Genera 7.2
Installation & Notes

symbolics™

Product Number 999078

Including:

Genera 7.2 Software Installation Guide
Genera 7.2 Patch Notes

Cambridge, Massachusetts

Genera 7.2
Software Installation Guide

symbolics™

Cambridge, Massachusetts

Genera 7.2 Software Installation Guide
999076

February 1988

This document corresponds to Genera 7.2 and later releases.

Table of Contents

List of Figures

1. 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", page 22.

1.1. 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 server's database. More information is available about namespace objects and the namespace system.

See the document *Networks*.

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", page 4.

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:

- A Symbolics computer *or*

- A VAX running Berkeley UNIX 4.2 or earlier (this machine must be available via Chaosnet or IP-TCP) *or*

- A VAX running Berkeley UNIX 4.3 or later (this machine must be avaiable via IP-TCP) *or*

- A VAX running VMS (this machine must be available via DECnet).

More information is available about system sources and online documentation. See the section "System Sources and Online Documentation", page 6.

More information is available about managing the namespace database. See the document *Networks*.

1.2. 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're ready to configure your site, choose a site name that — in all probability — won't 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.

1.3. Choosing Machine Names and Chaosnet Addresses

It is easier to refer to a computer by its name than by its Chaosnet (the Symbolics network) 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.

Chaosnet addresses are represented on the Symbolics computer as a 16-bit number whose most significant eight bits are a subnet number and whose least significant eight bits are a machine number.

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.

If you use networks such as 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 document *Networks*.

Note: All 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.

1.4. 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), while others store them on a VAX/VMS, a VAX/Berkeley UNIX host running Chaosnet or IP-TCP, or a VAX/VMS host running DNA. More information is available about these VAX hosts. See the section "Site Configuration and Namespace Service", page 1.

In any case, 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.

A site that stores the system software on a VAX/VMS system translates logical pathnames into VAX/VMS 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.

More information is available about logical pathnames and VAX/VMS hosts. See the section "VAX/VMS Version 4.4 Pathnames and Genera 7.2", page 52.

1.5. 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 document *Site Operations.*

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.

1.5.1. 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*. For example, the name of the current distribution world is Genera 7.2.

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", page 4.

1.5.2. 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 "Dictionary of Command Processor Commands" in *Genera Handbook*.

1.5.3. System Sources and Online Documentation

Symbolics distributes a portion of the sources from which the Genera 7.2 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 7.2 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", page 3. Also See the section "Pruning the Documentation Database", page 6.

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. Once Document Examiner's window appears, you can view all of Symbolics' documentation by using the mouse to click Middle on the menu option [Help].

1.5.3.1. 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.

The function **sage:prune-online-document-set** lets you remove specific books from the online set. This function displays the names of all of the books in the document set. You mark the books that you want to remove by clicking on their names. (See online help for further details.)

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 (like 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.

1.5.4. File Systems on a Symbolics Computer Disk

The disk of a Symbolics computer is divided into FEP files. This collection of files constitutes the FEP file system, called FEPFS.

The FEPFS is a very simple file system that can be manipulated by the Front-End Processor (the FEP). The FEPFS typically contains a small number of special-purpose files necessary for the operation of the Symbolics computer. One of the special types of files commonly found within the FEPFS is a Lisp Machine File System partition, called LMFS, where the system source files are stored.

There can be several disk units connected to a given Symbolics computer. Each one has a FEP file system. To distinguish between disk units, the syntax *FEPn* is used, where *n* is the disk unit number in the FEPFS filename. The first disk unit number is FEP0, and subsequent disks are numbered accordingly. For example, FEP3:>boot.boot means the FEPFS file called boot.boot residing on disk unit 3 of this Symbolics computer.

The LMFS partitions are typically large FEPFS files that provide storage space for use by the Lisp Machine File System.

The LMFS is the file system used for day-to-day storage and access of files. Users' files, system sources, and all files that are not specifically required to be FEP files are all stored in the LMFS.

Note that the LMFS is a file system within a file system, and that when one speaks of the file system of a Symbolics computer, one is usually referring to the LMFS (not the FEPFS) of that machine. An example of a LMFS partition:

```
FEP0:>LMFS.FILE.1
```

A Symbolics computer can have several FEP file systems, but only one Lisp Machine File System (LMFS). The LMFS can contain more than one LMFS partition. Remember, each LMFS partition is a file in one of the machine's FEP file systems.

The total storage capacity of the LMFS is the sum of the sizes of the LMFS partitions. No matter how many partitions make up the LMFS, together they make up only one Lisp Machine File System.

The following situation, which might initially seem confusing, is not uncommon: A Symbolics computer can have three disk units, each containing a FEP file being used as a partition of the machine's LMFS. Each of these files resides in a different FEPFS, but the three files together make up one LMFS.

More information is available about the FEP File System. See the section "Examining the FEP File System", page 8.

More information is available about the LMFS File System. See the section "Examining the Lisp Machine File System", page 10.

1.5.4.1. Examining the FEP File System

The Command Processor command Show FEP Directory prints information about the FEP directory on your machine. All numbers are expressed in decimal. Lines that are in bold are the files in use. More information is available about the Show FEP Directory command. See the document *Site Operations*.

An example follows:

```
Host REDWING:
Unit 0:  23456 free, 238688/262144 used (91%)
Lines highlighted in bold represent files currently in use.
```

World Load Files:
```
FEP0:>Netboot-Core-from-Genera-7-2.load.1          157
```
FEP0:>Genera-7-2.load.1   **32755 [Genera 7.2]**
FEP0:>site-print-spooler-Genera-7-2.load.1 8398
[print spooler, Genera 7.2]

Microcode files:
FEP0:>3675-mic.mic.419   **126**
```
FEP0:>3675-mic.mic.396                        123 [3675-MIC 396]
```

Paging files:
FEP0:>page.page.1 60000
FEP0:>page1.page.1 40000

Boot files:
```
FEP0:>Genera-7-2.boot.1     1
FEP0:>7-2-print.boot.1      1
FEP0:>boot.boot.1           1
FEP0:>Hello.Boot.1          1
FEP0:>netboot.boot.1        1
```

Lisp Machine File System Partitions:
```
FEP0:>fspt.fspt.1           1
FEP0:>lmfs.file.1       20000
```

Fep-specific files:
```
FEP0:>BAD-BLOCKS.FEP.1              40 [File of bad blocks]
FEP0:>DISK-LABEL.FEP.1             24 [Disk label]
FEP0:>FREE-PAGES.FEP.1             29 [Free pages map]
FEP0:>Reserve.FEP.1              1000 [Reserved for future use]
FEP0:>SEQUENCE-NUMBER.FEP.1         1 [FEP FS sequence nos]
FEP0:>UNIQUE-ID.FEP.1               2
```
FEP0:>v127-debug.flod.37   51
```
FEP0:>v127-disk.flod.36            38
```
FEP0:>V127-info.flod.62   17
FEP0:>V127-lisp.flod.62   52
FEP0:>V127-loaders.flod.62   52
```
FEP0:>V127-tests.flod.62           10
```

The first line of the display gives useful status information. In this case, you can see that there are 23456 blocks available and 238688/262144 blocks used in Redwing's FEPFS disk number 0. Each line of the display is of the form:

```
Filename Size-In-Blocks Comment
```

One FEPFS block contains approximately 1100 8-bit bytes of data. In general, you can assume that approximately 10 to 15 percent of the storage space of any given disk is consumed as overhead due to formatting, or is unavailable due to various media defects.

Thus, if you have a 335 megabyte drive, only about 300 megabytes are available after formatting. In the example above, the disk contains 262144 FEPFS blocks of storage, which is approximately 300 megabytes.

In the above directory listing we can see that LMFS.FILE.1 takes up 20000 blocks. Therefore, in this example, 20000 blocks of disk space are consumed by the Lisp Machine File System (and are unavailable for other purposes) regardless of how much of the space is actually allocated to files within the LMFS.

In other words, 22 megabytes (20000 * 1100) of the FEPFS have been allocated as "Lisp Machine File System storage", and are not available for any other use.

Also keep in mind that once storage has been allocated as LMFS storage, it is effectively committed to being LMFS storage for all time. It is an easy procedure, however, to allocate additional disk space as LMFS storage if necessary.

Warning: Never delete your LMFS partitions under any conditions.

In addition, a small amount of space in every FEPFS is taken up by a collection of files that should not be deleted or modified. Examples of these are files with types FSPT and DIR, as well as FEP.

A chart showing Symbolics computer disks and their storage capacities is available. See the document *Site Operations*.

1.5.4.2. Examining the Lisp Machine File System

The Command Processor command Show Directory *directory-pathname* prints out useful information about the LMFS directory. Look at the LMFS of the machine named Picasso, with the following command:

```
Show Directory (files) Picasso:>*.*.*
```

```
PICASSO:>*.*.*
 1650 free, 3340/4990 used (67%)

    art-suppliers.directory.1   1   DIRECTORY !  01/23/88 10:09:07 X=01/23/88 lispm
    patrons.directory.1   1   DIRECTORY !  02/04/88 09:32:58 X=02/04/88 lispm
    brush-types.directory.1   1   DIRECTORY !  01/14/88 12:58:56 X=01/14/88 lispm
    artists.directory.1   1   DIRECTORY !  03/04/87 10:23:46 X=07/10/87 lispm
    galleries.directory.1   1   DIRECTORY !  02/08/88 16:39:20 X=02/08/88 lispm


    5 blocks in 5 files.
```

The first line of information shows the status of the LMFS. This LMFS has 4990 records of space, of which 1650 are currently unused. Each of the remaining lines is of the form:

```
Filename Size-In-Records Other-Useful-Information Status-Bits
Creation-Date Modification-Date Author
```

One LMFS record is equal to four FEP blocks, or approximately 4500 bytes of data. Thus, in the above case, this LMFS has room for 4990 * 4500, or approximately 22 megabytes of file storage.

Currently, 3340 * 4500, or approximately 15 megabytes of the LMFS are in use. (Note that this number, 22 megabytes, is the same as what we computed as the size of the LMFS when analyzing the FEPFS.)

1.6. The Booting Process

Cold-booting a Symbolics computer means loading a saved Lisp environment, called a world load, into memory. A world load is a snapshot of a Lisp environment. A Symbolics computer may have several different world load snapshots on its disk, each one tailored for a specific application or need.

For example, you might have two world loads on your machine: one consisting of the standard Symbolics distribution world with minor local configuration-related changes, and one consisting of the first world load enhanced with software that you have developed.

When you boot a Symbolics computer, you communicate with the Front-End Processor (FEP). The FEP is a small, auxiliary processor included in the machine. It loads those files that enable the Symbolics computer to boot, and takes care of certain machine needs, such as listening for the keyboard and mouse.

The FEP's command format is similar to that of the Lisp Command Processor. (As is the case nearly everywhere in the Symbolics computer environment, the FEP gives useful information if you press the HELP key.)

Every Symbolics machine is shipped with a hello.boot file on its disk. This file bootstraps the FEP. The commands in hello.boot initialize the FEP's command tables. Whenever you power up, manually reset your machine, or issue the FEP command Reset FEP, you must type the FEP command Hello before you can boot.

The Boot command tells the FEP to read in a boot file and execute the lines in it sequentially as FEP commands. The name of the default boot file is displayed for you if you type a space after Boot or B. The default boot file when the Symbolics computer is powered up is FEP0:>boot.boot. After that, the last boot file used becomes the default.

The boot command allows you to specify a file from which to boot. For example, you can create a different boot file for each world load on your machine. All machines shipped by Symbolics have a boot.boot file on the default disk unit (FEP0). The boot.boot file contains the commands needed to boot the distribution world.

Here is an example of the minimum command sequence needed in a boot file:

```
Clear Machine
Load Microcode Microcode-file-name
Load World World-load-file-name
Add Paging-File FEP0:>more-page.page
Set Chaos-Address Chaosnet-Address
Start
```

You can either type all of these FEP commands manually, or place them in a boot file, from which they'll be executed in sequence.

Here's a detailed explanation of what happens when you tell the FEP to boot using a boot file with the minimum commands:

- *Clear Machine* clears the internal state of the registers and memories.

- *Load Microcode* loads the microcode memory and other high-speed memories from the specified file.

- *Load World* copies a portion of a world load into main memory. This command also allocates a paging file (FEP:>page.page by default). This paging file is used for virtual memory.

- *Add Paging-File* specifies more paging space (for faster access to virtual memory). You should not boot Genera 7.2 with less than 60,000 blocks of paging space.

- *Set Chaos-Address* sets the correct Chaosnet address for your Symbolics computer.

- *Start* sets the Lisp program counter at a fixed address, where the low-level function **sys:system-startup** resides, and says "Go".

After the FEP command Start is executed, Lisp takes over; when the boot process completes, you are ready to log into the machine.

After Lisp has been started, the rest of the world load is integrated into the paging areas as required. Newly created or modified pages are placed in the paging area of the disk (one or more files in the FEP file system).

When you start the Symbolics computer, a number of operations occur. These include:

- Creating/reshaping the initial Lisp Listener window, if necessary.

- Polling the network to determine the current date and time.

- Polling the network to determine the identity of the local machine.

For a list of booting commands: See the section "FEP Command Summary", page 13.

1.7. FEP Command Summary

This section contains a list of FEP commands commonly used for booting.

The FEP command processor requires you to type only as much of the command name as it takes to make it unique. The FEP redisplays the line so it shows the full spelling of the command name. Thus, you may type "l m" instead of "load microcode". The HELP key is active in the FEP command processor, telling you what commands are possible, given what you've already typed in.

For further documentation: See the section "The Front-End Processor" in *Site Operations*.

Add Paging-File   Adds a file to be used for virtual memory swap space. Use this command to specify paging files in addition to those declared with the Declare Paging-Files command. See *Declare Paging-Files*, below.

Boot   Reads a file on the FEP directory that contains FEP commands. If you type *Boot*, it will read the commands in the file Boot.boot. If you type Boot test-file.boot, it will read the commands in that file.

Clear Machine   Clears memory. Also clears various other parts of the machine hardware, such as the GC map. Use the *Clear Machine* command before loading new microcode, or using the Disk Restore or Disk Format commands.

Declare Paging-Files

Declares specified files to be the paging files for all subsequent Load World commands (until a new *Declare Paging-Files* command overrides it). A list of specified files must be separated by spaces, not commas.

Hello   Scans the NFEP overlay files (flod files). A hello.boot file comes on each Symbolics machine; this file initializes the FEP's (hardware) command tables. Each time the machine is powered up or reset, use the FEP command *Hello* to load the hello.boot file.

Load Microcode   Loads microcode along with other machine memories. You must *Load Microcode* before loading a world, and also if you want to use Disk Format. If you clear the machine using Clear Machine, you must Load Microcode again.

Load World   Loads Genera into the machine, and loads any paging files declared with the Declare Paging-Files command. To cold boot the machine, you use Load World followed by Start.

Set Chaos-Address   Sets the Chaosnet address for this machine. Each Symbolics computer must have a Chaosnet address. For information about the appropriate values for a Chaosnet address: See the section "Choosing Machine Names and Chaosnet Addresses", page 2.

Set Ethernet-Address

> Sets the Ethernet address for this machine. This command is needed only if your machine is running DNA.

Start

> Starts Genera. *Start* determines screen size, clears the screen, and initializes the rest of the environment. Then the run bars (small bars blinking on the bottom of the screen) appear, along with screen borders. When initialization completes, the Command prompt appears. A machine that is not connected to a Chaos network will request the time.

1.8. Setting and Defining Sites

When you boot a new machine, the Show Herald command displays the machine's name as *DIS-LOCAL-HOST*. In order to begin operating your computer, you must configure the new software by using the Set Site or the Define Site command.

If you are installing software where the namespace files have already been created, use the Set Site comand. This gives your machine access to the already-configured site's resources.

If you are installing software where the namespace files do not yet exist, use the Define Site command. This creates a new site for your machine. Later, you can expand the site to include other users, hosts, printers, and networks. For more information: See the section "Customizing and Saving Worlds" in *Site Operations*.

1.8.1. Set Site Command

Set Site *site-name*

Starts a dialogue to set the current site to be *site-name*. It should be the first thing you type to your machine after booting a new distribution world. The command makes the identity of all objects included in the site's namespace known to your machine.

In order for the Set Site command to work:

1. Your machine must be declared as a host in the site's namespace. See the section "Registering Hosts" in *Site Operations*.

2. The namespace database for each site is stored on a machine called the namespace server. If the site's namespace server is not the local host, you must know the name and chaosnet address of the namespace server. For more information: See the section "Choosing Chaosnet Addresses".

1.8.2. Set Site Dialogue

Here is an explanation of the Set Site dialogue. Following each line of the Set Site dialogue is the explanatory text.

First you issue the Set Site command. If you accept the default ("Get from network"), your machine automatically queries other network machines to learn their site name, and uses that site name.

```
Command: Set Site (site name [default Get from network])RETURN
```

If you wish to change your site, or if your site information is unavailable, enter a site name manually. In these examples, *Downunder* is the site name.

1. When the local host is the namespace server for your site:

    ```
    Command: Set Site (site name [default Get from network]) Downunder
    (server host name [default Local])RETURN
    ```

    Then the system prompts:

    ```
    What host is a namespace server for DOWNUNDER
    (default: Local):
    ```

    This question prompts for the name of the namespace server for your site. Press RETURN to accept the default. **Do not type the name of the local host.**

    The machine then asks for the location of the *descriptor file*. The *descriptor file* holds the names of all the namespace data files. The default value is the same one offered when defining a site:

    ```
    Where is the descriptor file for DOWNUNDER
    (default local:>sys>site>downunder-namespace.text):RETURN
    Assuming that KOALA is the real pathname host name of the local host.
    [17:53:49 Namespace on KOALA: Reloading namespace DOWNUNDER.
             Recent servers contacted are KOALA]
    [17:53:51 Namespace DOWNUNDER has become unloaded:
             No longer server for this namespace.]
    The local host is now KOALA.
    [17:54:10 Namespace on KOALA: Reloading namespace DOWNUNDER.
             Recent servers contacted are KOALA]
    ```

    The machine is now on the network at the site *Downunder*.

2. When the local host is not the namespace server for your site, provide the name of the namespace server for the site and the namespace server's Chaos address. In this example, *Kangaroo* is the name of the namespace server.

    ```
    Command: Set Site (site name [default Get from network]) Downunder
    (Server host name [default Local]) Kangaroo (Server chaos address
    (octal)) 16022
    The local host is now KOALA.
    ```

    The site's namespace determines all other relevant information, and the

machine is now on the network at the site *Downunder.*

Your machine now has access to all objects located at the site, and you are ready to log in and use the new world load. You may want to save the new, site-specific version of your world. For more information: See the section "Customizing and Saving Worlds" in *Site Operations.*

1.8.3. Define Site Command

Define Site *site-name*

Starts a dialogue to create a new site called *site-name.*

The default values for the site's namespace server, SYS host, computer storing the namespace data files, and host for bug reports, are all the local host. If you provide a non-local host for any of these, you must know the non-local Chaosnet address(es) and operating system(s). For more information:See the section "Choosing Chaosnet Addresses". Here are the default values for the Define Site command:

```
System Type*:  LISPM
Service:  CHAOS-STATUS CHAOS-SIMPLE CHAOS-STATUS
Service:  SHOW-USERS CHAOS NAME
Service:  TIME CHAOS-SIMPLE TIME-SIMPLE
Service:  UPTIME CHAOS-SIMPLE UPTIME-SIMPLE
Service:  LOGIN CHAOS TELNET
Service:  LOGIN CHAOS SUPDUP
Service:  LOGIN CHAOS 3600-LOGIN
Service:  SEND CHAOS CONVERSE
Service:  SEND CHAOS SEND
Service:  NAMESPACE CHAOS NAMESPACE
Service:  NAMESPACE-TIMESTAMP CHAOS-SIMPLE NAMESPACE-TIMESTAMP
Service:  LISPM-FINGER CHAOS-SIMPLE LISPM-FINGER
Service:  FILE CHAOS NFILE
Service: FILE CHAOS QFILE
Service: CONFIGURATION CHAOS CONFIGURATION
Address: CHAOS nnnnn
```

In the Address attribute line, *nnnnn* represents a valid 5-digit octal Chaos address.

1.8.4. Define Site Dialogue

Here is an explanation of the Define Site dialogue. Following each line of the Define Site dialogue is the explanatory text.

```
Command: Define Site (site name) downunder
```

First you issue the command and give the site name. This might be the name of your company, or, if you are more whimsical, it might be related to the machine names you have chosen. In this example, *Downunder* is the name of a sample site.

```
Define a new site named DOWNUNDER (as opposed to looking for an
existing definition of DOWNUNDER on disk)? (Yes or No) Y
```

This verifies that you want to create a new site rather than setting your site to an existing one.

```
What host is to be a namespace server for DOWNUNDER
(default LOCAL): RETURN
```

This question prompts for the name of the machine where the namespace database is stored. If the namespace server is not the local host, then you should provide the name of the namespace server for the site. **If the namespace server is the local host, do not type the host's name. Instead, use the default by pressing** RETURN.

```
What host is to be the SYS host for DOWNUNDER
(default LOCAL): RETURN
```

This question prompts for the name of the machine where Symbolics-supplied source and documentation files are to be stored by default. If the SYS host is not the local host, you can provide the name of the SYS host for the site. For more information: See the section "Logical Pathnames and the SYS Host", page 3.

```
What Symbolics computer will store the namespace data
files for DOWNUNDER (default LOCAL): RETURN
```

This question prompts for the name of the machine that stores the namespace data files. Although this machine must be a Symbolics computer, it does not have to be the same machine that is the namespace server, but we strongly suggest that it be the same machine.

```
What host is to be used for bug reports for DOWNUNDER
(default LOCAL): RETURN
```

This question prompts for the name of the machine that receives bug reports. This will most likely be the machine that you use for your mailer. For information about the mailer: See the section "Installing and Configuring the Mailer" in *Site Operations.*

```
What is the real name of the local host: Koala
```

This question prompts for your machine's name at the site. Machine names can be different at different sites. In this example, *Koala* is the name of the local host. For more information: See the section "Why do you name machines and printers?" in *Genera User's Guide.*

The default values for the namespace server, the SYS host, the computer storing the namespace data files, and the host for bug reports, are all the local host. If you provide a non-local host for any of these, the computer prompts for the non-local chaosnet address(es) and operating system(s)

```
What directory on KOALA will hold the namespace data files
(default >sys>site>): KOALA:>sys>site>
```

This question prompts for the location of your namespace data files. These files are accessed each time the site is set. We strongly suggest that you use the default value to avoid possible confusion.

```
What directory on KOALA corresponds to SYS: SITE;
(default >sys>site>): KOALA:>sys>site>
```

This question prompts for the directory that holds your translations files. These are necessary for logical pathname translations. We strongly suggest that you use the default value to avoid possible confusion.

```
What account should be used for the system to login to KOALA
(default: LISPM): Wombat
```

This question prompts for a name the system uses automatically when it needs to access files. In this example we are using the name *Wombat*. Do not supply your own name or that of another real user. When the system needs to log in automatically (for example, to read the namespace data files for the first time) the namespace server logs in with this user name.

```
What is the local timezone [default EST]: EST
```

This question prompts for the time zone. The default is Eastern Standard Time (EST) or Eastern Daylight Time (EDT) depending on whether Daylight Savings Time is in effect. For more time zone information:See the section "Specifying a Time Zone for Your Site", page 20.

```
Is DOWNUNDER a standalone site (there are no servers to respond
to a Who-am-I request)? (Y or N) Yes.
```

This question prompts for whether your site is a standalone site consisting of one Symbolics machine, or if it has many Symbolics machines. Several notification messages follow and confirm the definition of your site.

You are now ready to log in and use the newly configured software. The site of your local host is automatically set 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" in *Site Operations.*

1.9. Creating a World-Build Script File

Here is a sample world-build script file. It contains some forms that you might put into a world-build script file for your site.

Note: This file is one that contains forms enclosed in a wrapper function; it isn't a script file in the traditional sense.

Read the comments (prefaced by three semi-colons) for an explanation of the file's contents.

```
;;; -*- Syntax: common-lisp; Base: 10; Mode: LISP; Package: SYSTEM-INTERNALS; -*-

;;; a large function to do the entire job of building
;;; a world, given that you execute it in a site-configured environment

;;; use this to create a user or server world
(defun produce-world (&key server)

    (format t "~2& Enabling the EGC")

    ;;; make sure that the GC is in a consistent condition
    (gc-on :ephemeral t :dynamic nil)

    ;;; disable screen-dimmer to avoid interrupts-off surprises
  (let ((tv:*dim-screen-after-n-minutes-idle* nil))

    (setq time-start (time:get-universal-time)
    ;;; disable the services, to avoid any unwanted network interactions
    (disable-services)
    ;;; turn the global more breaks off, in case not already done
    (setq tv:more-processing-global-enable nil)
    ;;; this assures that someone is logged in to the machine
    (fs:force-user-to-login)

    (format t "~2& *** Constructing your site's world, System ~D.~2%~
       ~4TServices disabled, more processing on locally, off globally,
         logged in as ~A.~@@ ~4TLoading patches." (get-system-version) user-id)

    ;;; load patches to make sure you're up to the current patch level
    (load-patches nil :query nil)

    (format t "~2& *** Up to current patch level; loading added systems.~%")

    ;;; now load the extra systems that you want in your world
    (flet ((load-a-system (name)
               (unless (sct:find-system-named name nil t t)
                 (format t "~2&    Loading ~A.~2%" name)
                 (sct:load-system name :query :no-confirm))))

      (load-a-system "IP-TCP")
      (when server
        (load-a-system "Print")
        (load-a-system "Mailer"))

    (format t "~2& *** Added systems loaded.~%")

      ;;; compile the who-calls database
    (si:enable-who-calls :all-no-make)
```

```
;;; now do a full gc.  This takes about 1.75 hours
(format t "~2% *** Beginning Full-GC.")
(full-gc)

   (format t "~2% *** Beginning Reorder-Memory.")
      (reorder-memory :incremental nil
                      :run-without-interrupts t) ;;;run-without-interrupts is faster

;;; Last (and least) make this function disappear
(fundefine 'produce-world)

;;; Print final statistics.
(format t
        "~2& *** Full-GC, Reorder-Memory and final parameter settings complete.

        If everything~@@ ~5@@Tlooks OK, please save the result via Save World .~2%")
(values)))
```

```
(format t "~2&Start the world production by calling ")
(present '(produce-world) 'sys:form)
(format t ".~%")
```

1.10. 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 attribute 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**.

1.10.1. 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:

Time Zone	*Abbreviation*
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

Time Zone	*Abbreviation for Daylight Savings*
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

1.11. 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 a namespace object (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 site object (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.

2. Creating a Site

Unless you already have an existing site, you must create one before you can perform the Genera 7.2 software installation.

Familiarize yourself with site configuration concepts before you attempt to create a site. See the section "Site Configuration: A Conceptual Overview", page 1. Then perform the following steps:

1. Select a name for the site, and for each machine (including non-Symbolics machines). Assign a Chaosnet address to each Symbolics computer.

    Remember that each machine must have a unique Chaosnet address. Machines that are not Symbolics computers might already have network addresses for DNA or IP-TCP networks. If not, assign the appropriate network addresses to them.

    Note: All 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.

    More information is available about choosing site names. See the section "Choosing a Site Name", page 2. More information is available about Chaosnet addresses. See the section "Choosing Machine Names and Chaosnet Addresses", page 2.

2. Choose a machine to be your namespace server. More information is available about choosing this machine. See the section "Machines and Worlds", page 4.

3. 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", page 4.

4. Boot the namespace server manually. When you do this, provide the Chaosnet address that you have chosen for your namespace server. Type these commands:

    ```
    Clear Machine
    Load Microcode Microcode-file-name
    Load World World-load-file-name
    Set Chaos-Address Chaosnet-Address
    Start
    ```

5. After you have booted your namespace server, this prompt appears:

    ```
    Please type in the date and time (default is 2/4/88 12:33:52 or NONE
    to leave unspecified):
    ```

If the default date and time are correct, press <RETURN>. Otherwise, specify the correct date and time. More information is available about setting the time. See the section "Specifying a Time Zone for Your Site", page 20.

6. When the Symbolics window appears, check for the release title that's displayed. You'll need to know the title of your release while performing the installation steps in this — and in other — portions of the Genera 7.2 installation procedure.

7. Log in as the default system user. At the prompt, type:

    ```
    Login Lisp-Machine
    ```

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

8. Use the Edit File command to edit the boot file and add the correct Chaosnet address. The boot file is usually called FEP0:>Boot.boot. After editing the file, save it by pressing c-X c-S.

    More information is available about the boot file. See the section "The Booting Process", page 11. More information is available about the Edit File command. See the document *Site Operations.*

9. Log out. At the prompt, type:

    ```
    Logout
    ```

10. Halt the machine. At the prompt, type:

    ```
    Halt Machine
    ```

11. Answer Yes to the query:

    ```
    Do you really want to halt the machine (Yes or No)?
    ```

    This message will appear:

    ```
    Lisp stopped itself.
    ```

    Then you'll see the FEP Command prompt.

12. Boot the namespace server again. This time, use the boot file that you edited and saved. At the FEP Command prompt, type:

    ```
    Boot
    ```

13. Log in again as the default system user. Type:

    ```
    Login Lisp-Machine
    ```

14. If you are running Genera 7.2, complete the Define Site dialogue. Define Site creates the initial namespace database. Once you have entered data into this database, the namespace files are stored permanently. Define Site also enables the local machine to access the permanent namespace database files.

    If you are running an earlier Symbolics software release, you must use the Set Site dialogue to create the initial namespace database. Detailed information is available about the Define Site and Set Site commands.See the section "Setting and Defining Sites", page 14.

15. If this machine is the only Symbolics computer at your site, answer Yes to this query:

    ```
    Is DOWNUNDER a standalone site (there are no other servers to respond to
    a Who-am-I request)?
    ```

16. Use the namespace editor to set up the namespace; if you know there will be other Symbolics (and non-Symbolics) computers or printers at your site, add them now. Additionally, if you know the names of users at your site, add them now. At the command prompt, type:

    ```
    Edit Namespace Object
    ```

    More information is available about using the namespace editor. See the section "Updating the Namespace Database" in *Genera User's Guide*.

17. Boot each user machine, using the Chaosnet address that you assigned to it. You can use the netboot feature included in the Genera 7.2 distribution world. See the section "Netbooting User Systems", page 41.

Now that you have finished creating your site, you are ready to perform the Genera 7.2 software installation. See the section "Installing the Genera 7.2 Software", page 28.

3. Installing the Genera 7.2 Software

Genera 7.2 consists of a distribution world (on three tapes), which you must configure for your site, and of documentation, sources, examples, fonts, and non-loaded systems (on three more tapes).

This section describes how to install the Genera 7.2 software. It consists of an overview and several sections that describe how to perform the installation.

Read the overview so that you will understand any installation prerequisites, and to see a list of the installation steps you'll need to perform. After you've read the overview, go on to each section and perform the Genera 7.2 installation.

3.1. Overview for Installing the Genera 7.2 Software

Before you install Genera 7.2 at an existing site, make sure that you have first installed the Release-6-7 Compatibility software on any machines that will continue to run Release 6. See the section "Installing the Release 6-7 Compatibility Software", page 47.

Genera 7.2 includes a new feature called netbooting, which gives you the ability to boot and run a world from a remote machine.

Netbooting is accomplished through a special kind of world file called a netboot core. More information is available about netbooting. See the section "Netbooting", page 39.

Here is a list of the steps you will need to perform in order to successfully install Genera 7.2 at your site:

1.   Restore the Genera 7.2 world, netboot core, and appropriate microcode files into your FEP file system. *This step applies only to sites already running Genera 7.1 or an earlier Symbolics software release.*

2.   Make a Genera 7.2 site-configured world for your site.

3.   Restore the Genera 7.2 documentation, sources, examples, fonts, and non-loaded systems into your Lisp Machine File System (LMFS).

4.   Make customized Genera 7.2 user and server worlds for your site.

5.   Back up your world(s) to tape.

Note: Do not load Genera 7.1 software on machines running the Genera 7.2 world.

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

3.2. Restoring the Genera 7.2 World, Netboot Core, and Microcode

To restore the Genera 7.2 world, netboot core, and appropriate microcode files into your FEP file system, use the three tapes labeled *Symbolics Genera 7.2 Distribution World* (1/3, 2/3, 3/3).

1. Choose a machine to use for restoring your site's Genera 7.2 worlds. If possible, do not choose your namespace server; namespace service and restoration will slow each other down.

    Try to use a machine with a local tape drive (this will speed up the installation process considerably). In summary, your best choice is a machine that:

    - Has a tape drive.

    - Has enough free disk space to boot Genera 7.2 (about 40,000 blocks).

    - Is not a critical machine at your site (for example, is not the namespace server).

    If no machine at your site fits all three criteria, find one machine that has a tape drive (the remote machine), and another machine that has enough disk space to boot Genera 7.2 (the local machine). Make sure that the local machine isn't a critical machine at your site.

    You can restore the new distribution world to the local machine's FEP, using the remote machine's tape drive. (This procedure is somewhat slower than restoring Genera 7.2 from the local machine's tape drive.)

2. Insert the first *Genera 7.2 Distribution World* tape (1/3) into the tape drive. Type the CP command Select Activity FEP-Tape or, if you are running Release 6, use **tape:fep-file-restore**.

    If the tape is in another machine's cartridge drive, type

    > Read Tape :host remote-host-name.

    If the tape is in the local machine's cartridge drive, click on [Read Tape].

3. The FEP-Tape program first asks you whether you want to load each microcode file. The tape contains a complete set of microcodes for Genera 7.2. Before you begin to restore the microcodes, you'll need to know the microcode types for each machine model (and the microcode pathname to restore). More information is available about microcode naming conventions. See the section "Genera 7.2 Microcode Types", page 58.

    For each microcode type, you are asked whether or not you want to restore the file. Here is an example of what you see on the screen. In this example,

the microcode is for a 3640 machine with floating-point accelerator (FPA):

```
Filename: SYS:L-UCODE;3640-FPA-MIC.MIC.419
Length: 127843 bytes (112 blocks)
Created: 2/05/88 17:28:33
Author: Lowe
Restoration Comment: For microcode 419
Restore this file?  (Y or N)
```

If you answer N (for No) the program skips the microcode file and prompts you for the next version. If you answer Y (for Yes) the program asks you:

```
Into the file (default is FEP0:>3640-FPA-MIC.MIC.419)
```

When you respond to this question, accept the default pathname, or specify a FEP pathname that indicates microcode, such as

```
FEP0:>3640-fpa-mic.mic.419
```

on the local machine. If you are planning to boot Genera 7.2 on a different host (machine), use the pathname that includes the disk unit number plus the name of the host. For example:

```
Host-name|FEP0:>3640-fpa-mic.mic.419
```

After the microcode files are presented, and you have restored the appropriate one, 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-7-2.load`, if the *local* machine will boot Genera 7.2, or

- `Host-name|FEP0:>Genera-7-2.load`, if you want to boot Genera 7.2 on *another* machine.

The FEP-Tape loading program then loads the world from the tapes in the Genera 7.2 distribution world tapes, prompting you to insert a new tape when the end of each tape has been reached.

After the world has been restored, you are asked if you want to restore the netboot core. If you want to use netbooting, answer Y (for Yes) and accept the default pathname or specify a FEP pathname that indicates the Genera 7.2 netboot core, such as

```
FEP0:>Netboot-Core-from-Genera-7-2.load
```

More information is available about netbooting. See the section "Netbooting", page 39.

4. Create and save a new boot file (for example, FEP0:>rel-7-2.boot) on the machine on which you loaded both the Genera 7.2 world and the Genera 7.2 microcode. More information is available about booting and boot files. See the section "The Booting Process", page 11.

Be sure that the new boot file specifies the appropriate microcode that you restored from tape. We recommend that you insert the Enable IDS command after the Load World or Netboot command in the boot file.

Here is a recommended sequence of commands for a boot file using Load World:

```
Clear Machine
Declare Paging-files paging-file-1 paging-file-2
Load Microcode microcode-file-name
Load World world-load-file-name
Enable IDS
Set Chaos-Address Chaosnet-address
Set Ethernet-Address Ethernet-address (if your machine runs DNA)
Start
```

Now that you have finished restoring the world, netboot core, and microcode, you are ready to make a Genera 7.2 site-configured wòrld.

3.3. Making a Genera 7.2 Site-Configured World

To make a Genera 7.2 site-configured world, perform the following steps:

1. Edit (update) the SYS:SITE;SYS.TRANSLATIONS file. Because the new software has been recompiled, system files for Genera 7.1 and Genera 7.2 must be kept in separate directories.

    More information is available about updating the SYS.TRANSLATIONS file. See the section "Updating the SYS.TRANSLATIONS File", page 32.

2. Boot the Genera 7.2 distribution world on the machine that has the updated boot file, and on which you have restored the Genera 7.2 world and the Genera 7.2 microcode.

3. Log in as the default system user. At the command prompt, type:

    ```
    Login Lisp-Machine
    ```

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

4. Use the Set Site command to configure the machine for your site. More information is available about the Set Site command. See the document *Site Operations.*

    If your site has only one machine, use the Set Site command like this (where *Koala* is the name of the local host and *Downunder* is the name of the local site):

```
Set Site (site name [default Get from network]) Downunder
(Server host name [default local]) local
What host is a namespace server for DOWNUNDER (default: Local): Local
Where is the descriptor file for DOWNUNDER
(default local:>sys>site>down-under-namespace.text):
local:>sys>site>down-under-namespace.text
Assuming that KOALA is the real pathname host name of the local host.
Warning: the local host, DIS-LOCAL-HOST, was not known to have that name.
[11:14:55 Namespace on DIS-LOCAL-HOST: Reloading namespace DOWNUNDER.
         Recent servers contacted are DIS-LOCAL-HOST]
The local host is now KOALA.
[11:15:00 Namespace DOWNUNDER has become unloaded:
         No longer server for this namespace.]
[11:15:14 Namespace on KOALA: Reloading namespace DOWNUNDER.
         Recent servers contacted are KOALA]
```

If your site has multiple machines, use the Set Site command like this (where *Koala* is the name of the local host and *Downunder* is the name of the local site):

```
Set Site (site name [default "Get from network]) "Get from network"
CHAOS|24451 thinks the local host is DOWNUNDER|KOALA.
Setting site to DOWNUNDER.
The local host is now KOALA.
```

5. 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 document *Site Operations*. This example shows how to save an incremental version of the world:

```
Save World Incremental FEP0:>Inc-Site-from-Genera-7-2.load
```

This world will serve as the base world for all Genera 7.2 worlds you build at your site, except for the namespace server's world. Information is available about making user and server worlds for your site. See the section "Making Customized Genera 7.2 User and Server Worlds", page 37.

3.3.1. Updating the SYS.TRANSLATIONS File

Before restoring the Genera 7.2 Basic Distribution Tapes, edit the file SYS:SITE;SYS.TRANSLATIONS so that Genera 7.2 will "know" where its sources reside. Name the top level directory Rel-7-2 for a smooth upgrade path.

Genera 7.2 checks the namespace site object to find the physical location of the logical directory SYS:SITE; . The software imposes no restriction on the physical name of the site directory. It is helpful, however, to give the site directory the name >sys>site>.

Genera 7.2 software supports "wild-inferiors". This means that wildcard matching can occur at any level in a directory.

Update the site definitions by reading in and then editing the file SYS:SITE;SYS.TRANSLATIONS. Once you have finished editing the SYS.TRANSLATIONS file for your site, save and load the file.

Updating the SYS.TRANSLATIONS File for Genera 7.2 Only

If the machines at your site will be running only Genera 7.2, use this example as a model for your SYS.TRANSLATIONS file:

```
;;; -*- Mode: LISP; Package: FS; Syntax: ZetaLisp; Base: 10; -*-
;; Translations for Genera 7.2
(fs:set-logical-pathname-host "SYS"
    :physical-host "ACME-YUKON"
    :translations '(("SYS:**;*.*.*" "ACME-YUKON:>Rel-7-2>**>*.*.*")))
```

Updating the SYS.TRANSLATIONS File for Release 6, and/or Genera 7.1, and Genera 7.2

If the machines at your site will continue to use Release 6 and/or Genera 7.1 after you have installed Genera 7.2, use this example as a model for your SYS.TRANSLATIONS file. (This example shows a SYS.TRANSLATIONS file that's been conditionalized to allow Release 6 and/or Genera 7.1 to coexist with Genera 7.2 (each with its own translations):

```
;;; -*- Mode: LISP; Package: USER; Lowercase: Yes; Syntax: Common-lisp -*-

(select (si:get-release-version)
  ;; Translations for Release 6
  (6 (fs:set-logical-pathname-host "SYS"
        :translations '(("SYS:**;*.*.*" "ACME-YUKON:>REL-6>**>*.*.*"))))
  ;; Translations for Genera 7
  (7 (let ((release-minor-version
              (second (multiple-value-list (si:get-release-version)))))
        (select release-minor-version
        ;; Translations for Genera 7.1
        (1 (fs:set-logical-pathname-host "SYS"
              :translations '(("SYS:**;*.*.*" "ACME-YUKON:>REL-7>**>*.*.*"))))
        ;; Translations for Genera 7.2
        (2 (fs:set-logical-pathname-host "SYS"
              :translations '(("sys:**;*.*.*" "ACME-YUKON:>REL-7-2>**>*.*.*"))))
        (otherwise (error "~D unknown release or system version.~2T
                          Edit SYS:SITE;SYS.TRANSLATIONS to add this system."
                          release-minor-version))))))
```

3.3.1.1. 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 Release 6 and/or Genera 7.1 sources, you may have to employ a workaround.

To employ the workaround, change the SYS.TRANSLATIONS file so that Release 6 and/or Genera 7.1 can find files in the Genera 7.2 sources. *This workaround is limited to fonts, microcode, site files, and FEP flod file directories. Do not use the workaround on documentation, Lisp sources, or binaries.*

For example, if you have to delete the Release 6 and Genera 7.1 font files from your SYS host, modify the translations file as shown in bold (this translation allows users of Release 6 and Genera 7.1 systems to find screen fonts and to print jobs on printers spooled from Genera 7.2 systems):

```
;;; -*- Mode: LISP; Package: USER; Lowercase: Yes; Syntax: Common-lisp -*-


(select (si:get-release-version)
  ;; Translations for Release 6
  (6 (fs:set-logical-pathname-host "SYS" :translations
        '(("sys:fonts;**;*.*.*" "ACME-YUKON:>REL-7-2>fonts>**>*.*.*")
          ("SYS:**;*.*.*" "ACME-YUKON:>REL-6>**>*.*.*"))))
  ;; Translations for Genera 7
  (7 (let ((release-minor-version
             (second (multiple-value-list (si:get-release-version)))))
       (select release-minor-version
         ;; Translations for Genera 7.1
         (1 (fs:set-logical-pathname-host "SYS" :translations
               '(("sys:fonts;**;*.*.*" "ACME-YUKON:>REL-7-2>fonts>**>*.*.*")
                 ("SYS:**;*.*.*" "ACME-YUKON:>REL-7>**>*.*.*"))))
         ;; Translations for Genera 7.2
         (2 (fs:set-logical-pathname-host "SYS"
               :translations '(("sys:**;*.*.*" "ACME-YUKON:>REL-7-2>**>*.*.*"))))
         (otherwise (error "~D unknown release or system version.~2T
                            Edit SYS:SITE;SYS.TRANSLATIONS to add this system."
                           release-minor-version))))))
```

It is possible to keep system documentation, sources, examples, fonts, and non-loaded systems in files that reside on non-Symbolics machines. See the section "Special Installation Situations", page 47.

Now that you have made a site-configured world, you are ready to restore the Genera 7.2 documentation, sources, examples, fonts, and non-loaded systems.

3.4. Restoring the Genera 7.2 Documentation, Sources, Examples, Fonts, and Non-Loaded Systems

To restore the Genera 7.2 documentation, sources, examples, fonts, and non-loaded systems, use the three tapes labeled *Genera 7.2 Basic Distribution* 1/3, 2/3, 3/3).

You may use a non-Symbolics machine to store the Genera 7.2 documentation, sources, examples, fonts, and non-loaded systems, but you'll need more information in order to do so. See the section "Storing Documentation, Sources, Examples, Fonts, and Non-Loaded Systems on a Non-Symbolics Machine", page 50.

1. Boot the Genera 7.2 site-configured world and put the first of the three *Genera 7.2 Basic Distribution* tapes into the tape drive.

2. Use the following command to restore the files from that tape (and later, from the second and third tapes):

    ```
    Restore Distribution :Menu Yes
    ```

    This displays a menu, as shown in Figure 1.

3. Click on [Initialize Restoration]. Each time you insert a tape, use the mouse to deselect systems that you do not want to restore from the menu.

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

    a. All sites *must* restore the following systems:

        NFEP-Overlays
        Microcode

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

        PS (Print Spooler)
        ML (Mailer, for using mail at your site)
        Domain Name Server (Name Server, for sending mail to other sites)

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

        Hacks (demos)
        Conversion-Tools (Zetalisp to Common Lisp)
        Examples
        Metering
        Serial-Networks
        FED (Font Editor; in previous releases this was included in the standard distribution world)
        Unsupported (unsupported software such as the old Namespace Editor)

d.   If you want up-to-date online sources and documentation, you should restore the following systems:

System
Utilities
SU (Server Utilities)
Hardcopy
Zmail
LMFS
Tape
NSage
Doc (Documentation)

Restore Distribution	
Help **Perform Restoration**	Files to Restore
Initialize Restoration	
Skip restoration of files that already exist:	In system LEGAL-NOTICE:
Yes No	SYS:SITE;NOTICE.TEXT.42
Write informational output to:	In system SYSTEM:
Standard-Output	SYS:SYS;SYSDCL.LISP.849
Read distribution from tape or disk:	SYS:PATCH;SYSTEM.SYSTEM-DIR.185
Tape Disk	SYS:PATCH;SYSTEM-376.COMPONENT-DIR.2
Spec for tape:	SYS:SYS2;DEFMAC.LISP.102
NUTHATCH: cart, den=1600	SYS:SYS2;LMMAC.LISP.546
	SYS:SYS2;STRUCT-DEFS.LISP.7
	SYS:SYS2;STRUCT.LISP.371
Actions during Restore Distribution	SYS:SYS2;SETF.LISP.72
	SYS:SYS2;LOOP.LISP.815
	SYS:CLCP;MACROS.LISP.37
Systems to Restore	SYS:IO;RDDEFS.LISP.66
	SYS:SYS;WIRED-EVENT-DEFS.LISP.3
LEGAL-NOTICE	SYS:SYS;AARRAY.LISP.36
SYSTEM	SYS:SYS2;ADVISE.LISP.53
UTILITIES	SYS:SYS2;CHARACTER-SETS.LISP.60
HACKS	SYS:SYS2;CHARACTER-STYLES.LISP.197
SERVER-UTILITIES	SYS:SYS;COLD-LOAD.LISP.207
HARDCOPY	SYS:SYS;COMMAND-LOOP.LISP.156
ZMAIL	SYS:SYS;EXPAND-DO.LISP.8
LMFS	SYS:IO;DRIBBL.LISP.54
TAPE	SYS:SYS2;ENCAPS.LISP.51
NSAGE	SYS:SYS;EVAL.LISP.299
PRINT	SYS:IO;FORMAT.LISP.354
MAILER	SYS:SYS;FSPEC.LISP.268
DOMAIN-NAME-SERVER	SYS:IO;GRIND.LISP.152
CONVERSION-TOOLS	SYS:SYS2;HASH.LISP.115
METERING	SYS:SYS2;HASH-COMPATIBILITY.LISP.1
METERING-SUBSTRATE	SYS:SYS2;HEAP.LISP.24
NFEP-OVERLAYS	SYS:IO;INDENTING-STREAM.LISP.28
MICROCODES	SYS:IO1;INFIX.LISP.35
DOC	SYS:IO;INTERACTIVE-STREAM.LISP.233
EXTENDED-HELP	SYS:IO;INPUT-EDITOR.LISP.310
Systems to Restore	*Files to Restore*

Restore: *Initialize Restoration*
Tape rounted on drive CART.
Restore:

Mouse-L, -R: Restore the selected files to the filesystem.
To see other commands, press Shift, Control, Meta-Shift, or Super.
[Thu 14 Jan 6:08:59] Lown   CL USER: User Input

Figure 1.   System Selection Menu for the Restore Distribution Command

4.   Copy the new version of NFEP overlay files (flod files) to every Symbolics computer at your site.

From the machine running the Genera 7.2 world, copy the overlay files (flod files) from sys:n-fep; onto the FEP file system. Use the Copy Flod Files command to do this. More information is available about the Copy Flod Files command. See the document *Site Operations*.

For example, for Symbolics models 3600, 3640, 3645, 3670, and 3675, type

```
Copy Flod Files :version V127 :hosts host1, host2
```

For Symbolics models 3610, 3620, 3630, 3650, and 3653, type:

```
Copy Flod Files :version G206 :hosts host3, host4
```

or

```
Copy Flod Files :version G208 :hosts host3, host4
```

To find out the correct FEP EPROM version number for these machine models (G206 or G208), use the FEP command Show Version.

5. Halt each machine with the Halt Machine command.

6. Scan the new flods with the Hello command. More information is available about the Hello command. See the document *Site Operations*.

7. Return to Genera with the Continue command.

Now that you have restored the documentation, sources, examples, fonts, and non-loaded systems, — and you have copied the new flods to all machines at your site — you are ready to make customized Genera 7.2 user and server worlds.

3.5. Making Customized Genera 7.2 User and Server Worlds

Follow these instructions to make customized worlds containing the software you will be using at your site.

3.5.1. 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. (Check to see that your boot file contains the command Enable IDS immediately after the Load World command.) 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 (including non-loaded systems like Hacks, Conversion-Tools, Metering, or the Mailer). For example, to load the Print Spooler, type:

    ```
    Load System Print
    ```

5. Optimize the world using the Optimize World command:

    ```
    Optimize World
    ```

    The (optional) Optimize World command reorganizes the world to improve paging performance. More information is available about the Optimize World command. See the document *Site Operations*.

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:

    ```
    Save World Incremental FEP0:>Inc-Color-7-2-from-Inc-Site.load
    ```

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

7. Once you have created these worlds, you can move them out to the appropriate machines. You have a choice of using the Copy World command or of netbooting.

    If you make an IDS world, all machines must also have the parent world (which is your site's version of the Genera 7.2 world from which the IDS world was made). If you are using Copy World, the parent world must be present on each machine's FEP file system. If you are netbooting, the parent world must be present on the netboot server. More information is available about netbooting. See the section "Netbooting", page 39.

    If the parent worlds are on disk, the names appear when you use the Show FEP Directory command. Here are some worlds you might have on disk:

    • A distribution world.

    • A site-configured version of the world.

    • An incremental version of the world, specifically created for the user machine(s).

    • An incremental version of the world, specifically created for the server machine(s).

    • An incremental version of the world, specifically created for the namespace server machine.

8. If the user machines at your site will all run different software, create individual incremental worlds for each user machine. Base these worlds on your site-configured Genera 7.2 world.

    Alternatively, you can make one version of the world that contains both the site-configured software and any special software. To do this, create the site-configured world and then:

    • Load additional software

    • Use the Optimize world command to optimize the world

    • Use the Save World Incremental command to save an incremental version of the world.

3.5.2. Netbooting

Netbooting is the ability to boot a world from a remote machine and run it without having the booted world-load file on the local disk.

Netbooting should be considered a site-management tool, rather than a disk-conservation tool. While you do not have to have a world-load file on your local disk, you should add an extra paging file of the same size (approximately 35,000 blocks) if you want equivalent performance. On the other hand, you never need to have two worlds on your local disk.

Each user needs only a netboot core (and a backup netboot core) on the local disk to take advantage of netbooting. Netboot cores have the .load file type, but use only approximately 100 blocks each. Of course, you continue to have other files on the local disk, such as microcode files, FEP overlays (flods), paging files, boot files, and the like.

There are no clear rules on when netbooting is the correct choice. Symbolics is interested in hearing from users about how they use netbooting.

In the meantime, some general principles can be suggested.

The choice of booting methods is a site-management issue. The parameters are the number of machines at the site, the number of worlds commonly used, and the number of users.

Netbooting a world is slower than booting a world from the local disk. On the other hand, netbooting is much faster than copying a world to the local disk and then booting it. This means that users who reboot the same world frequently may prefer booting from the local disk.

Users who run many different worlds on the same machine may want to pay the price of slower boot time in exchange for eliminating the need to constantly juggle paging and world space on the local disk.

If several worlds are in common use at your site, netbooting may be a better choice because you'll only have to maintain two copies of each world (one of them

a backup) per site, instead of one copy for each user. Netbooting is particularly useful if systems loaded in those worlds are patched frequently.

Here is the setup for a site using netbooting.

- One or more netboot servers with world-load files for all worlds used at the site.

- User machines, each with a netboot core. This is a special kind of world-load file used to accomplish netbooting. Each user machine must also have a boot file including the Netboot command at the point that the Load World command usually appears.

Here are the limitations on netbooting:

- Netboot servers must run Genera 7.2 or a later release.

- Only worlds based on Genera 7.2 or a later release can be netbooted.

- Netbooting works only on the local Ethernet. You can netboot only when there is a netboot server on the same subnet.

3.5.2.1. Netboot Servers

Here is how you set up a system as a netboot server:

- Use the Copy Flod Files command to copy the new NFEP-Overlays to the FEP file system on the server running the Genera 7.2 world.

- Add the following services to the server's Host Object in your local namespace:

```
Service: NETBOOT SLAP NETBOOT
   . . .
Server Machine: Yes
```

The effect of Server Machine: Yes is to cause the server to boot with services disabled. This prevents any user system from netbooting before the server itself is completely booted. This means you must enable services after the server system is booted.

See the section "Enable Services Command" in *Genera Handbook*. See the function **sys:enable-services** in *Networks*.

- Copy all world-load files that users will be netbooting to a top-level FEP directory. This can be any top-level directory on the netboot server.

When you enable services on the netboot server, netboot service is enabled by default. You can also enable netboot service by name.

Netbooting includes a queuing mechanism. Once a user machine requests a world to netboot, no further user attention is necessary.

Netboot service is provided serially, on a first-come-first-serve basis. This minimizes boot time. Sites with many machines may find this a problem when they all try to boot at once, such as after a power failure.

Therefore you will probably want to have a backup netboot server, but it need not have netboot service enabled. If the primary netboot server goes down, you must enable netboot service on the backup so users can access the worlds there.

A server can be a user machine, but the system may be slow while it is serving up a world. A user machine is fine as a backup server. Symbolics does not recommend using a file server as a netboot server because file service may be severely degraded when the server is also serving a netbooting client.

You can have two netboot servers without having identical sets of worlds on them. The queuing mechanism can find a world on any netboot server.

If you have the same world on several servers, take care that the worlds are really the same, that is, at the same patch level. Otherwise, there is some possibility of booting the older version. If you decide to stop supporting a world on a particular server, you should delete all copies of the world. They need not be expunged; marking them for deletion is good enough.

3.5.2.2. Netbooting User Systems

Here is how you set up a user machine for using netbooting:

- Use the Copy Flod Files command to copy the Genera 7.2 FEP overlay files (flods) to the FEP file system. See the section "Copy Flod Files Command" in *Genera Handbook*. Copy the netboot core to the FEP file system of the local machine using the Copy World command. This file has the same name as the distribution world, prefixed by `Netboot-core-from-`, such as `Netboot-Core-from-7-2.load`, and should be found on the FEP file system of the same machine as the distribution world.

  If you have multiple disks, it is best to put the core on FEP0. (But if you explicitly mount another disk in your boot file, before the Netboot command, you can safely put the netboot core on that disk.)

- Add extra paging space, equivalent to the size of the world you will netboot. This paging space is in addition to the paging space you normally use. You must add this extra paging space if you want the netbooted world to have equivalent performance to a world booted from the local disk.

Then, to netboot, use the FEP command Netboot in your boot file where you would normally use the Load World command.

Netboot

Netboot *world-description*

> Netboots the world described by *world-description*. Polls the netboot servers on the local subnet for worlds that match

world-description and netboots the most recent of those worlds.

Here is an example:

```
Netboot chip-poker
```

In effect, the Netboot command replaces Load World if you are netbooting.

See the section "Netbooting", page 39.

The Load World command can initiate netbooting in certain circumstances. See the section "Netbooting IDS Worlds", page 44.

The Load World command is loaded when you scan the FEP overlay file *-loaders.flod.

World Description for Netbooting

The Netboot command accepts world descriptions, rather than the pathname of a world-load file. The world description can be the initial substring of the name of the .load file of the world you wish to netboot. Users can netboot using a boot file with the proper world identifier in the name without having to keep track of the correct name of the most recent version of the world they use.

The world description is not the same as a pathname, but it is based on the file-name portion of a world-load pathname. The world description never includes the host name or FEP unit. You should employ a system of world-naming at your site that puts the most significant information about the world at the beginning of the name. This is because finding a match for the world description starts at the beginning of the name of each world-load file on all netboot servers.

The more specific your world description is, the greater the likelihood of netbooting the world you want.

The Wrong Way to Describe Worlds

To illustrate the possible problems that unsystematic naming of worlds can cause, let's assume you have two related world-load files on your netboot server. Their names are

```
poker-chip.load
poker-game.load
```

If you use the command

```
Netboot poker
```

you will boot the most recently created of these two files.

On the other hand, if you use the command

```
Netboot Poker-chip
```

you will boot the most recently created world called poker-chip.

The Right Way to Describe Worlds

You would be much better off naming the two worlds

chip-poker
game-poker

Then you can use the commands

```
Netboot chip
```

or

```
Netboot game
```

without fear of ending up with the wrong world.

Here is a sample boot file with a recommended sequence of commands for using Netboot:

```
Clear Machine
Declare Paging-files  fep0:>page.page aux.page betty.page satchel.page
Load Microcode FEP0:>3640-mic.mic.419
Netboot chip-poker
Enable IDS
Set Chaos-Address 14157
Start
```

(If your system runs DNA, the boot file should also include a Set Ethernet-Address command.)

A world matching the world description is netbooted.

User machines that have been netbooted do not need any world in their FEP file system except the netboot core (and a backup copy of the core). If you have a Genera 7.2 world-load file on your FEP, you can use the world-load file as a netboot core. That is, any Genera 7.2 world load can be used as a netboot core, but the netboot core files are much smaller.

When you netboot, the console screen goes blank and then provides an attenuated narrative of what is happening. In most cases, you won't need this information.

You can see how far along the netbooting has proceeded by observing the progress bar on the lower right of the screen. The leading dotted line indicates how many pages of the world have been requested to be transferred and the trailing solid line indicates how much of the world has actually been transferred. The progress-bar label describes what phase the netboot process is in. In addition, there are short bars indicating, from left to right, Disk-wait, CPU-run, and Net-wait.

You can halt netbooting at any time by pressing h-c-FUNCTION, as indicated in the upper left of the screen.

If, for any reason, your netboot core does not work, you may need to use the FEP command Set World-to-Netboot.

Set World-to-Netboot

Set World-to-Netboot *world-description*

> Replacement for the Netboot command in certain rare circumstances.

> Usually, the Netboot command selects the "best" world to use as a netboot core. If the "best" core does not, for some reason, seem to work, you can use Load World to load another world (Genera 7.2 or later) to use as a netboot core. In that case, you use the Set World-to-netboot command to netboot your *world-description* using the loaded world as a core rather than selecting the "best" core.

> In other words, this command uses the currently loaded world as a core to netboot your *world-description*, rather than searching for a "better" one. See the section "World Description for Netbooting", page 42.

> This command is loaded when you scan the FEP overlay file *-loaders.flod.

Netbooting IDS Worlds

There is no limit on using netbooting with IDS worlds. If the Netboot command in your boot file identifies an IDS world on a netboot server, all the parents of that world are also netbooted, provided they are on the same server.

You can also combine loading an IDS world from your local disk with netbooting. If you are accustomed to using a standard world for your site with your own custom IDS built on top of it, you can continue to do so. In this case, however, use the Load World command and specify the pathname of the IDS world on your local disk in your boot file. Do not use the Netboot command in this case.

Note: Not having all parents of an IDS world on the local disk was once an error, but is now interpreted as a request to netboot the missing parents.

Two conditions must be met to use this technique:

- You have an IDS world on your local disk, but not all its parents.

- All the remaining parent world loads are available on the same enabled netboot server.

In this case, the Load World command in the boot file loads the IDS world normally and searches the local disk for its parents. When some parent is not found, net-

booting is initiated to boot that and all remaining parent worlds. When any parent world on the netboot server is deleted, you must, of course, create a new IDS on your local disk. You can netboot the new parent worlds and then create the new IDS.

Note: If there is no enabled netboot server on the local subnet, the Start command will start Lisp and Lisp will wait forever for a netboot server. No error is returned. If your site does not support netbooting, you will have to use h-c-FUNCTION to halt the netboot process and get back to the FEP and boot from the local disk.

3.5.3. 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 of 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.

Now that you have made customized user and server worlds, you are ready to back up your world to tape.

3.6. 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
   ```

   Or, you can click on [Write Tape].

Now that you have backed up the world(s) to tape, your Genera 7.2 installation is complete.

4. Special Installation Situations

During site configuration and your Genera 7.2 software installation, you may encounter special situations. Such situations include:

- Running Release 6 and/or Genera 7.1 at your site, along with Genera 7.2.

- Adding Non-Symbolics Computers to Your Site.

- Storing documentation, sources, examples, fonts, and non-loaded systems on a non-Symbolics computer.

- Installing the Genera 7.2 software on a 140-Mbyte disk.

This chapter contains information about these special situations.

4.1. Installing the Release 6-7 Compatibility Software

Use these instructions to load the Release 6-7 Compatibility software onto all machines that will continue to run any Release 6 world at the same time other machines at your site are running Genera 7.2.

1. In a Release 6 world, restore the Release 6 distribution tape containing the Release 6-7 Compatibility software (patches) into your file server's LMFS. Use the function **dis:load-distribution-tape**. The patches are loaded into the logical directory SYS:REL-6-7; .

2. When you use the function **dis:load-distribution-tape**, a menu appears. This menu contains the names of all the systems to load. Click on [Do It] to load all systems.

    These systems contain the Release 6-7 Compatibility patches, and the newest versions of the v127 and G206 FEP overlay files (flods). The overlay files are loaded into the logical file sys:n-fep; .

3. Copy the appropriate files for your machine to FEPn (where n is the disk unit number). If you have a 3620 or 3650 machine, you should use the G206-* flods. Otherwise, use the v127-* flods.

    If you do not know what machine model you have, use the Show Herald command. This command displays the machine model for the local machine. On each machine at your site, use the Copy Flod Files command to copy the overlay files from sys:n;fep; into the FEP file system. In this example, you

copy the overlay files into FEP unit 0 of your machine:

```
Copy File SYS;n-fep;v127-*.flod.newest FEP0:>
```

4. Make sure that you have a hello.boot file on your FEP. The hello.boot file scans the overlay files and makes them available for use. If your machine is a 3620 or 3650, your hello.boot file should look like this:

```
Scan FEP0:>G206-info.flod
Scan FEP0:>G206-loaders.flod
Scan FEP0:>G206-lisp.flod
Scan FEP0:>G206-debug.flod
Initialize Hardware Tables
```

If you have a machine other than a 3620 or a 3650, your hello.boot file should look like this:

```
Scan FEP0:>V127-info.flod
Scan FEP0:>V127-loaders.flod
Scan FEP0:>V127-lisp.flod
Scan FEP0:>V127-debug.flod
Initialize Hardware Tables
```

If you do not have the appropriate hello.boot file, create it (and then save the file). More information is available about hello.boot files. See the section "FEP System hello.boot File" in *Site Operations*.

5. Log out. At the prompt, type:

```
Logout
```

6. Halt the machine. At the prompt, type:

```
Halt Machine
```

7. Answer Yes to the query:

```
Do you really want to halt the machine (Yes or No)?
```

This message will appear:

```
Lisp stopped itself.
```

Then you'll see the FEP Command prompt.

8. Scan the new flods with the Hello command. More information is available about the Hello command. See the document *Site Operations*.

9. Return to Genera with the Continue command.

10. Build incremental worlds on top of the existing Release 6 user and server worlds at your site. All of these incremental worlds must contain the Release 6-7 Compatibility System.

Follow these instructions to build user and server worlds on non-namespace server machines and to build namespace server worlds on namespace server machines:

a.   Boot an existing Release 6 world (be sure to use one with the appropriate systems loaded).

Make sure that the boot file you're using for this has the Enable IDS command in it (the Enable IDS command must be located after the Load World command and before the Start command within the boot file).

b.   At the prompt, type the Disable Services command:

```
Disable Services
```

c.   Turn on the Ephemeral Garbage Collector (EGC). At the prompt, type:

```
Start GC :Ephemeral Yes
```

d.   Log in as the default system user. At the prompt, type:

```
Login Lisp-Machine
```

e.   Load the Release 6-7 Compatibility System into virtual memory. At the prompt, type:

```
Load System Rel-6-7
```

f.   Use the Save World Incremental command to save an incremental version of the world. At the prompt, type:

```
Save World Incremental FEPn:>Inc-Rel-6-7-from-Site-6-1.load
```

More information is available about the saving incremental worlds with the Save World command. See the document *Site Operations*.

11.   Copy the appropriate user, server, and namespace server worlds to any machines that will require the Release 6-7 Compatibility software in their worlds.

4.2. 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 document *Site Operations*.

- 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 window. For information about the properties of a new host: See the document *Networks*.

4.3. Storing Documentation, Sources, Examples, Fonts, and Non-Loaded Systems on a Non-Symbolics Machine

If you are unable to store the documentation, sources, examples, fonts, and non-loaded systems on a Symbolics machine, you can use a non-Symbolics server.

To use a VAX/VMS server for source files, your Symbolics machine(s) must run DECnet (DNA). To use a UNIX server for source files, your Symbolics machine(s) must run IP/TCP (unless you are running Berkeley UNIX 4.2 with the Chaosnet Package).

Install the appropriate network product and use the sample translations files contained within this section.

More information is available about the translations files and non-Symbolics hosts. See the section "Updating the SYS.TRANSLATIONS File", page 32. Also: See the section "Logical Pathnames and the SYS Host", page 3.

More information is available about installing DNA and IP/TCP. See the section "DNA Installation" in *Symbolics Digital Network Architecture (DNA) Software Package*. See the section "IP/TCP Installation" in *Symbolics IP/TCP Software Package*.

Here is a sample translations file for a VAX/VMS operating system:

```
(fs:set-logical-pathname-host "SYS"
    :translations '(("SYS:**;*.*.*" "ACME-VMS:SYMBOLICS:[REL7-2...]*.*;*")))
```

Here is a sample translations file for a UNIX operating system:

```
(fs:set-logical-pathname-host "SYS"
    :translations '(("SYS:*;*.*.*" "ACME-UNIX://usr//symbolics//rel-7-2//*//*.*.*")
                   ("SYS:*;*;*.*.*" "ACME-UNIX://usr//symbolics//rel-7-2//*//*//*.*.*")
                   ("SYS:*;*;*;*.*.*"
                    "ACME-UNIX://usr//symbolics//rel-7-2//*//*//*//*.*.*")
                   ("SYS:*;*;*;*;*.*.*"
                    "ACME-UNIX://usr//symbolics//rel-7-2//*//*//*//*//*.*.*")))
```

Here is a sample translations file for storing files simultaneously on VAX/VMS, UNIX, and a Symbolics computer:

Note: In this example, all directories and subdirectories on sys: are mapped to subdirectories of ACME-YUKON:>Rel-7-2>. Because of disk-space limitations we put

the documentation files on ACME-VMS, some of the larger source directories on ACME-UNIX, and the rest on ACME-YUKON.

```
;;; -*- Mode: LISP; Package: FS; Syntax: ZetaLisp; Base: 10; -*-


;; Translations for Genera 7.2
(fs:set-logical-pathname-host "SYS"
    :translations '(("SYS:DEBUGGER;*.*.*" "ACME-UNIX://usr//rel7-2//debugger//*.*.*")
                    ("SYS:IO;*.*.*" "ACME-UNIX://usr//rel7-2//io//*.*.*")
                    ("SYS:IO1;*.*.*" "ACME-UNIX://usr//rel7-2//io1//*.*.*")
                    ;; LMTAPE; has subdirectories
                    ("SYS:LMTAPE;*.*.*" "ACME-UNIX://usr//rel7-2//lmtape//*.*.*")
                    ("SYS:LMTAPE;*;*.*.*" "ACME-UNIX://usr//rel7-2//lmtape//*//*.*.*")
                    ("SYS:LMTAPE;*;*;*.*.*"
                     "ACME-UNIX://usr//rel7-2//lmtape//*//*//*.*.*")
                    ("SYS:NETWORK;*.*.*" "ACME-UNIX://usr//rel7-2//network//*.*.*")
                    ("SYS:SYS;*.*.*" "ACME-UNIX://usr//rel7-2//sys//*.*.*")
                    ("SYS:SYS2;*.*.*" "ACME-UNIX://usr//rel7-2//sys2//*.*.*")
                    ("SYS:WINDOW;*.*.*" "ACME-UNIX://usr//rel7-2//window//*.*.*")
                    ("SYS:ZWEI;*.*.*" "ACME-UNIX://usr//rel7-2//zwei//*.*.*")
                    ;; All documentation goes on ACME-VMS
                    ("SYS:DOC;**;*.*.*" "ACME-VMS:DISK$USER:[REL7-2.DOC...]*.*.*")
                    ;; Examples go on ACME-VMS, too.
                    ("SYS:EXAMPLES;**;*.*.*" "ACME-VMS:DISK$USER:[REL7-2.DOC...]*.*.*")
                    ;; Everything else goes on Yukon
                    ("SYS:**;*.*.*" "ACME-YUKON:>Rel-7-2>**>*.*.*")))
```

Because sys:lmtape; has subdirectories, and because UNIX does not support the "wild-inferior" type of multiple-directory-level wildcard, there is an entry for each directory level.

Otherwise, we would have placed these directories on ACME-YUKON, because only the last entry in the translation table would match.

The following directories include subdirectories:

sys:patch;
sys:demo;
sys:server-utilities;
sys:hardcopy;
sys:zmail;
sys:lmfs;
sys:lmtape;
sys:nsage;
sys:print;
sys:mailer;
sys:ip-domain-server;
sys:conversion-tools;
sys:doc;
sys:serial-networks;
sys:fonts;

Three logical directory levels are sufficient for these. For example:

sys: lmtape;
sys: lmtape; *;
sys: lmtape; *; *;

Some subdirectories can, in turn, contain an additional level of subdirectory. Most of the system software does not currently use more than three levels of logical directory, but the following systems need four logical subdirectory levels:

sys:window;
sys:metering;

Once you have edited the translations file, save and then load it.

4.3.1. VAX/VMS Version 4.4 Pathnames and Genera 7.2

Genera 7.2 supports VAX/VMS Version 4.4 pathnames with hyphens in them. Previously, hyphens in logical pathnames were changed to the underscore character when they were translated from logical pathnames to VAX/VMS physical pathnames. For a general description of the treatment of VAX/VMS pathnames, including how hyphens are handled: See the section "VAX/VMS Pathnames" in *Reference Guide to Streams, Files, and I/O.*

Symbolics recommends that you upgrade your VAX/VMS system to Version 4.4 before you install Genera 7.2 software. This way, Symbolics pathnames and VAX/VMS pathnames will be identical. Hyphens and underscores will be used in the same places on both systems.

In this case, edit the host object so that the host has a system-type of VMS4.4. Then you'll have complete compatibility in pathnames between the two systems.

If, for some reason, you cannot upgrade your VAX/VMS system to Version 4.4 before upgrading your Symbolics system to Genera 7.2, be sure to identify the host object as being a VAX/VMS Version 4.0 through 4.3 server — or a VAX/VMS Version 3 server — in the namespace. Your pathname translations will remain the same, and you won't have Version 4.4 pathname support.

If you choose to upgrade to VAX/VMS Version 4.4 after installing the Genera 7.2 software, either leave the namespace identification the same as it was earlier, unless you plan to go back and rename all the affected files by hand. Alternatively, edit the host object so that the host has a system-type of VMS4, and rename all the affected files by hand. This will allow you to take advantage of VAX/VMS Version 4.4 pathname support.

4.4. Installing the Genera 7.2 World on a 3640 with a 140-Mbyte Disk

If netbooting is possible at your site, Symbolics recommends that you netboot — rather than restore from tape — Genera 7.2 on your 3640. To netboot, you must first perform a complete Genera 7.2 installatation on one (or more) host(s) at your site. More information is available about netbooting. See the section "Netbooting", page 39. This section contains information for sites where netbooting on a 3640 is not possible.

Installation Overview:

Since the 140-Mbyte disk drive on a 3640 has a smaller paging file than does the 3600 or the 3650, a special procedure exists for installing Genera 7.2 on it.

Although your disk has about 110,000 blocks of storage which can be divided into about 40,000 blocks for the world load, and 75,000 blocks for paging space, there is a moment in time when you will need more room than that (you'll need enough for two world loads: the one from which you've booted, and the one that you want to save).

A normal installation procedure requires that your disk be partitioned as follows: about 40,000 blocks for the world load, 45,000 blocks for the main paging area (typically named page.page), and 30,000 blocks for the auxilliary paging area (typically named aux.page). In fact, your 3640 was shipped to you configured (approximately) like this.

In a normal installation, your boot file would be set up to specify both the page.page and aux.page paging areas as virtual memory. However, in order to install Genera 7.2 on your 140-Mbyte disk, you'll need to boot manually, typing each command explicitly to the FEP, and adding the auxilliary paging area when you do so.

(When you boot this way, periodic low-address space warnings will appear. Ignore them; you won't be booted long enough to run out of address space.)

After you've booted with only aux.page specified, you can delete and expunge page.page and temporarily free up disk space with which to save the new world. Later, you'll be able to boot the new world, delete and expunge the old world, and create a new main paging file. Alternatively, you can rename the old world to page.page and use that. Either way, you'll be back to the original (optimal) configuration.

Do not attempt to merge two separate paging areas into one, larger paging area. This configuration is difficult to undo.

Before you Perform the Installation:

Use the Show FEP Directory command (at the Lisp Listener) to ensure that your FEP file system (FEPFS) has been configured with:

- An already-resident world load and microcode files.

- A 45,000 block paging file called page.page.

• A 35,000 block paging file called aux.page.

Installation Steps:

1. Manually boot the already-resident world load and the microcode files, specifying *only* the 30,000 block aux.page. At the FEP, instead of typing Boot, type:

    ```
    Clear Machine
    Load Microcode microcode-file-name
    Load World world-load-file-name
    Clear Paging
    Add Paging-File FEP0:>aux.page
    Start
    ```

    The Load World command automatically loads the 45,000 block page.page paging file, so you must Clear Paging and add *only* the 30,000 block aux.page file.

2. Log in, and delete the FEP0:>page.page file:

    ```
    Delete File FEP0:>page.page
    ```

3. Expunge the FEP0:>page.page file like this:

    ```
    Expunge Directory FEP0:>*.*
    ```

    Now there are 45,000 free blocks.

4. Rename the FEP0:>aux.page file to FEP0:>page.page file like this:

    ```
    Rename File FEP0:>aux.page   FEP0:>page.page
    ```

    Now page.page is a 30,000 block paging file.

5. Insert the first *Genera 7.2 Distribution World* tape (1/3) into the tape drive. Type the CP command Select Activity FEP-Tape or, if you are running Release 6, use **tape:fep-file-restore**.

    If the tape is in another machine's cartridge drive, type

    ```
    Read Tape :host remote-host-name.
    ```

    If the tape is in the local machine's cartridge drive, click on [Read Tape].

6. The FEP-Tape program first asks you whether you want to load each microcode file. The tape contains a complete set of microcodes for Genera 7.2. Before you restore the microcode for your 3640, you'll need to know the appropriate microcode pathname. This depends on the specific configuration of your 3640. More information is available about microcodes.See the section "Genera 7.2 Microcode Types", page 58.

For each microcode type, you are asked whether or not you want to restore
the file. Here is an example of what you see on the screen. In this example,
the microcode is for a 3640 machine with floating-point accelerator:

```
Filename: SYS:L-UCODE;3640-FPA-MIC.MIC.419
Length: 127843 bytes (112 blocks)
Created: 2/05/88 17:28:33
Author: Lowe
Restoration Comment: For microcode 419
Restore this file? (Y or N)
```

If you answer N (for No) the program skips the microcode file and prompts
you for the next version. If you answer Y (for Yes) the program asks you:

```
Into the file (default is FEP0:>3640-FPA-MIC.MIC.419)
```

When you respond to this question, accept the default pathname, or specify a
FEP pathname that indicates microcode, such as:

```
FEP0:>3640-fpa-mic.mic.419
```

After the microcode files are presented, and you have restored the appropriate
one, 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-7-2.load
```

The FEP-Tape loading program then loads the world from the Genera 7.2
Distribution World tape set, prompting you to insert a new tape when the end
of each tape has been reached.

After the world has been restored, you are asked if you want to restore the
netboot core. If you want to use netbooting, answer Y (for Yes) and accept the
default pathname or specify a FEP pathname that indicates a netboot load,
such as

```
FEP0:>Netboot-Core-from-7-2.load
```

For more information about netbooting: See the section "Netbooting", page 39.

7. Boot the new Genera 7.2 world. At the prompt, type:

    ```
    Clear Machine
    Load Microcode microcode-file-name
    Load World world-load-file-name
    Set Chaos-Address Chaosnet-address
    Start
    ```

    This sequence loads the page.page file (this version of the page.page file has
    30,000 blocks of paging space).

8. Once the Genera 7.2 world is up and running, you can delete and expunge the old world, like this:

    ```
    Delete File FEP0:>release-7-1.load
    Expunge Directory FEP0:>*.*
    ```

9. Rename the 30,000 block page.page paging file back to aux.page:

    ```
    Rename File FEP0:>page.page FEP0:>aux.page
    ```

10. Create a new 45,000-block paging file with the space retrieved, like this:

    ```
    Create FEP File FEP0:>page.page 45000
    ```

    You are now returned to the original configuration of a 45,000 block main paging area (page.page) and a 30,000 block auxilliary paging area (aux.page).

11. Edit your boot file by adding this command:

    ```
    Declare Paging-Files FEP0:>page aux
    ```

    Now, your boot file should look something like this:

    ```
    Clear Machine
    Load Microcode microcode-file-name
    Declare Paging-Files FEP0:>page aux
    Load World world-load-file-name
    Set Chaos-Address Chaosnet-address
    Start
    ```

    Both paging files will be loaded automatically when you next use the Boot command at the FEP command prompt. The Declare Paging-Files command specifies the paging file(s) for the Load World command. More information is available about the Declare Paging-Files command. See the document *Site Operations*.

5. Genera 7.2 Distribution Tapes

Genera 7.2 includes six tapes as described below:

1. Genera 7.2 Distribution World tapes. These contain microcodes and the Genera 7.2 world load:

    *   Symbolics Genera 7.2 Distribution World 1/3 IFS Format P/N: 995446

    *   Symbolics Genera 7.2 Distribution World 2/3 IFS Format P/N: 995446

    *   Symbolics Genera 7.2 Distribution World 3/3 IFS Format P/N: 995446

2. Genera 7.2 Basic Distribution tapes. These contain system sources, examples, fonts, and the documentation database:

    *   Symbolics Genera 7.2 Basic Distribution 1/3 Distrib. Format P/N: 995447

    *   Symbolics Genera 7.2 Basic Distribution 2/3 Distrib.Format P/N: 995447

    *   Symbolics Genera 7.2 Basic Distribution 3/3 Distrib.Format P/N: 995447

6. Genera 7.2 Microcode Types

Once you have restored microcode from the *Genera 7.2 Distribution World* tape, you can then distribute microcode versions from machine to machine. To do this, use the Copy Microcode command.

Microcode names consist of the machine model number and the designation MIC. They may also include the designation FPA, if the machine has a floating-point accelerator. In addition, some microcode names include the designation IFU, for the instruction fetch unit, or XSQ, for the extended sequencer that Prolog requires. This varies by model number.

Microcode names no longer include *ST506*, which signified a type of disk interface.

The standard microcode for a 3640 is:

    3640-MIC.MIC.419

If the 3640 has the floating-point accelerator, the microcode required is:

    3640-FPA-MIC.MIC.419

If the 3640 also has the extended sequencer required to run Prolog, the microcode required is:

    3640-FPA-XSQ-MIC.MIC.419

Use the CP command Show Herald to determine your model number. Use the CP command Show Machine Configuration (or **(si:show-configuration)** on a Release 6.1 system) to determine the presence (or absence) of the FPA, XSQ, or IFU.

The 3645 and 3675 always have an IFU. The presence of an IFU implies XSQ functionality. This means the standard microcode

    3645-MIC.MIC.419

includes IFU and the XSQ support.

Some 3600s have an IFU (and therefore, implied XSQ functionality). Microcode names for these machines include the designation IFU, but do not specifically designate the XSQ.

In addition, the 3610, 3620, and 3650 have an NBS processor, which implies both IFU and XSQ functionality. Microcode names for these machines do not specifically designate the IFU or XSQ.

Here is a list of the microcode pathnames for each machine model. If you have a 3600 machine model: See the section "3600 Machine Model Microcodes", page 60. Otherwise, read this list.

| 3610 AE | 3610-MIC.MIC.419 |
| 3610 AE, FPA | 3610-FPA-MIC.MIC.419 |

3620 model:

| 3620 | 3620-MIC.MIC.419 |
| 3620, FPA | 3620-FPA-MIC.MIC.419 |

3630 model:

| 3630 | 3630-MIC.MIC.419 |
| 3630, FPA | 3630-FPA-MIC.MIC.419 |

3640 model:

3640	3640-MIC.MIC.419
3640, FPA	3640-FPA-MIC.MIC.419
3640, PROLOG	3640-XSQ-MIC.MIC.419

3640, FPA, PROLOG
        3640-FPA-XSQ-MIC.MIC.419

3645 model:

| 3645 | 3645-MIC.MIC.419 |
| 3645, FPA | 3645-FPA-MIC.MIC.419 |

3650 model:

| 3650 | 3650-MIC.MIC.419 |
| 3650, FPA | 3650-FPA-MIC.MIC.419 |

3653 model:

| 3653 | 3653-MIC.MIC.419 |
| 3653, FPA | 3653-FPA-MIC.MIC.419 |

3670 model:

3670	3670-MIC.MIC.419
3670, FPA	3670-FPA-MIC.MIC.419
3670, PROLOG	3670-XSQ-MIC.MIC.419

3670, FPA, PROLOG
        3670-FPA-XSQ-MIC.MIC.419

3675 model:

3675   3675-MIC.MIC.419

3675, FPA   3675-FPA-MIC.MIC.419

6.1. 3600 Machine Model Microcodes

If your machine is a 3600, use the CP command Show Machine Configuration (or **(si:show-configuration)** on a Release 6.1 system) to see your machine's IO REV. Then follow the chart below to determine what microcode version to use for your machine.

If your 3600 has IO REV 2, use this chart of microcode pathnames to determine what microcode you should use:

3600, IO REV 2   3600-MIC.MIC.419

3600, IO REV 2, PROLOG
                          3600-XSQ-MIC.MIC.419

3600, IO REV 2, IFU
                          3600-IFU-MIC.MIC.419

3600, IO REV 2, FPA
                          3600-FPA-MIC.MIC.419

3600, IO REV 2, PROLOG, FPA
                          3600-FPA-XSQ-MIC.MIC.419

3600, IO REV 2, IFU, FPA
                          3600-FPA-IFU-MIC.MIC.419

If your 3600 has IO REV 6, use this chart of microcode pathnames to determine what microcode you should use:

3600, IO REV 6   3670-MIC.MIC.419

3600, IO REV 6, PROLOG
                          3670-XSQ-MIC.MIC.419

3600, IO REV 6, IFU
                          3675-MIC.MIC.419

3600, IO REV 6, FPA
                          3670-FPA-MIC.MIC.419

3600, IO REV 6, PROLOG, FPA
                          3670-FPA-XSQ-MIC.MIC.419

3600, IO REV 6, IFU, FPA
                          3675-FPA-MIC.MIC.419

Index

C

C

C

D

D

D

Restoring the Genera 7.2 Documentation, Sources, Examples, Fonts, and Non-Loaded Systems 35

Storing Documentation, Sources, Examples, Fonts, and Non-Loaded Systems on a Non-Symbolics Machine 50

document examiner 6

E · E · E

Enable IDS command 13

document examiner 6

Examining the FEP File System 8

Examining the Lisp Machine File System 10

Restoring the Genera 7.2 Documentation, Sources, Examples, Fonts, and Non-Loaded Systems 35

Storing Documentation, Sources, Examples, Fonts, and Non-Loaded Systems on a Non-Symbolics Machine 50

F · F · F

FEP command 44

FEP Command Summary 13

Show FEP Directory, Command 8

FEP files 7

Examining the FEP File System 8

FEPFS 7

Creating a World-Build Script File 18

Updating the SYS.TRANSLATIONS File 32

FEP files 7

Managing System Source Files When Running Multiple Releases 34

Examining the FEP File System 8

Examining the Lisp Machine File System 10

Lisp Machine File System 7

File Systems on a Symbolics Computer Disk 7

Restoring the Genera 7.2 Documentation, Sources, Examples, Fonts, and Non-Loaded Systems 35

Storing Documentation, Sources, Examples, Fonts, and Non-Loaded Systems on a Non-Symbolics Machine 50

front-end processor 11

G · G · G

VAX/VMS Version 4.4 Pathnames and Genera 7.2 52

Genera 7.2 Distribution Tapes 57

Restoring the Genera 7.2 Documentation, Sources, Examples, Fonts, and Non-Loaded Systems 35

Genera 7.2 Microcode Types 58

Making a Genera 7.2 Site-Configured World 31

Installing the Genera 7.2 Software 28

Overview for Installing the Genera 7.2 Software 28

Making Customized Genera 7.2 User and Server Worlds 37

Restoring the Genera 7.2 World, Netboot Core, and Microcode 29

Installing the Genera 7.2 World on a 3640 with a 140-Mbyte Disk 53

Glossary of Terms for Site Configuration 22

S   S   S

Z
Z
Z