory-Development, so loading MacIvory-Development is only relevant on 3600 family machines.

To load the MacIvory-Development system, type

```
Load System MacIvory-Development
```

UX-Development

UX-Development currently contains no files. It is reserved for future development work. There is currently no reason to load UX-Development on any 3600 family machine.

For more information about the Symbolics UX machine series, see the document *User's Guide to the Symbolics UX*.

Making Worlds for Server Machines

Server Machines:

If you have server machines that run additional software (such as Print, Domain Name Server, or Mailer), follow the procedure used in "Making Worlds for User Machines" and build an IDS world, with the appropriate server systems loaded, for use on these machines.

Make non-namespace server worlds on non-namespace server machines. This will enable non-namespace servers to boot more quickly, because their worlds will be smaller.

Namespace Server Machines:

Since namespace server machines load all the current namespace information into virtual memory, worlds saved on them are larger and boot more slowly than those saved on non-namespace server machines. Follow the procedure used in "Making Worlds for User Machines" and build an IDS world, with the appropriate server systems loaded, on a namespace server machine.

Backing Up Worlds to Tape

Back up the world(s) you just created (and the appropriate microcode) to tape. For complete information, see the section "FEP-Tape System" in *Site Operations*. Here is the procedure in brief:

1. Type the command in the Lisp Listener:


```
Select Activity FEP-Tape
```
2. Use the command Add File to add a single file to the list of files to be written to tape. For example, type to the FEP-Tape command prompt:


```
Add File the pathname of the base site world
Add File other world loads and microcodes you want to back up
```

Or, you can click on [Add File]. If you back up both the world load and the microcode files, these files can both be restored if needed.

3. Use the command Write Tape to write the file to tape; type this to the FEP-Tape command prompt:

```
Write Tape
```

We recommend that you type the Write Tape command, rather than clicking on the [Write Tape] menu item, because typing the command gives you more control over the tape specification.

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.

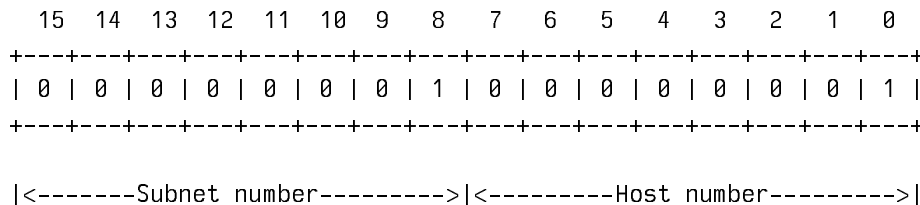
If you use networks such as NFS, DNA or IP-TCP along with Chaosnet, we suggest that when you configure your site, you employ an addressing scheme that will encompass all of these. More information is available about network addressing schemes, see the section "Network Addressing".

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



- 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;
zmail; 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; misc;

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; rpc; 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 137. 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 137 .

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 129 .
3. The network must be a primary network for the local machine. The address must be a valid address for the network. Figure 130 shows the complete dialog for setting the site when the Primary 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: a pair of a network and an address
<ABORT> aborts, <END> uses these values

```

Figure 138. 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 139. Complete menu 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 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 131 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 140. 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 134:

```

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 141. 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 142. 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
- a MacIvory running Genera 8.1.1
- a Sun-4

Note: Genera 8.1 is not sufficient, Genera 8.1.1 contains bug fixes for several significant CD-ROM problems.

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;*.;* *
```

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 8.1 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 *CDROMn* is the SCSI address of the CD-ROM drive.

If all machines at your site are running Genera 8.2 NXP, and if you wish to use a CD-ROM as the SYS host for all logical pathnames, 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 the host to which the CD-ROM is connected is 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/**/*")))
```

If some machines at your site are running a release older than Genera 8.2, 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" "1.1")
       (fs:set-logical-pathname-host "SYS" :translations
        '(("sys:**;*.*.*" "ACME-SMBX:>rel-8-1>sys>**>*.*.*)"))))
      ("2 NXP" "2 NXP ECO#1")
       (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 8.2 NXP 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 8.2):

```

;;; -*- 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-8-2>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 8.2 NXP distribution which contain such characters. If you run into one, simply take the retry option 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 8.1 distribu-

tion, however only three files produced ambiguous pathnames when truncated. On the Genera 8.2 NXP CD, 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_LANUAG_2.LISP
SYS:EXAMPLES;CONSTRAINT_FRAME_LANUAG_3.LISP
```

on the Genera 8.1 CD.

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. Your Symbolics machine must have the NFS-Client system loaded to access files on your UNIX file server. If you have any 3600-series machines, MacIvories, or XL machines, you should keep your SYS:SITE:, SYS:IP-TCP:, and SYS:NFS: directories on a Symbolics computer, as access to these directories is required to load the NFS Client system.

Storing Sources for NXP1000s (or UX-family machines)

NXP1000s (and UX-family machines) already have the network software bundled into their worlds, so the NFS Client system is already loaded.

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/rel-8-2/sys.sct/**/")))
```

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 .n.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.

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*.

Restoring Worlds From CD-ROM Using the FEP-Tape Activity

Once you are running Genera 8.1.1, you can restore worlds from the CD-ROM using the FEP-Tape activity. To do this, type Read Image File and supply a CD-ROM pathname to a world image. Do not use the [Read Image File] menu item, because it will not prompt you for a world image pathname. The FEP-Tape images on the Genera 8.1 CD are in the >DISTRIBUTION> directory (/distribution/, if the CD-ROM drive is connected to a Sun. The possible FEP-Tape images on the Genera 8.1 CD-ROM are:

- basic_world.tape (contains IFEP 325 and Genera-8-1.ilod)
- other_ivory_worlds.tape (contains remaining Genera 8.1 Ivory worlds)
- 3600_worlds.tape (contains 3600 microcodes and Genera 8.1 3600 worlds)