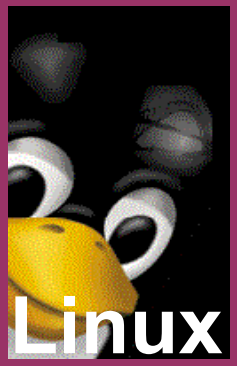
hp
invent

technical computing – invent. design. deliver.



Introduction to Linux for HP-UX System Administrators

HP-ux11

Rob Lucke
Hewlett-Packard
Technical Computing Advanced Technology Center
Rob_Lucke@hp.com

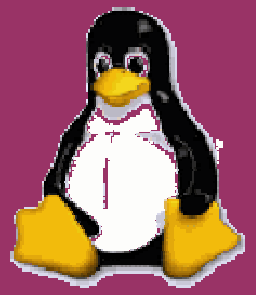
invent

technical computing – invent. design. deliver.

Seminar Agenda

- Introduction
- Disk Partitioning Tutorial
- System Installation
- Boot Managers – LILO and GRUB
- Booting and Startup
- Hardware Configuration and Troubleshooting
- File System Layout
- Software Installation and Update
- System Administration Tasks
- Building The Linux Kernel and Modules
- Dynamic Kernel Moduels
- Configuring Your Sound Card
- Configuring X-Windows
- Miscellaneous Redhat Configuration Tools
- System Installation with Kickstart
- Miscellaneous Linux Cool Stuff

HP-UX: UNIX for the Enterprise



technical computing – invent. design. deliver.

Warning!

This seminar is intended for HP-UX administrators that have little or no experience with Linux. If you have just spent a week in Linux boot-camp, then the contents of this seminar will be too basic for you.

technical computing – invent. design. deliver.

Introduction

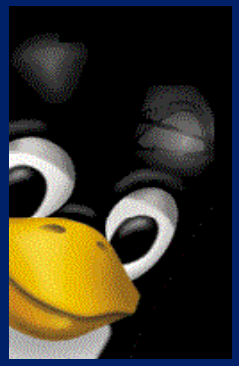
What is Linux?

What is a Linux Distribution?

Linux Versioning

What is Open Source?

Getting Help and Software



technical computing – invent. design. deliver.

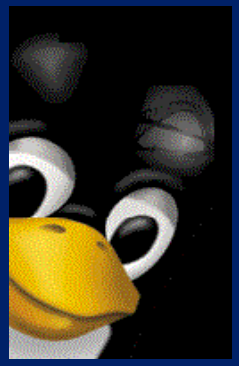
What is Linux?

- When we say “Linux”, we are really talking about three things:
  - An open source kernel, written by Linus Torvalds (Linux)
  - A set of open source tools from the “Free Software Foundation”, the GNU tools
  - A system “structure”, philosphy, and installation tool(s)
- Linux is Unix, like HP-UX, but has differences that we need to consider as HP-UX system administrators
- This is not an exhaustive tutorial, just some of the things that I have run into during my exposure to Linux, with an HP-UX administration slant
- This seminar is based on the RedHat Linux 7.2 and 7.3 *distribution*, there are many others (SuSE, Debian, …)

technical computing – invent. design. deliver.

What is a Linux Distribution?

- A particular version of the kernel
  - RedHat 7.3 is based on the 2.4.18-3 kernel
- A particular set of packages and a package manager
  - May be in Redhat Package Manager (RPM) format
  - May be in Debian (Deb) format
  - May be tar-balls or source packages
- A set of installation tools
  - Disk partitioning
  - System installation
- “Value Added”
  - Update
  - Support
  - Documentation



technical computing – invent. design. deliver.

Explaining Linux Versioning

- The Linux kernel and most packages follow a similar versioning scheme
- Kernel Version = MajorRelease.MinorRelease.Step
  - Odd minor-release kernels are “development”
  - Even minor-release kernels are “stable”
  - Minor release numbers are incremented with patches
  - Example “2.4.18-3” is a stable kernel
- Packages use a similar scheme (more later on this)
  - red-carpet-1.3.3-4.ximian.1.i386.rpm
  - gdb-5.2-2.i386.rpm
  - ethereal-0.9.4-0.7.3.0.i386.rpm

technical computing – invent. design. deliver.

What is Open Source?

- Software protected by the GNU General Public License (GPL) or similar license schemes
- There are multiple versions of the GPL and other “open-source” license schemes
- From /usr/src/linux2.4/Documentation/COPYING:

“Also note that the only valid version of the GPL as far as the kernel is concerned is _this_ particular version of the license (ie v2, not v2.2 or v3.x or whatever), unless explicitly otherwise stated.”

-- Linus Torvalds

technical computing – invent. design. deliver.

What is Open Source?

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it. For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software. […]

from: /usr/src/linux2.4/Documentation/COPYING © 1989, 1991 Free Software Foundation, Inc.

technical computing – invent. design. deliver.

Getting Help and Software?

- Open Source software is available from a *lot* of places. I tend to frequent:
  - http://www.rpmfind.net
  - http://www.sourceforge.net
- Documentation and HOWTOs can tend to be out of date. The Redhat site is good for Redhat specific documentation and has pointers to other sites
- There are lots of news groups and chat channels devoted to all aspects of Linux and GNU applications
- There are formal support channels available for Linux: HP, Redhat, etc.

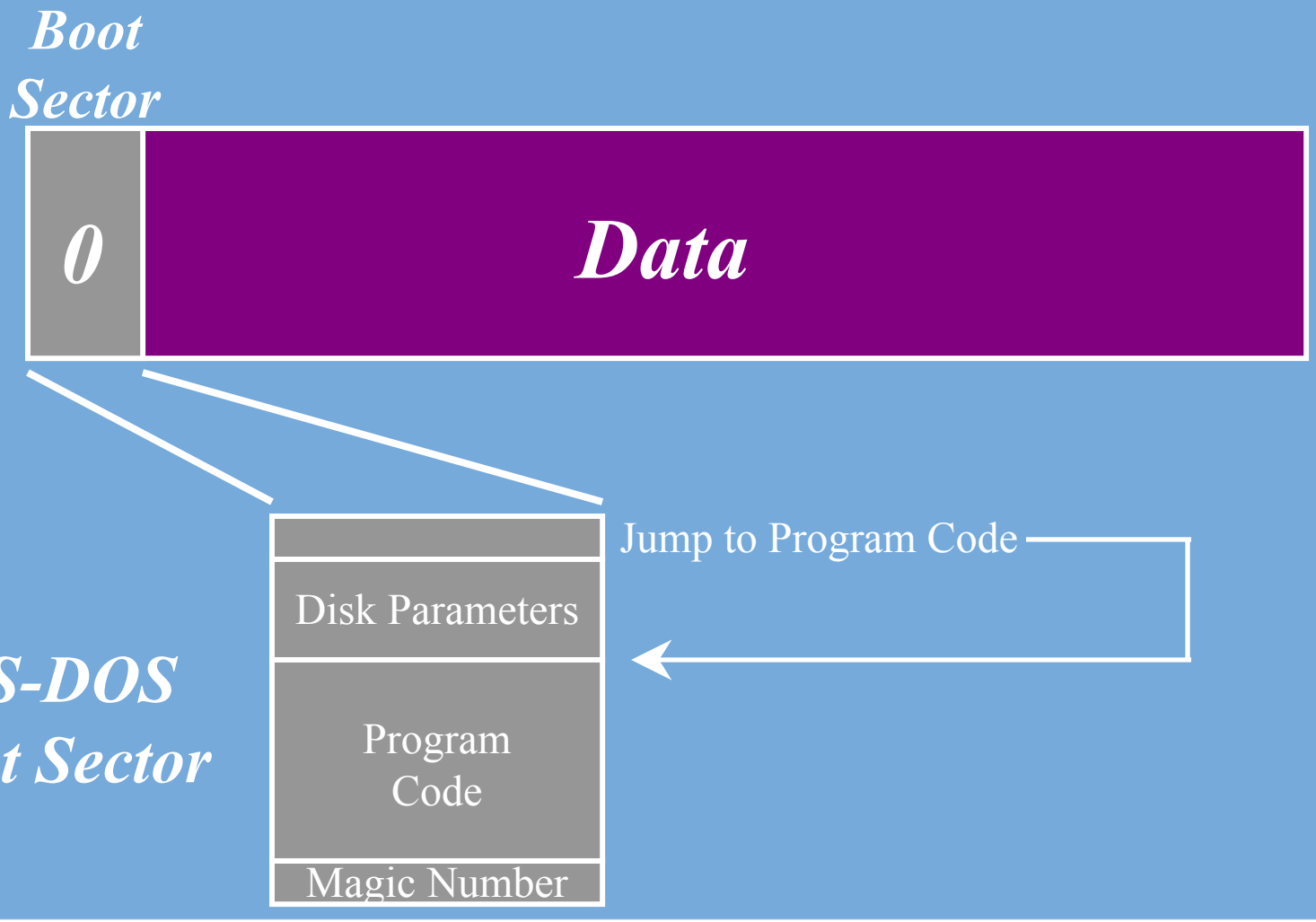
technical computing – invent. design. deliver.

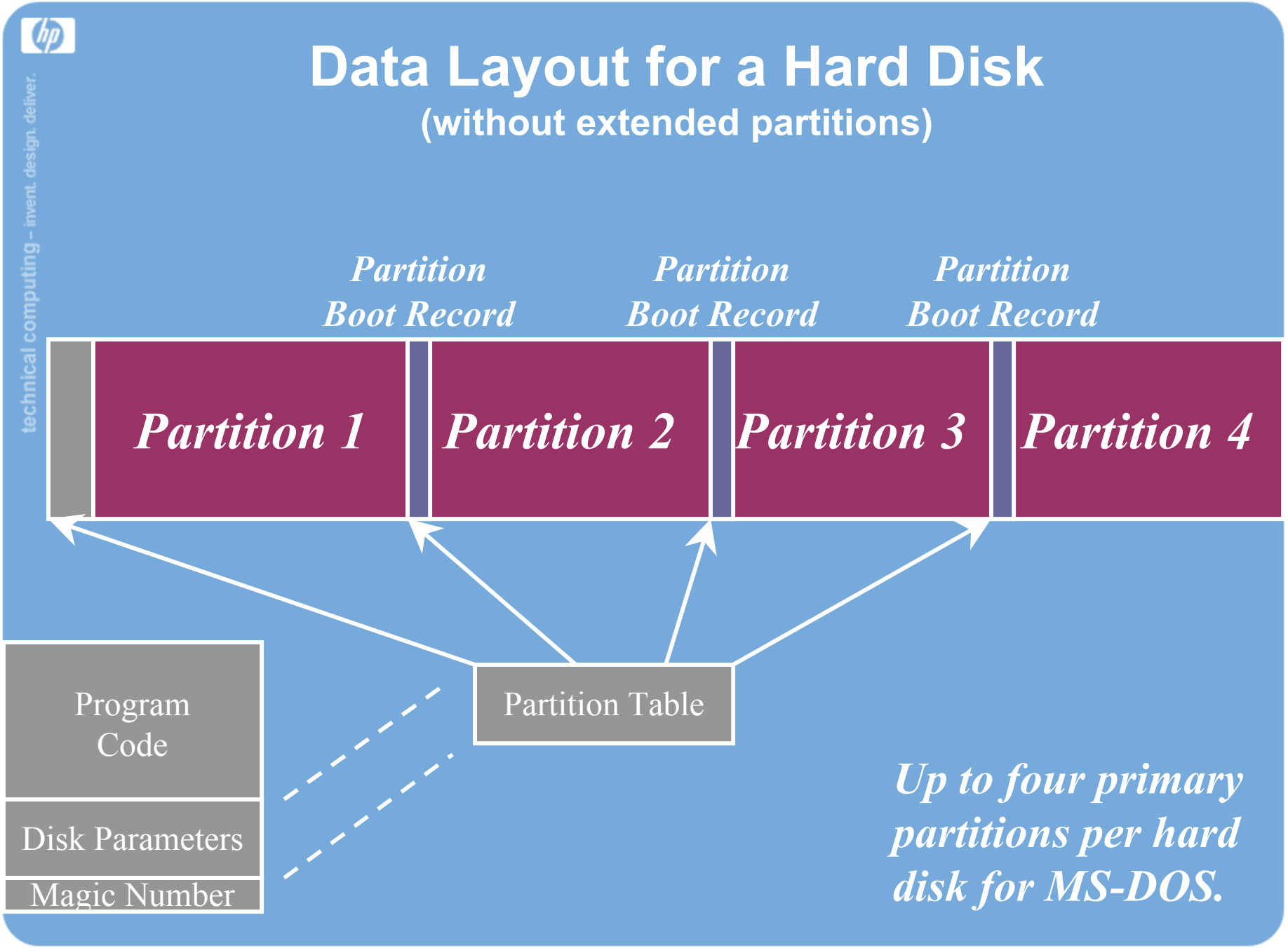
Disk Partitioning Tutorial

Floppy Disk Format
The Master Boot Record (MBR)
Partition Layout for a Hard Disk
Devices for Primary Partitions
Devices for Extended Partitions

technical computing – invent. design. deliver.

Data Layout for a Floppy Disk

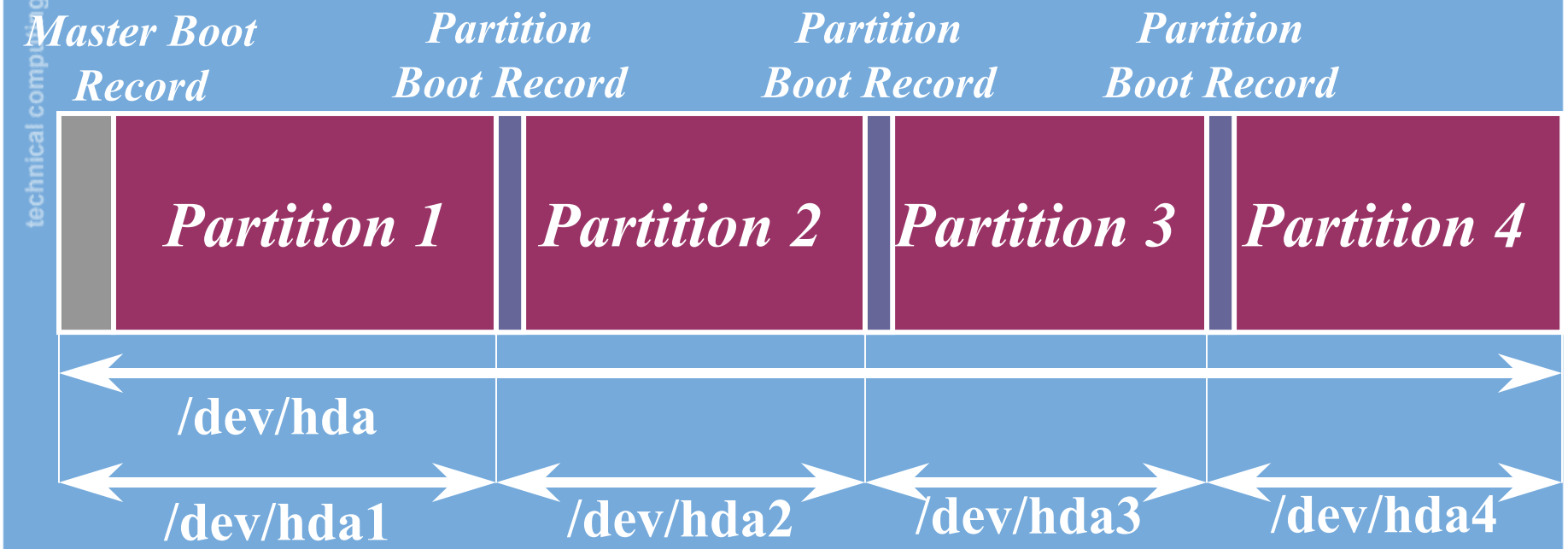



technical computing – invent. design. deliver.
Data Layout for a Hard Disk
(without extended partitions)
Partition Boot Record
Partition Boot Record
Partition Boot Record
Partition 1
Partition 2
Partition 3
Partition 4
Program Code
Disk Parameters
Magic Number
Partition Table
Up to four primary partitions per hard disk for MS-DOS.

technical computing – invent. design. deliver.

Data Layout for an IDE Hard Disk

(without extended partitions)

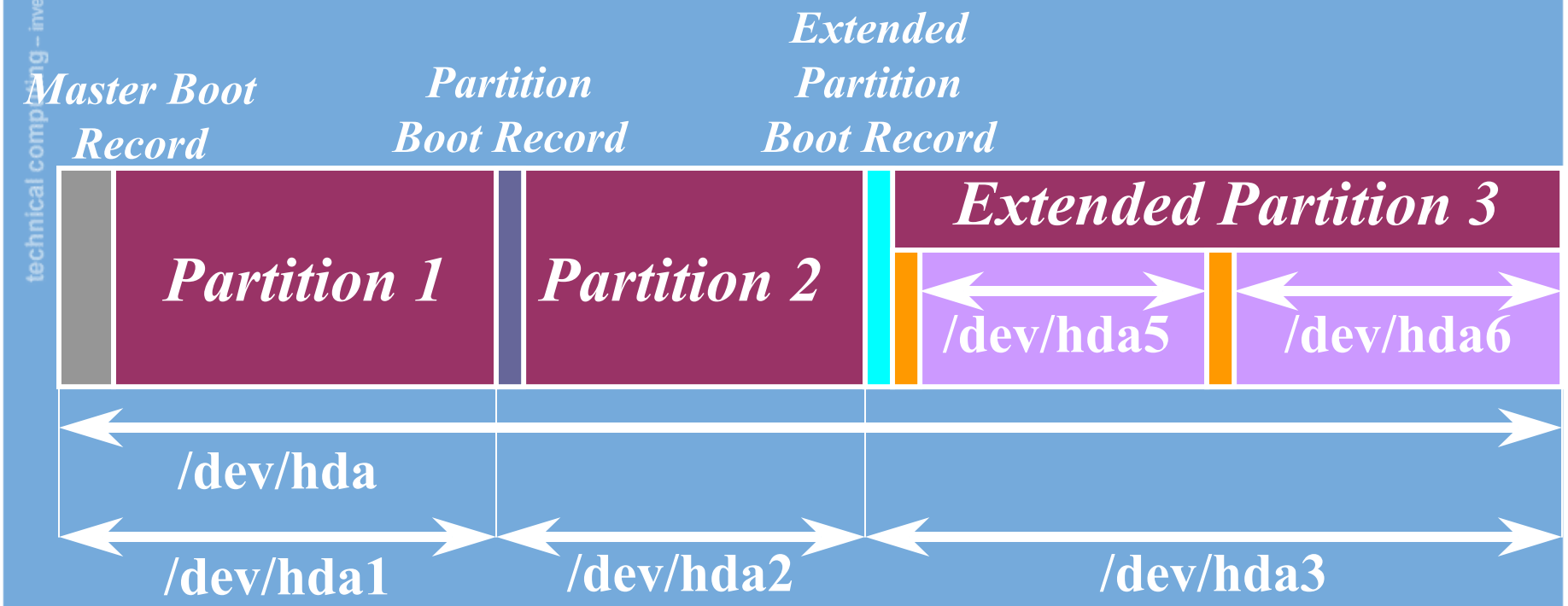


Note that each partition is accessible as if it were an independent device with its own boot sector. Linux creates devices for this configuration as shown.

technical computing – invent. design. deliver.

Data Layout for an IDE Hard Disk

(with extended partitions)



If more than four partitions are needed, one primary partition is divided into an extended partition containing several logical partitions. Note that the partition tables of the logical partitions are not accessible as the first block of some device.

technical computing – invent. design. deliver.

Partition Table from My Laptop

fdisk /dev/hda

The number of cylinders for this disk is set to 1559.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other Oss
(e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p

Disk /dev/hda: 240 heads, 63 sectors, 1559 cylinders
Units = cylinders of 15120 * 512 bytes

```
Device Boot      Start         End      Blocks   Id  System
/dev/hda1   *        1           5       37768+  83  Linux
/dev/hda2            6         109      786240  82  Linux swap
/dev/hda3          110        1559    10962000  83  Linux
```

technical computing – invent. design. deliver.

Lessons Learned About Multi-Boot Systems

- If you have an NTFS file system, then it must be in the first partition on the disk (example order: Windows NT/XP, Windows 98, /boot, FAT16)
- All bootable partitions must be within 1023 cylinders of the start of the drive
- Windows will only boot if within the first 4 GB of the disk
- There can be only one active/visible partition unless you use a boot manager like BootMagic
- Bootable partitions must be on IDE disk 0 or SCSI disk 0
- Only four primary partitions per disk
- NT does not know how to read FAT32 file systems, XP does
- If your first drive is SCSI or RAID, Linux may not have the drivers built into the kernel, so you MUST configure an initial RAM disk image containing the dynamically loadable kernel modules (more later)

technical computing – invent. design. deliver.

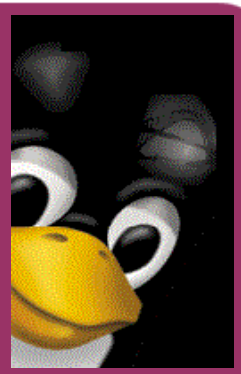
Redhat Linux System Installation

- Booting the Install Media
- Partitioning the Disk
- Installing the Boot Loader
- Configuring the Network
- Configuring the Firewall Options
- Picking Packages
- Installing Packages
- Video Configuration
- Boot Disk Creation
- X-Windows Configuration
- Reboot

technical computing – invent. design. deliver.

System Installation

- HP-UX
  - CD-ROM or dvd
  - Ignite-UX (network)
- Software Distribution Utilities (SDU)
- Graphical or text-based installation tool
- Three main phases:
  - Configure disk layout
  - Install system filesets
  - Configure subsystems



- Redhat Linux
  - CD-ROM or floppy
  - Kickstart (network)
- Redhat Package Manager (RPM)
- Graphical or text-based (VGA) installation tool
- Three main phases:
  - Partition the disk
  - Install system packages
  - Configure subsystems

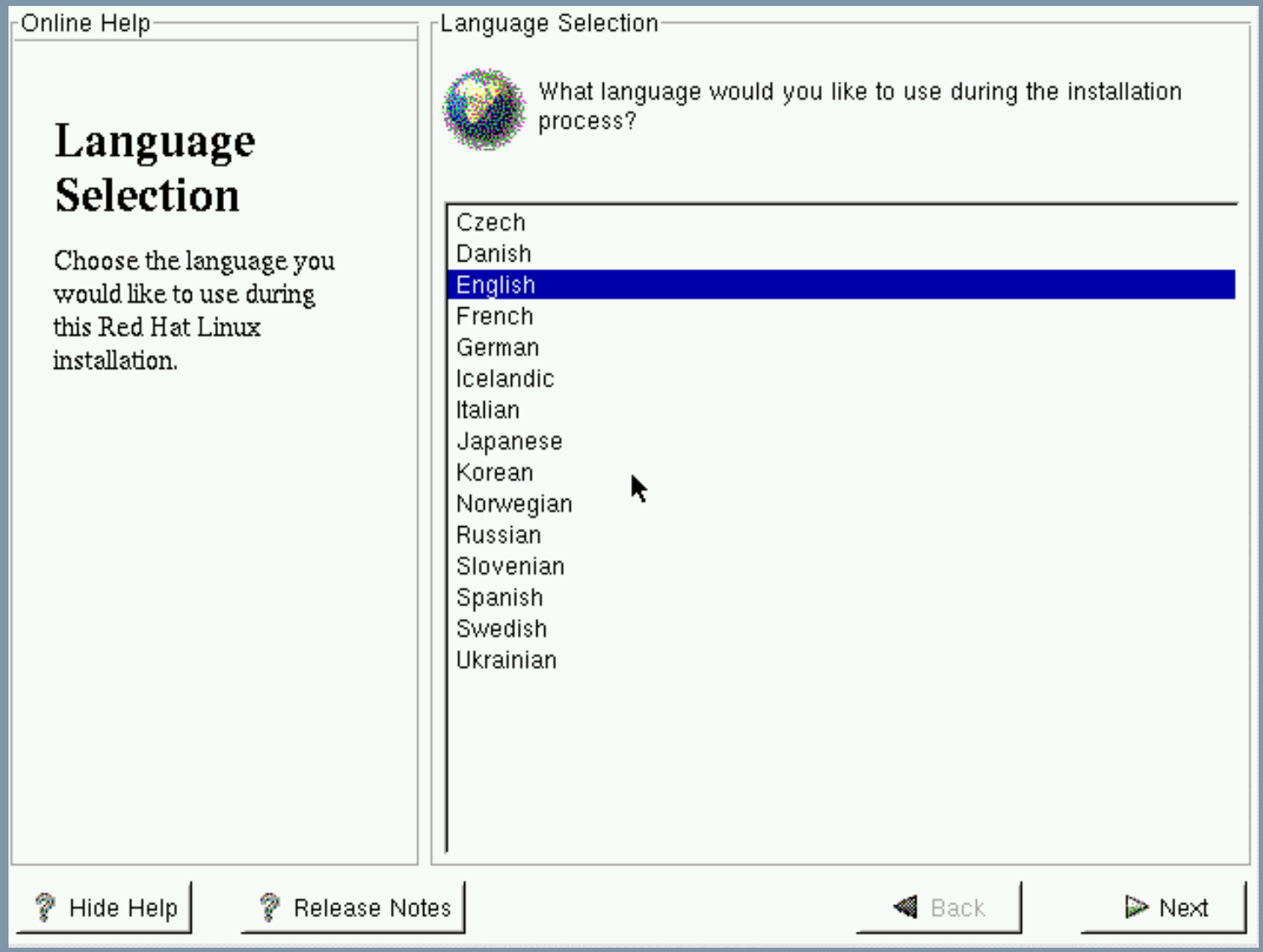
technical computing – invent. design. deliver.

Booting the Installation Media

```
                Welcome to Red Hat Linux 7.2!

 -  To install or upgrade Red Hat Linux in graphical mode,
    press the <ENTER> key.

 -  To install or upgrade Red Hat Linux in text mode, type: text <ENTER>.

 -  To enable low resolution mode, type: lowres <ENTER>.
    Press <F2> for more information about low resolution mode.

 -  To disable framebuffer mode, type: nofb <ENTER>.
    Press <F2> for more information about disabling framebuffer mode.

 -  To enable expert mode, type: expert <ENTER>.
    Press <F3> for more information about expert mode.

 -  To enable rescue mode, type: linux rescue <ENTER>.
    Press <F5> for more information about rescue mode.

 -  If you have a driver disk, type: linux dd <ENTER>.

 -  Use the function keys listed below for more information.

[F1-Main] [F2-General] [F3-Expert] [F4-Kernel] [F5-Rescue]
boot: _
```

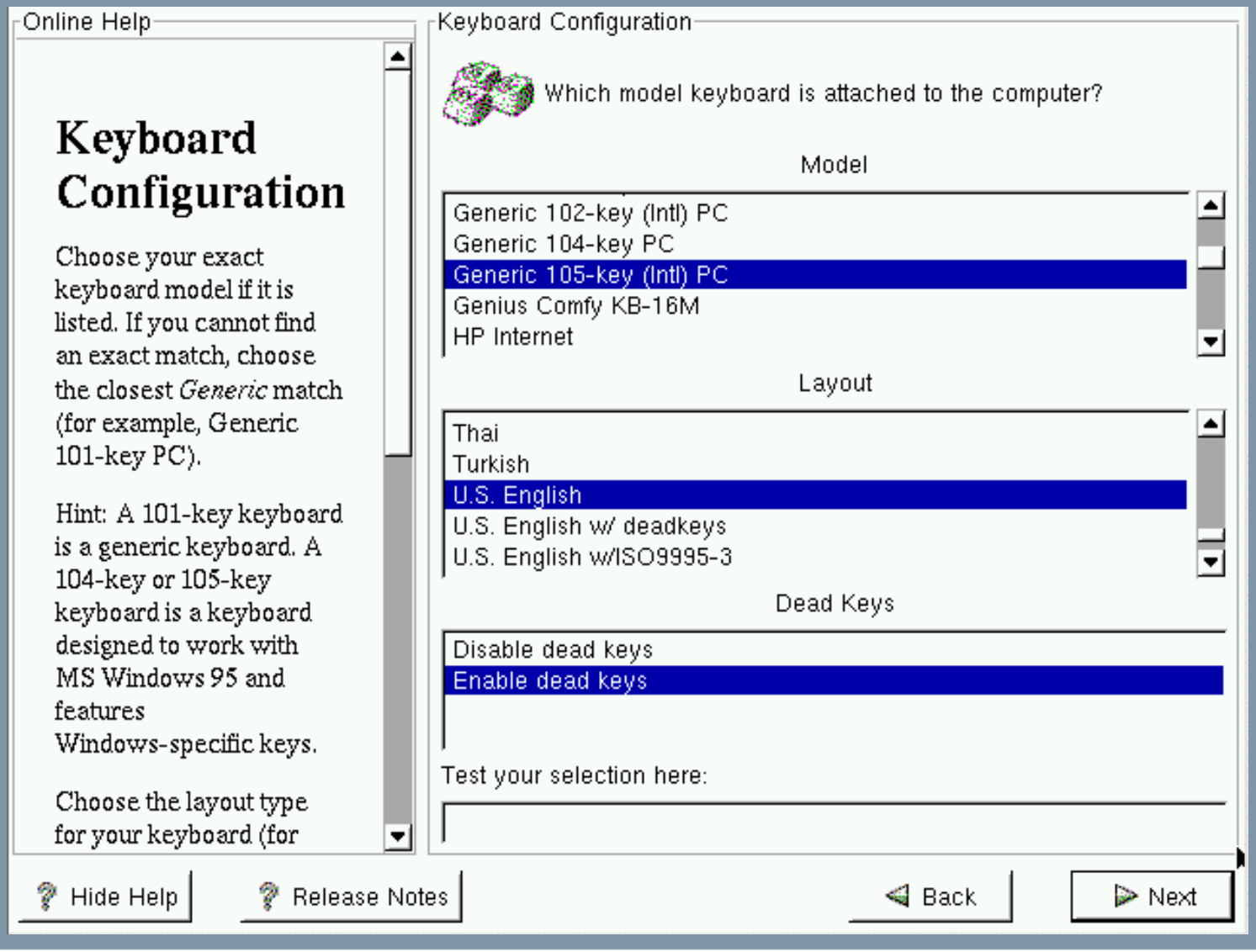
technical computing – invent. design. deliver.

Selecting the Default Language



technical computing – invent. design. deliver.

Configuring the Keyboard



Keyboard Configuration

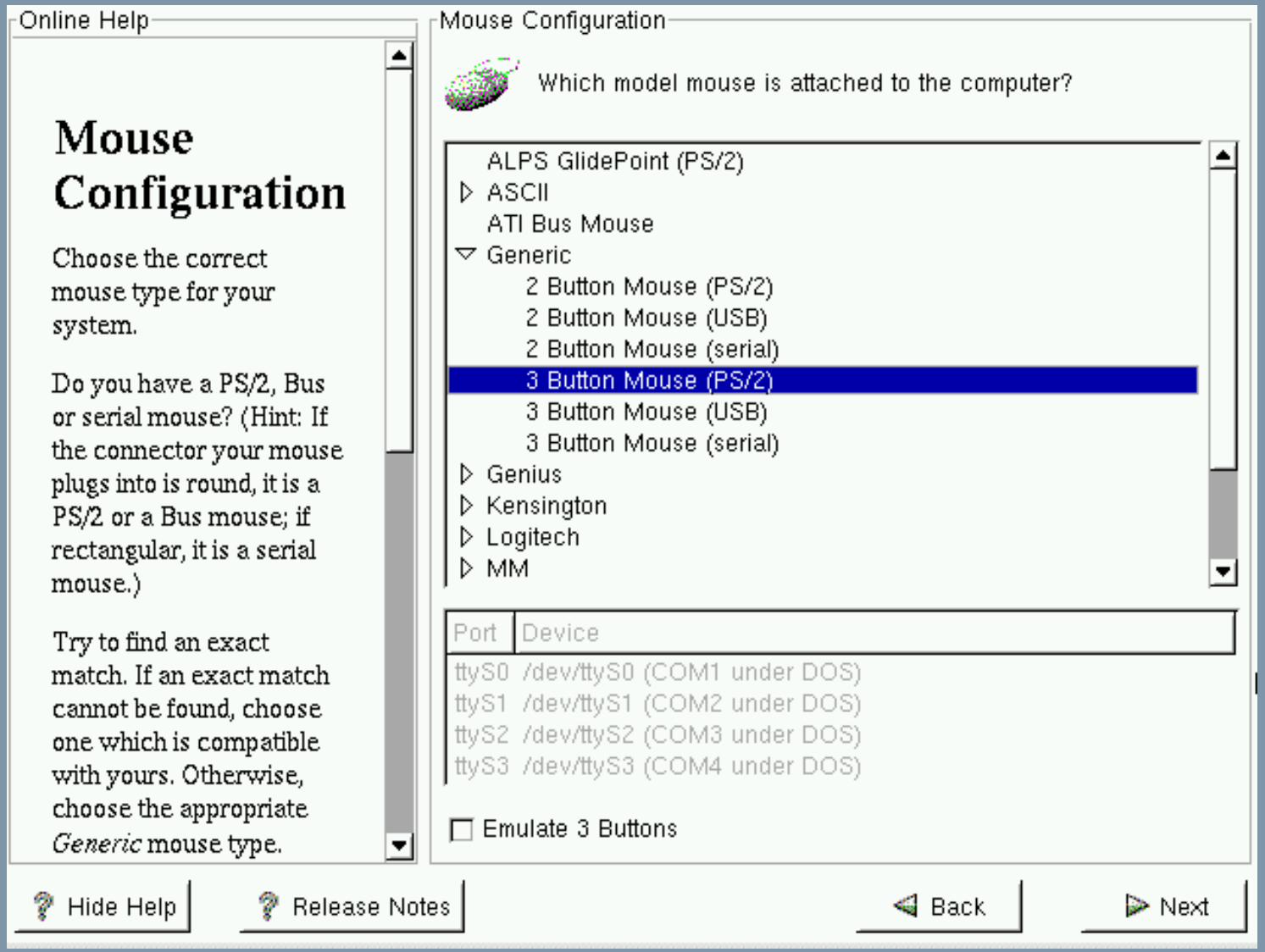
Choose your exact keyboard model if it is listed. If you cannot find an exact match, choose the closest *Generic* match (for example, Generic 101-key PC).

Hint: A 101-key keyboard is a generic keyboard. A 104-key or 105-key keyboard is a keyboard designed to work with MS Windows 95 and features Windows-specific keys.

Choose the layout type for your keyboard (for

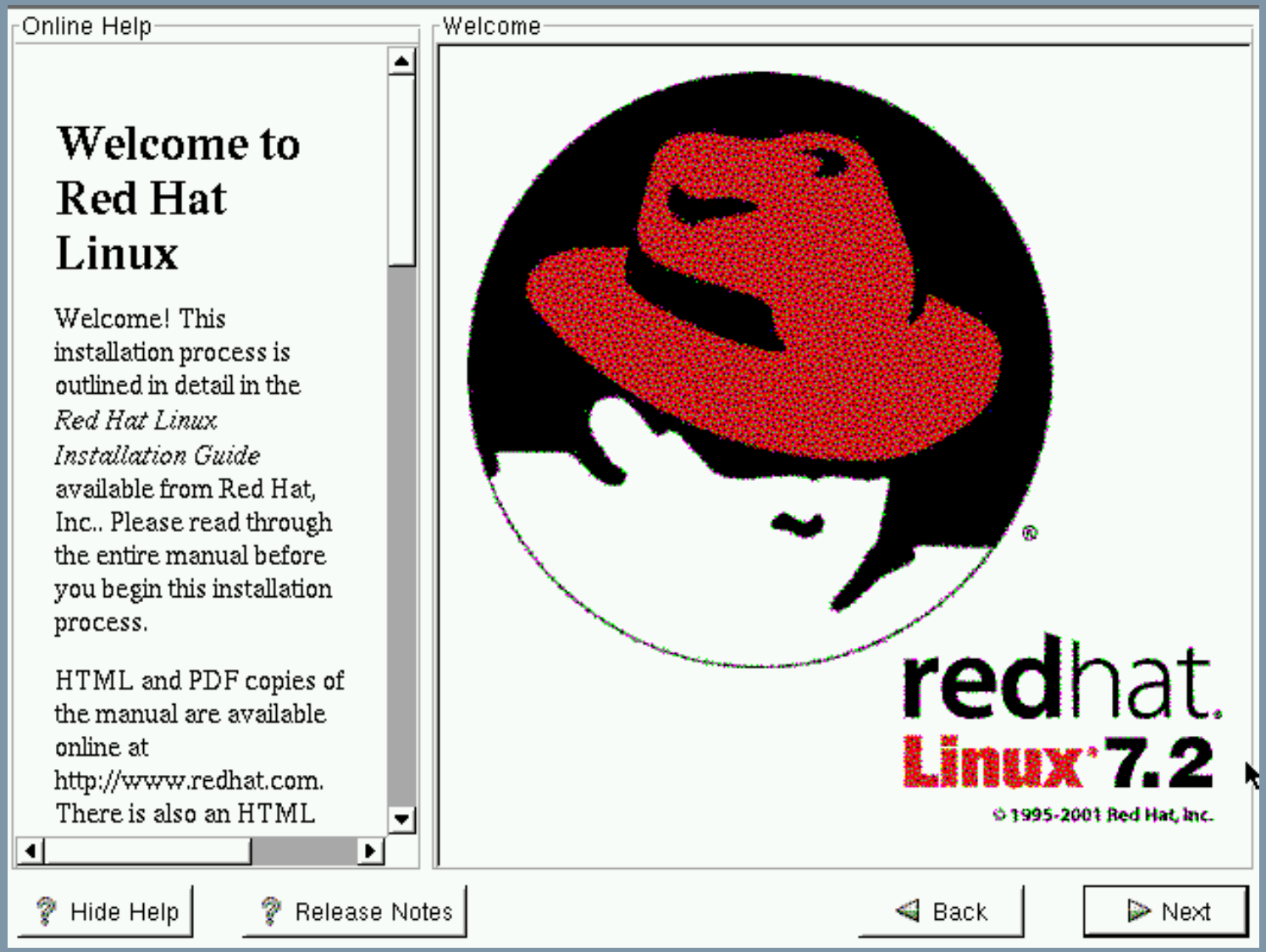
technical computing – invent. design. deliver.

Configuring the Mouse



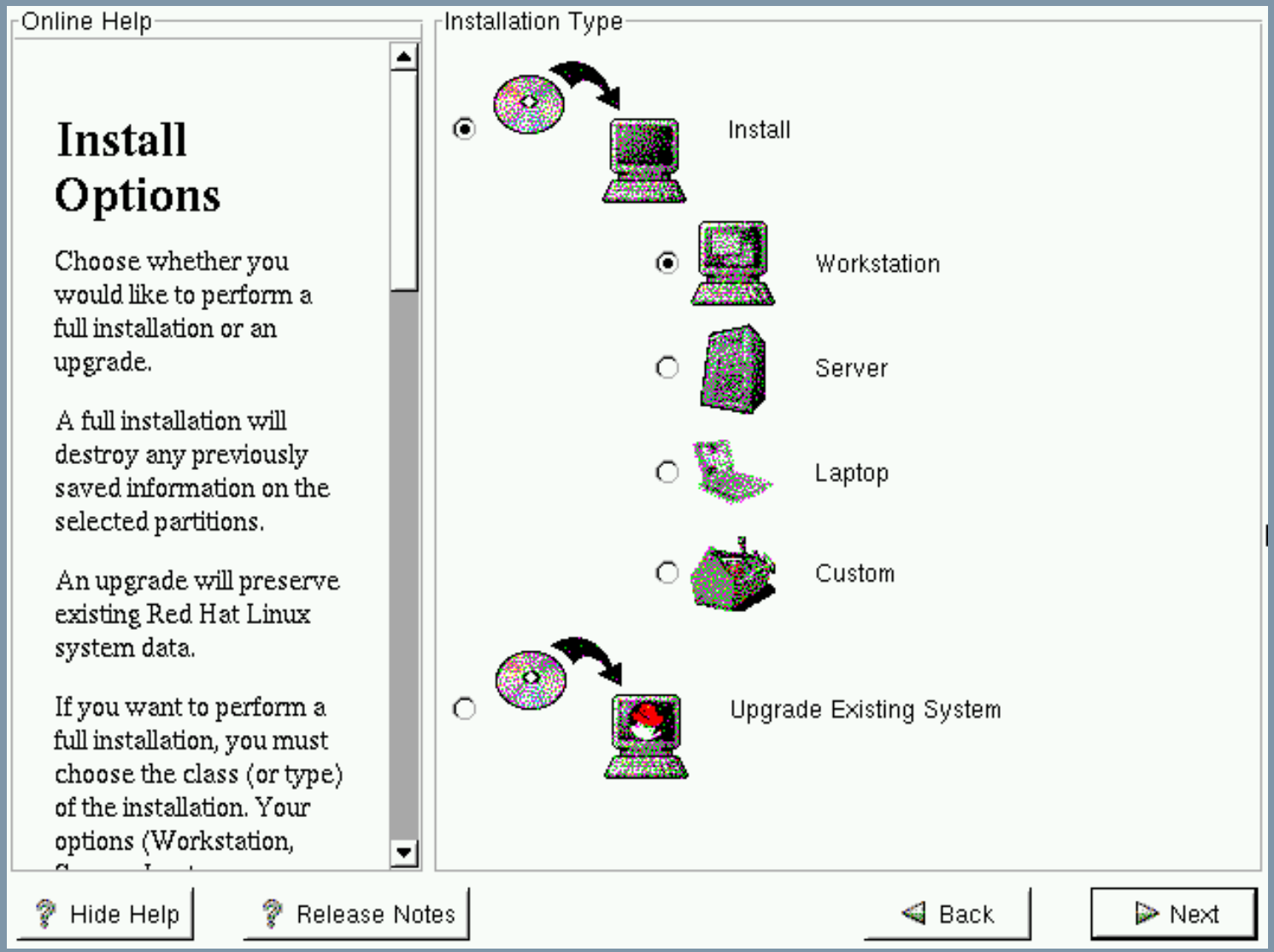
technical computing – invent. design. deliver.

Welcome to Redhat Installation

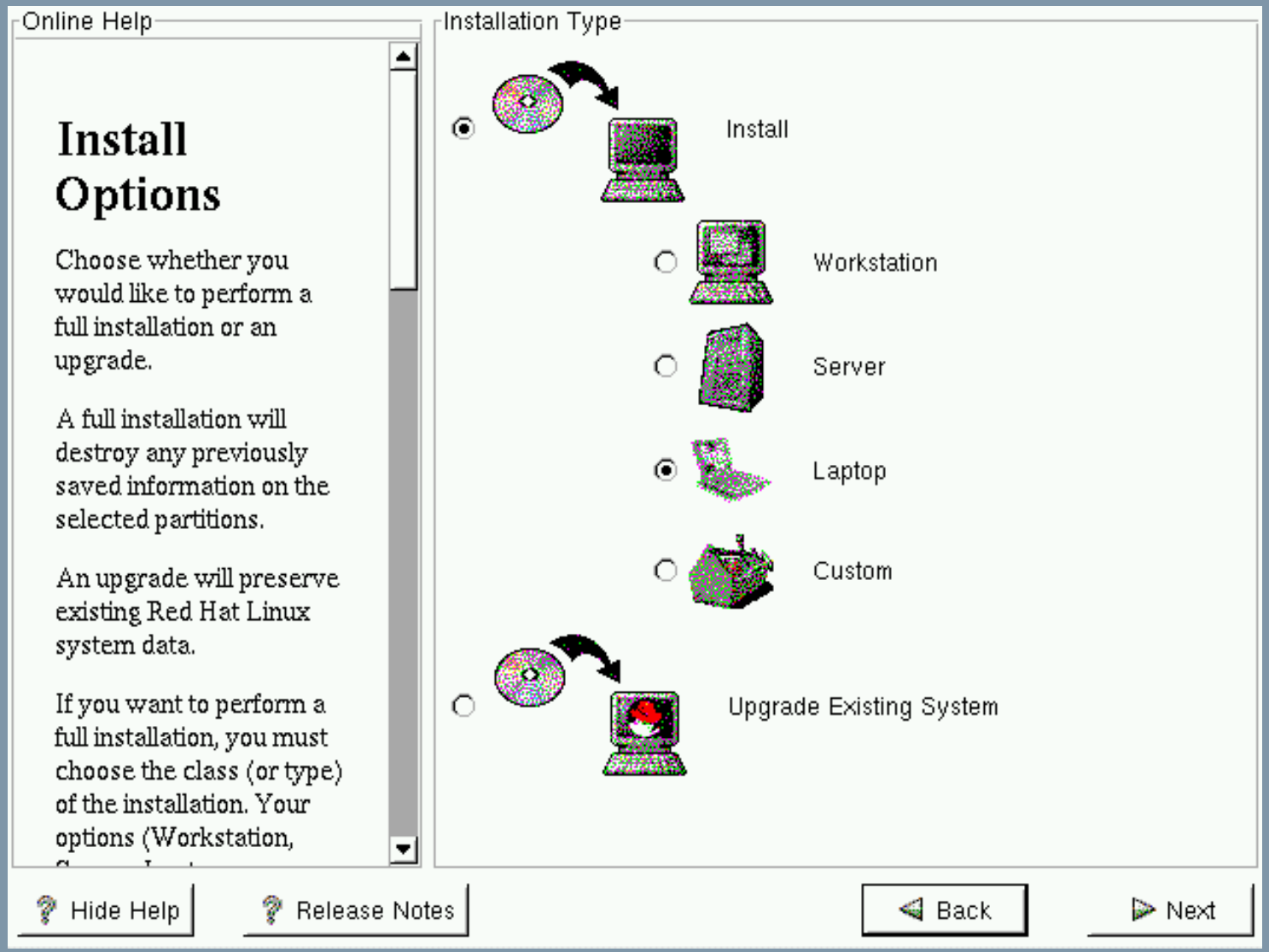


technical computing – invent. design. deliver.

Selecting Installation Type



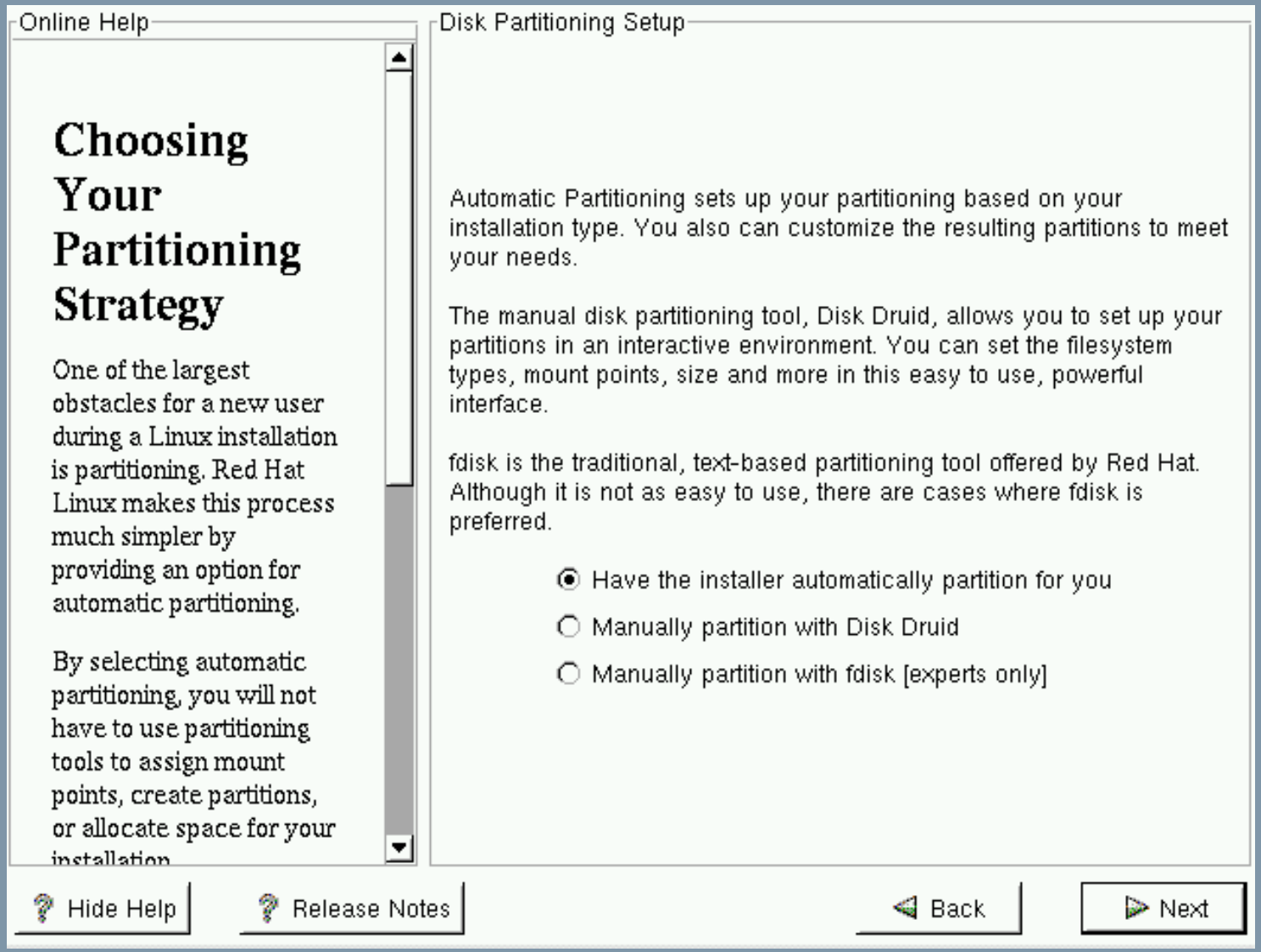
Selecting a Laptop Installation



technical computing – invent. design. deliver.

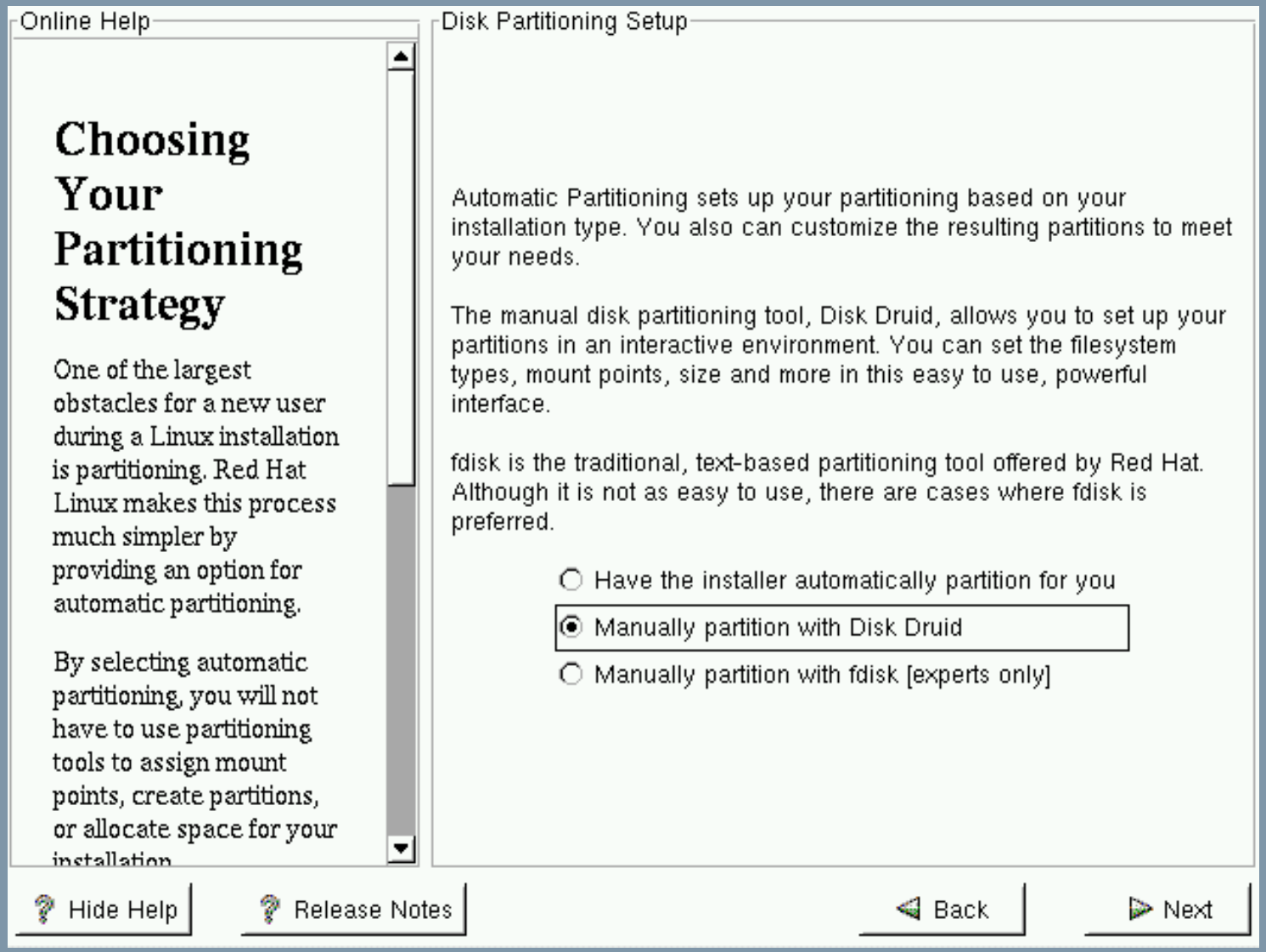
technical computing – invent. design. deliver.

Disk Partitioning: Choose Tool



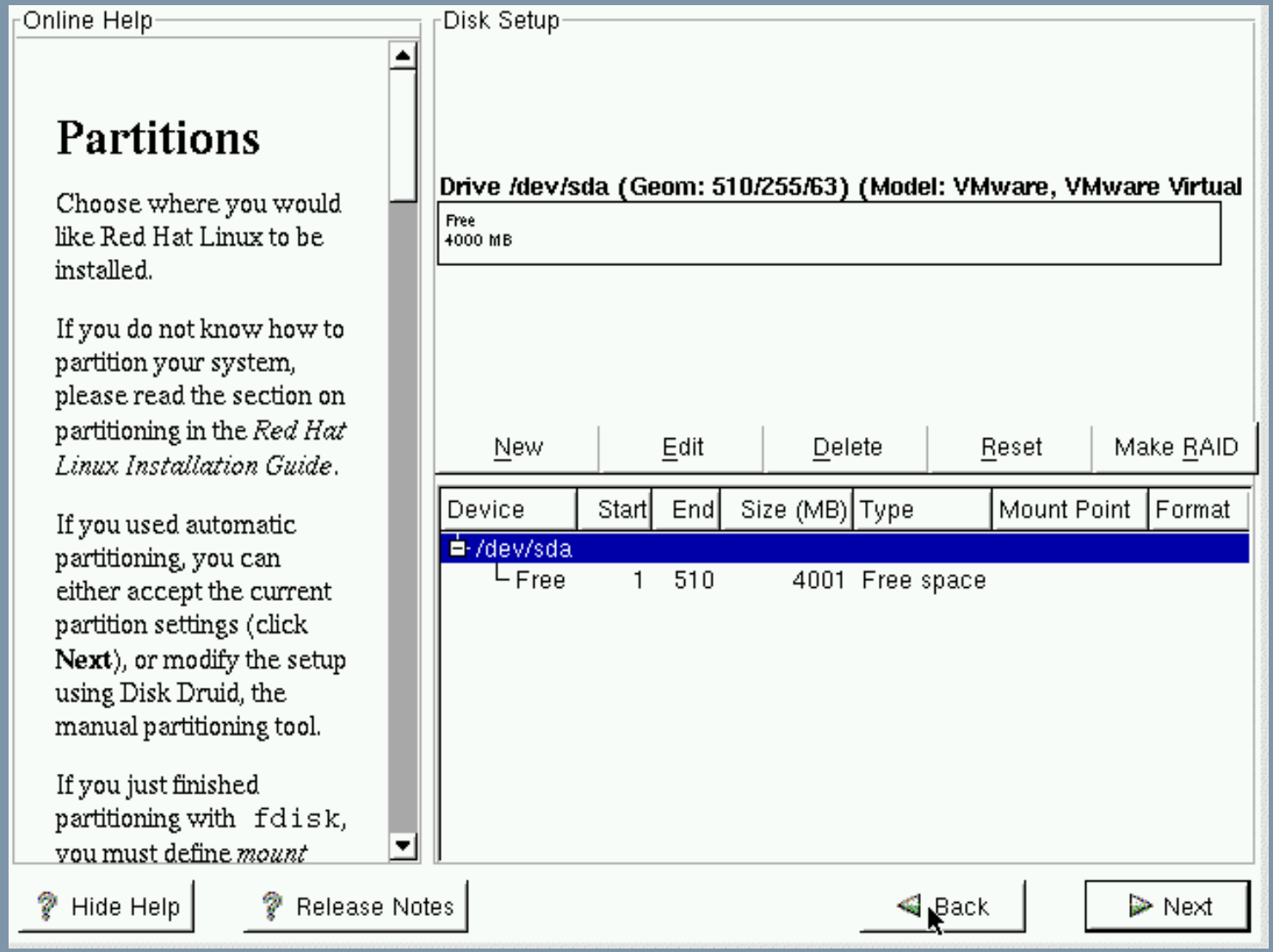
technical computing – invent. design. deliver.

Disk Partitioning: Disk Druid



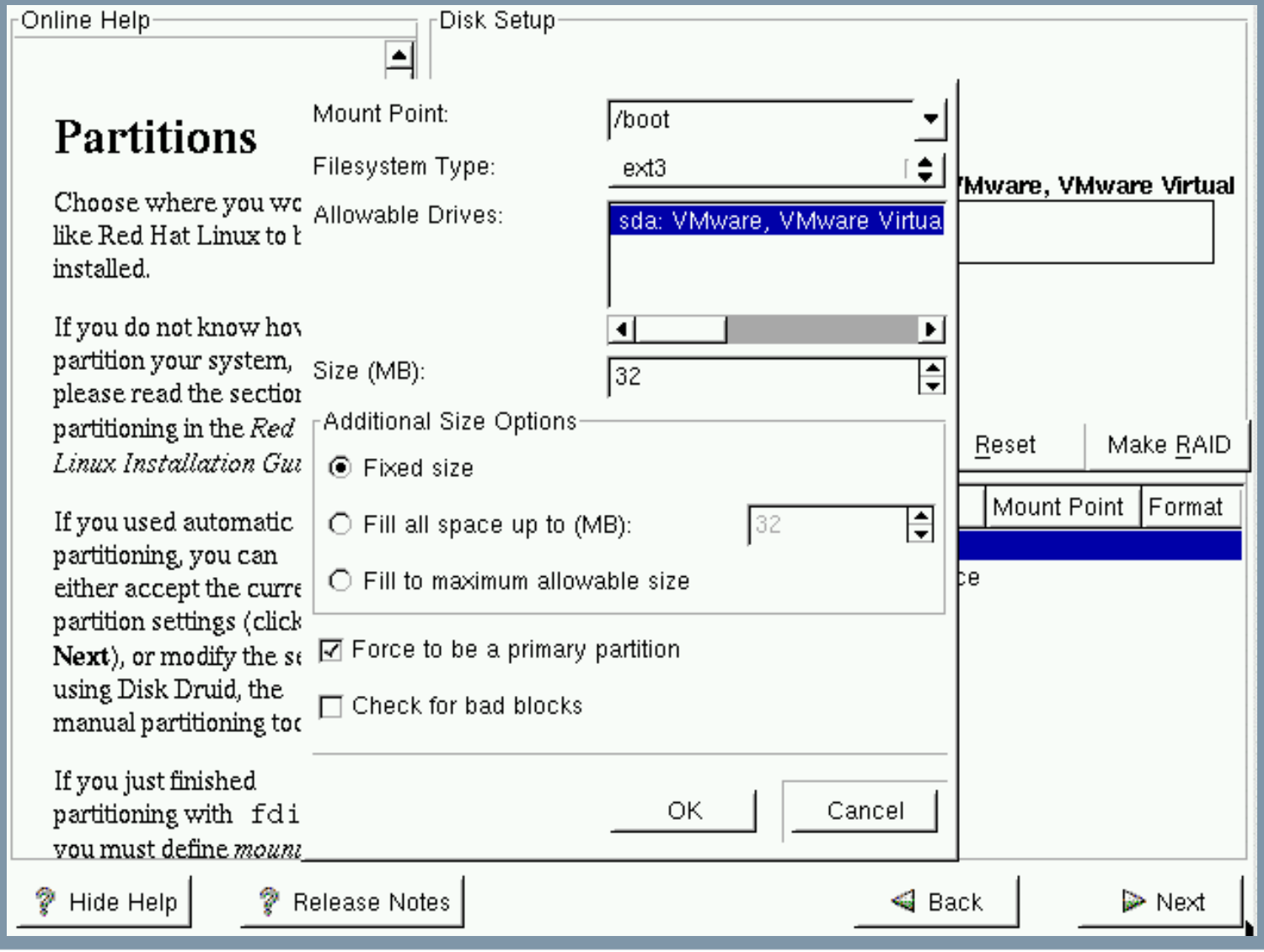
technical computing – invent. design. deliver.

Disk Partitioning: Empty Disk



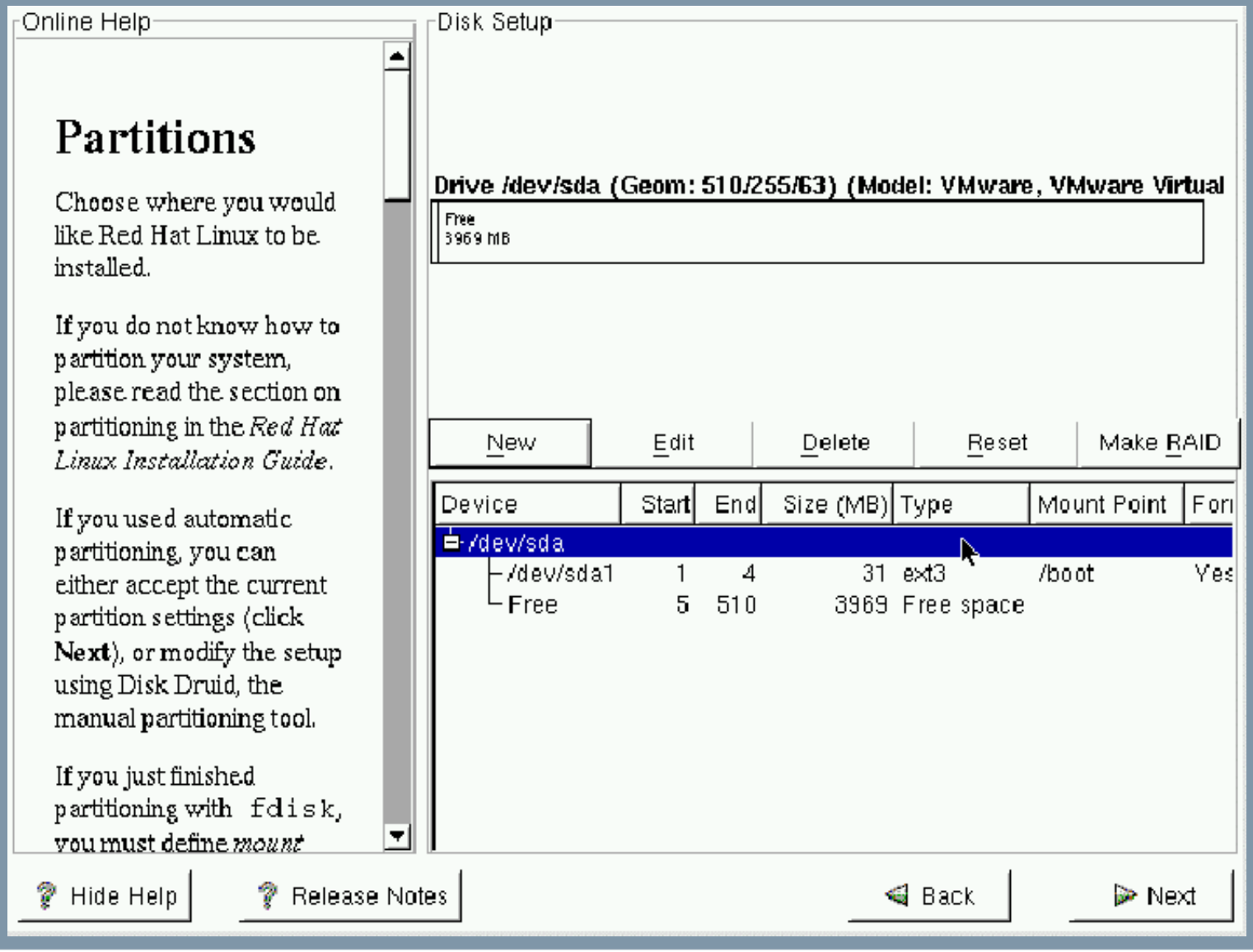
technical computing – invent. design. deliver.

Disk Partitioning: Creating /boot



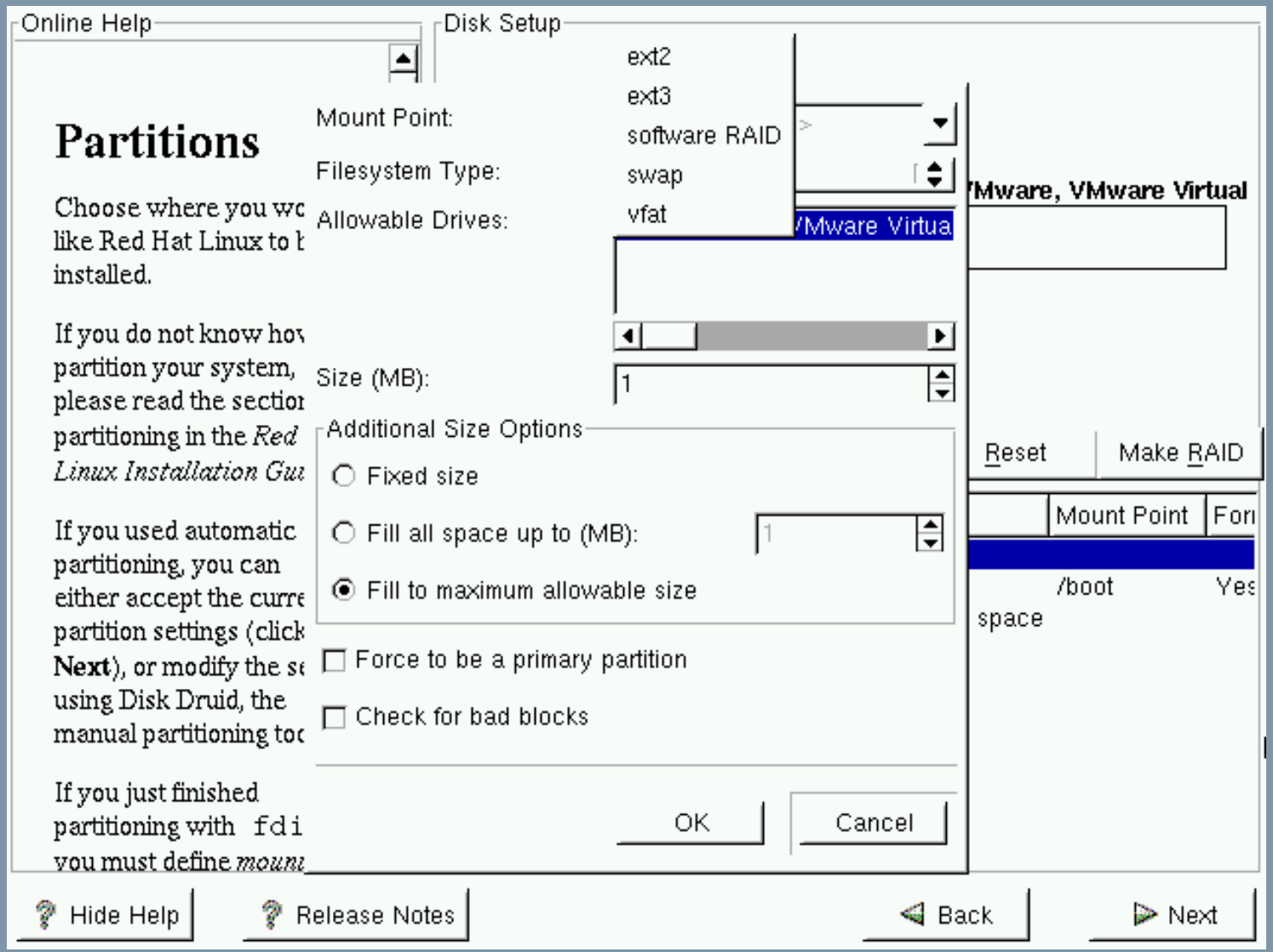
technical computing – invent. design. deliver.

Disk Partitioning: /boot Created



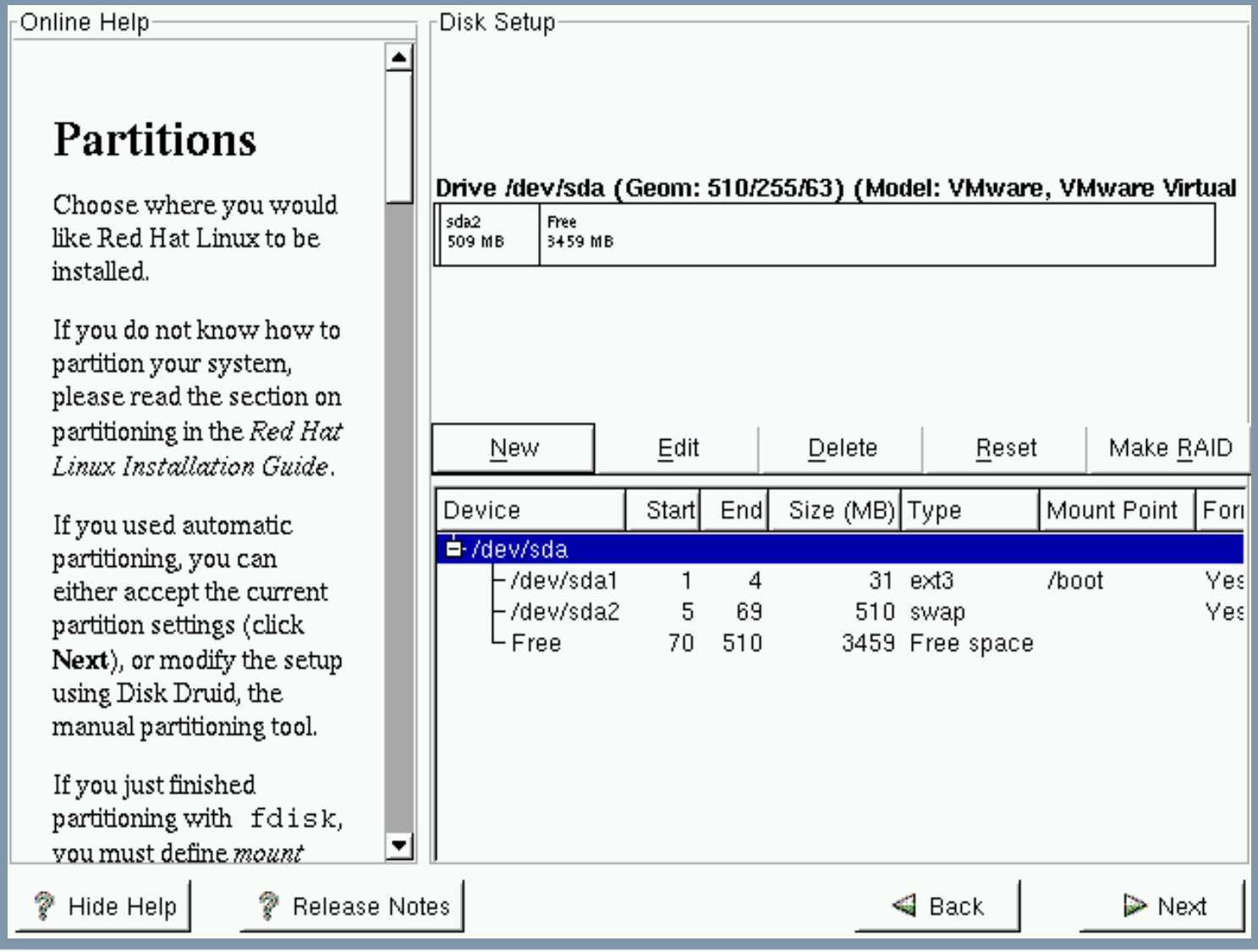
technical computing – invent. design. deliver.

Disk Partitioning: Format Choices



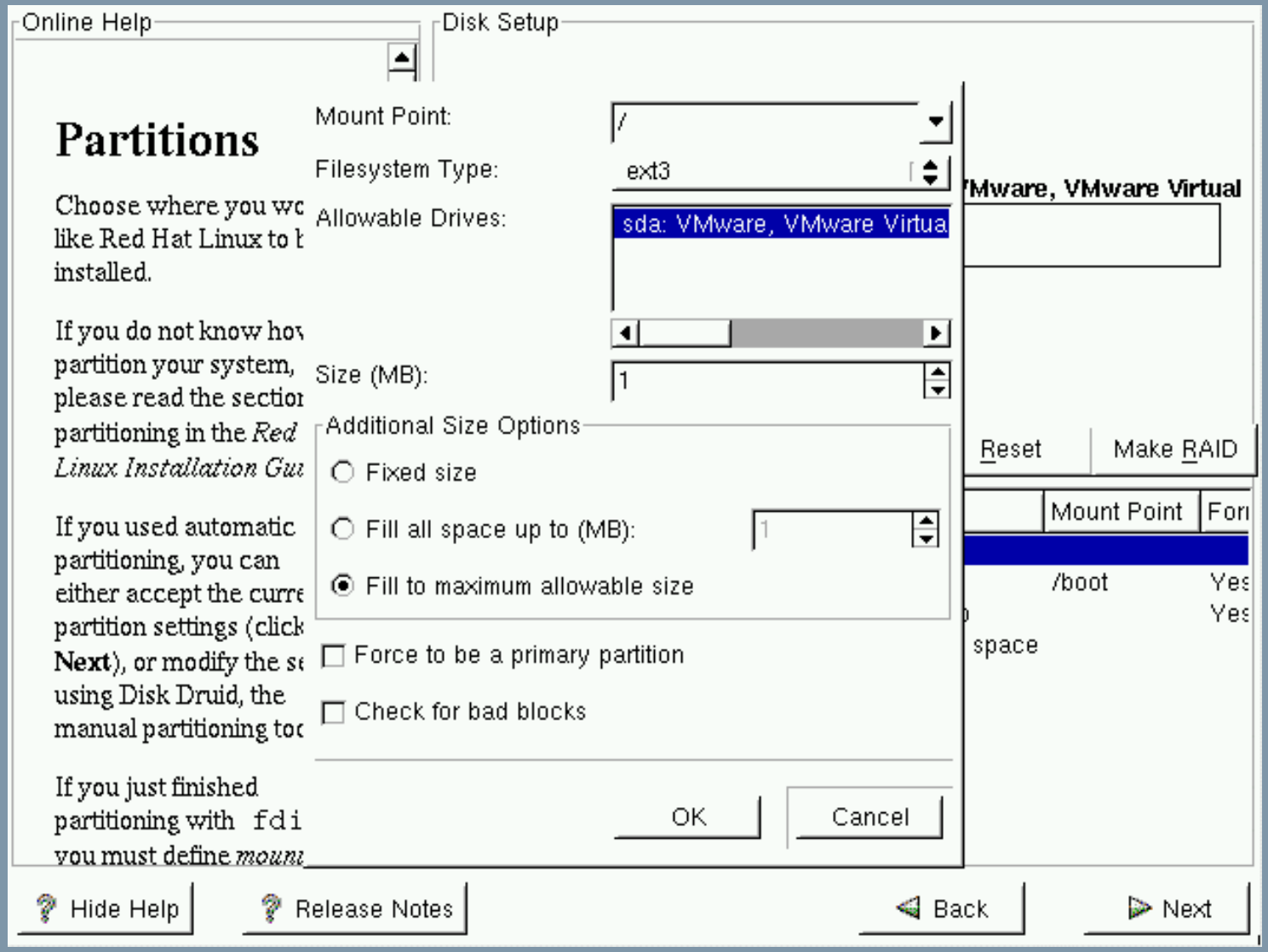
technical computing – invent. design. deliver.

Disk Partitioning: Swap Created



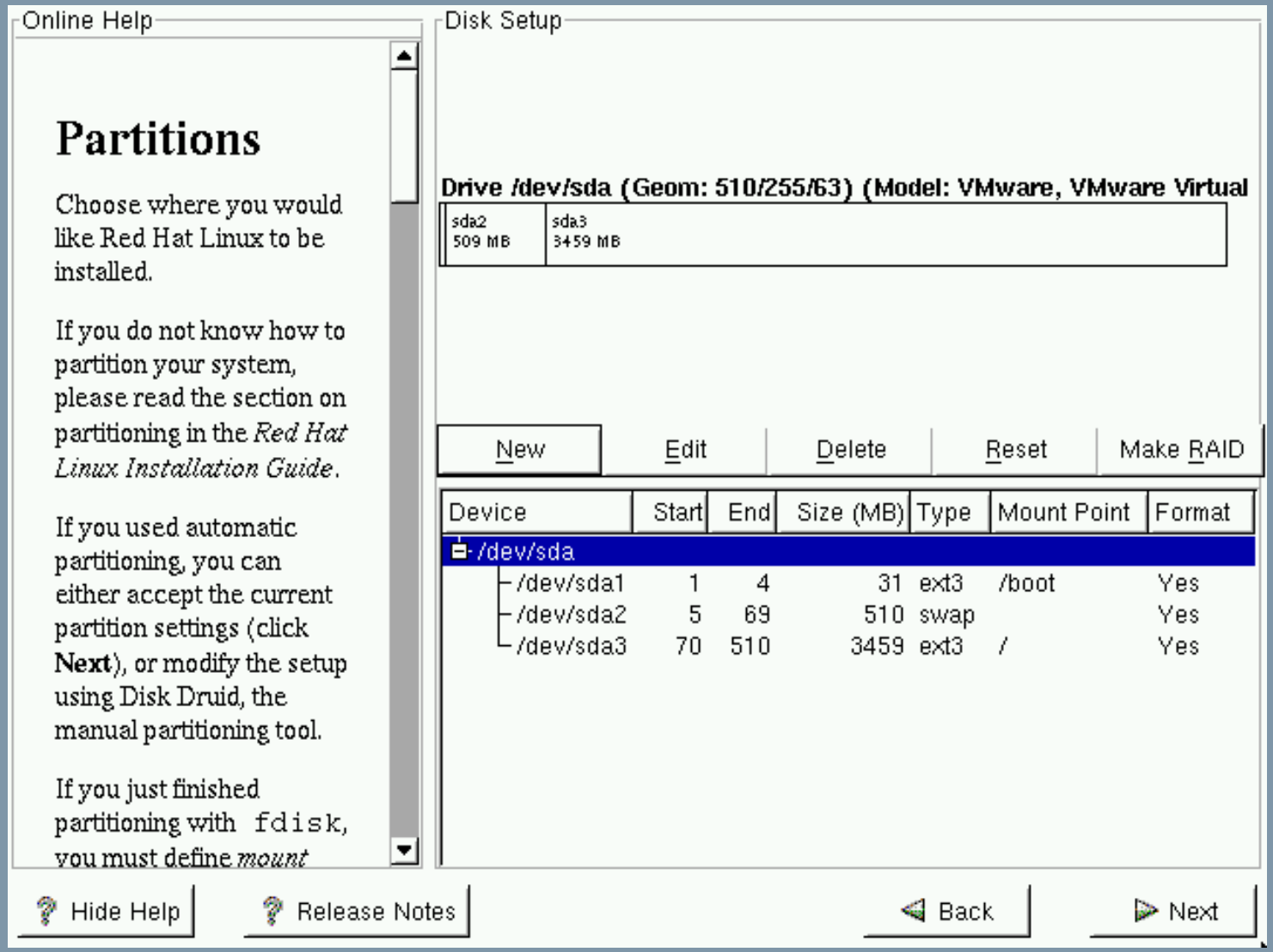
technical computing – invent. design. deliver.

Disk Partitioning: Creating /



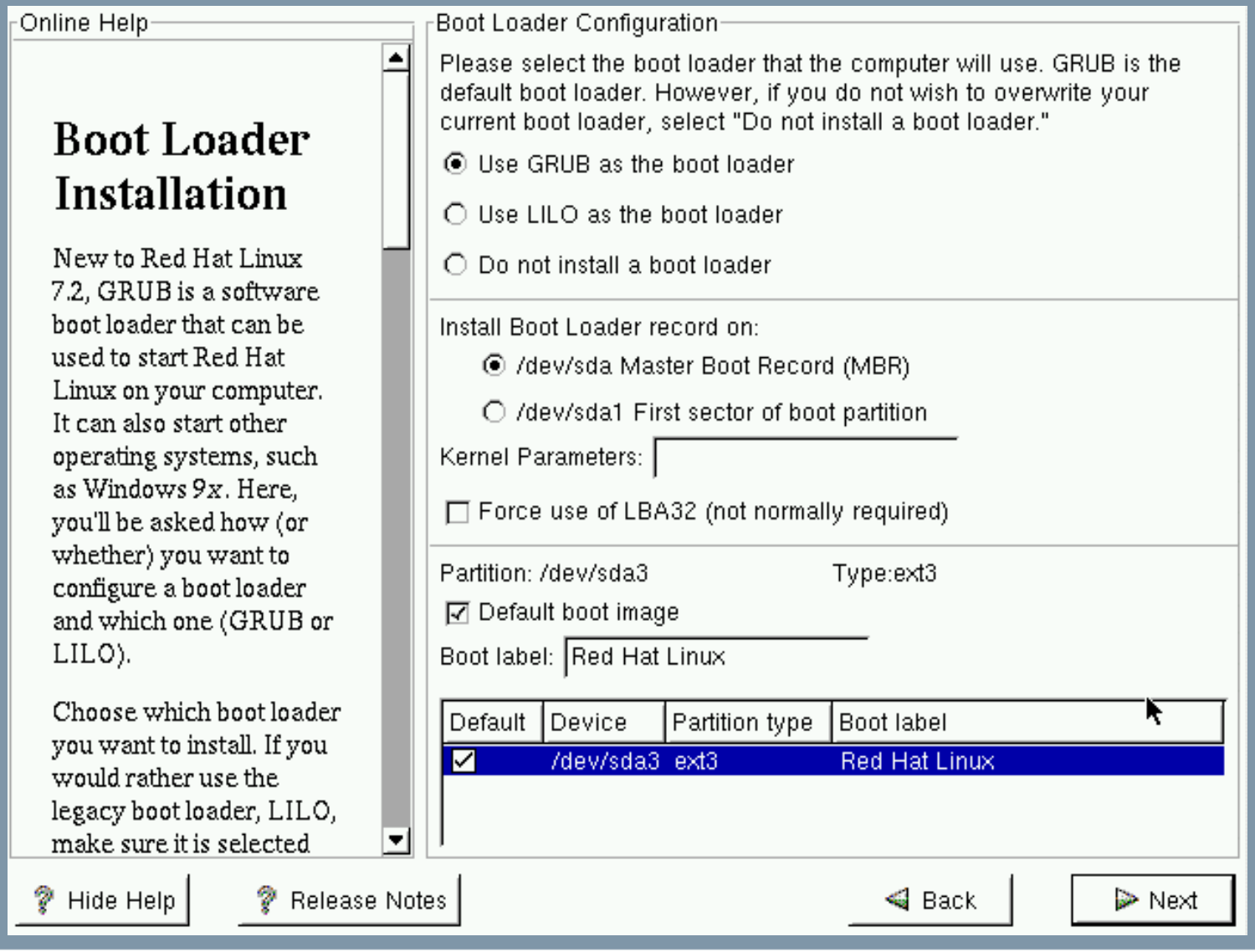
technical computing – invent. design. deliver.

Disk Partitioning: Final Configuration



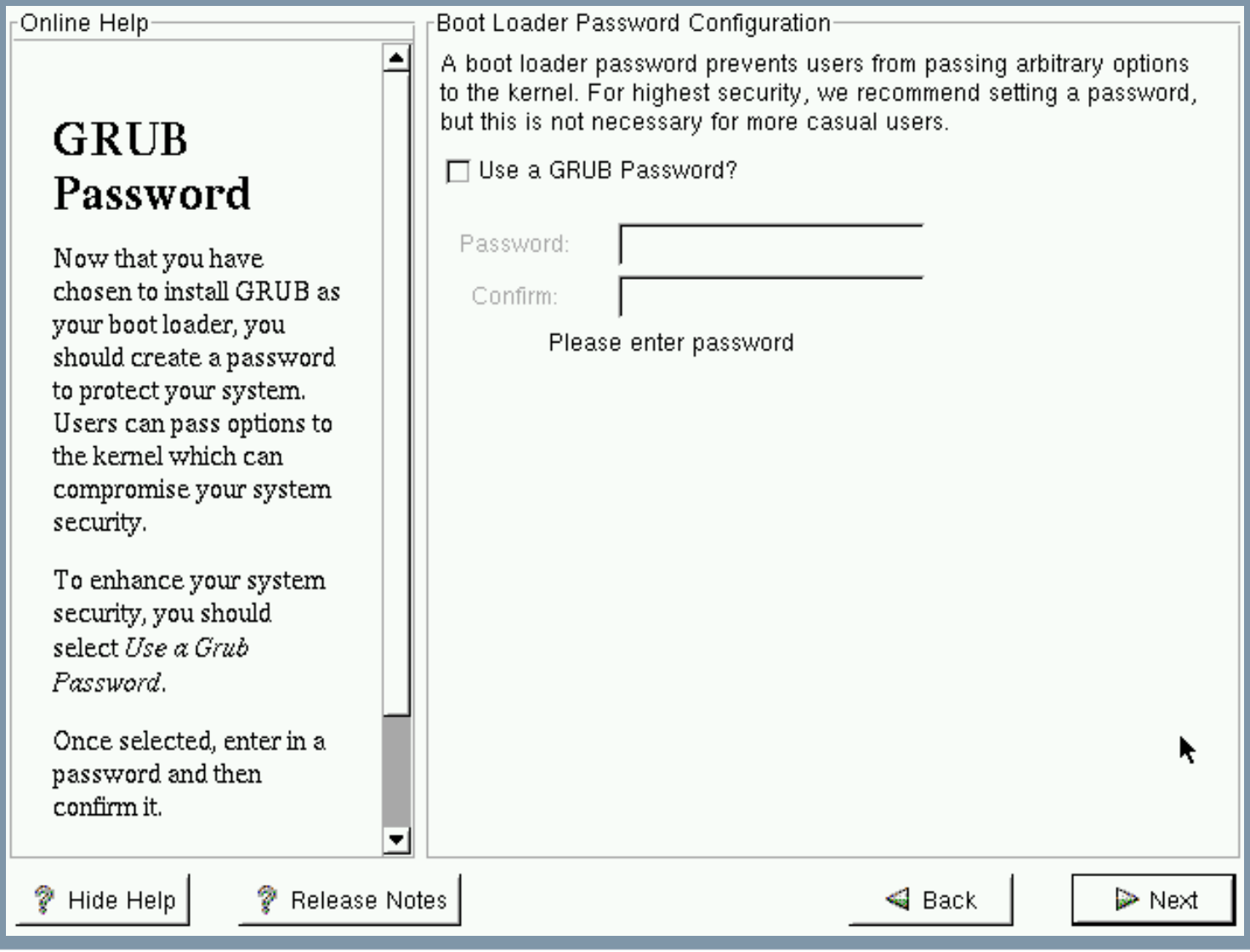
technical computing – invent. design. deliver.

Boot Loader Selection and Installation



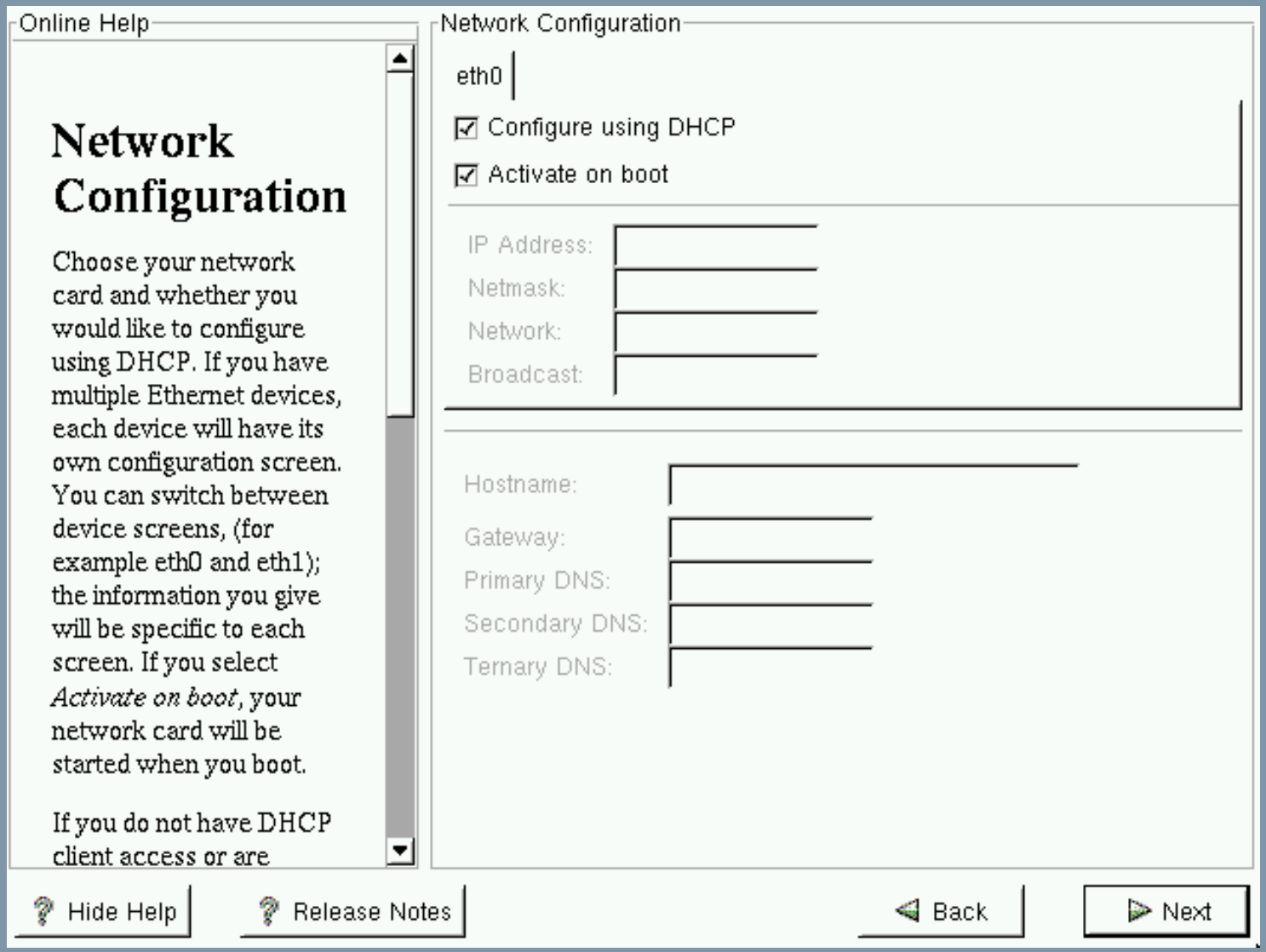
technical computing – invent. design. deliver.

GRUB Password Initialization



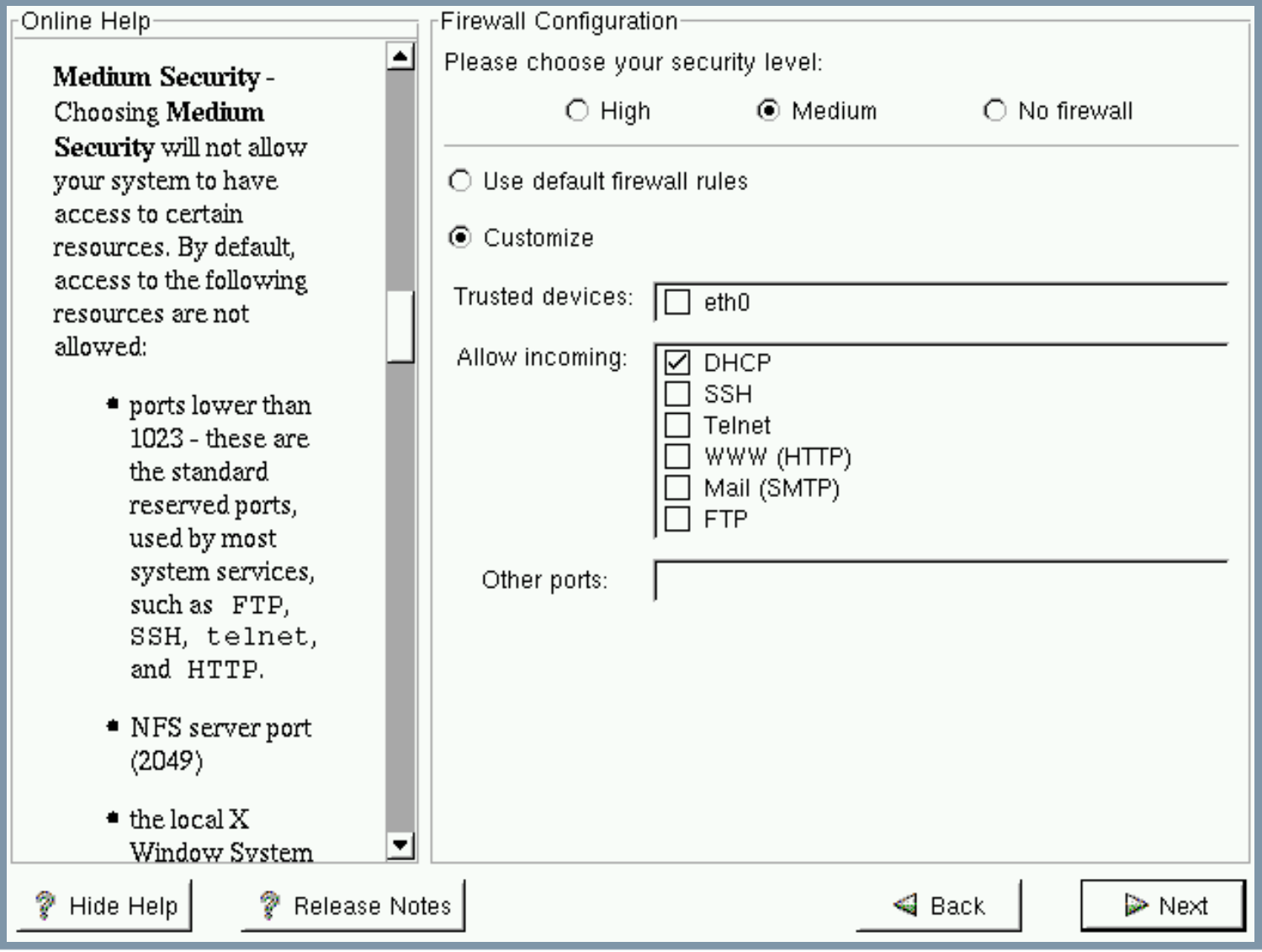
technical computing – invent. design. deliver.

Network Parameter Configuration



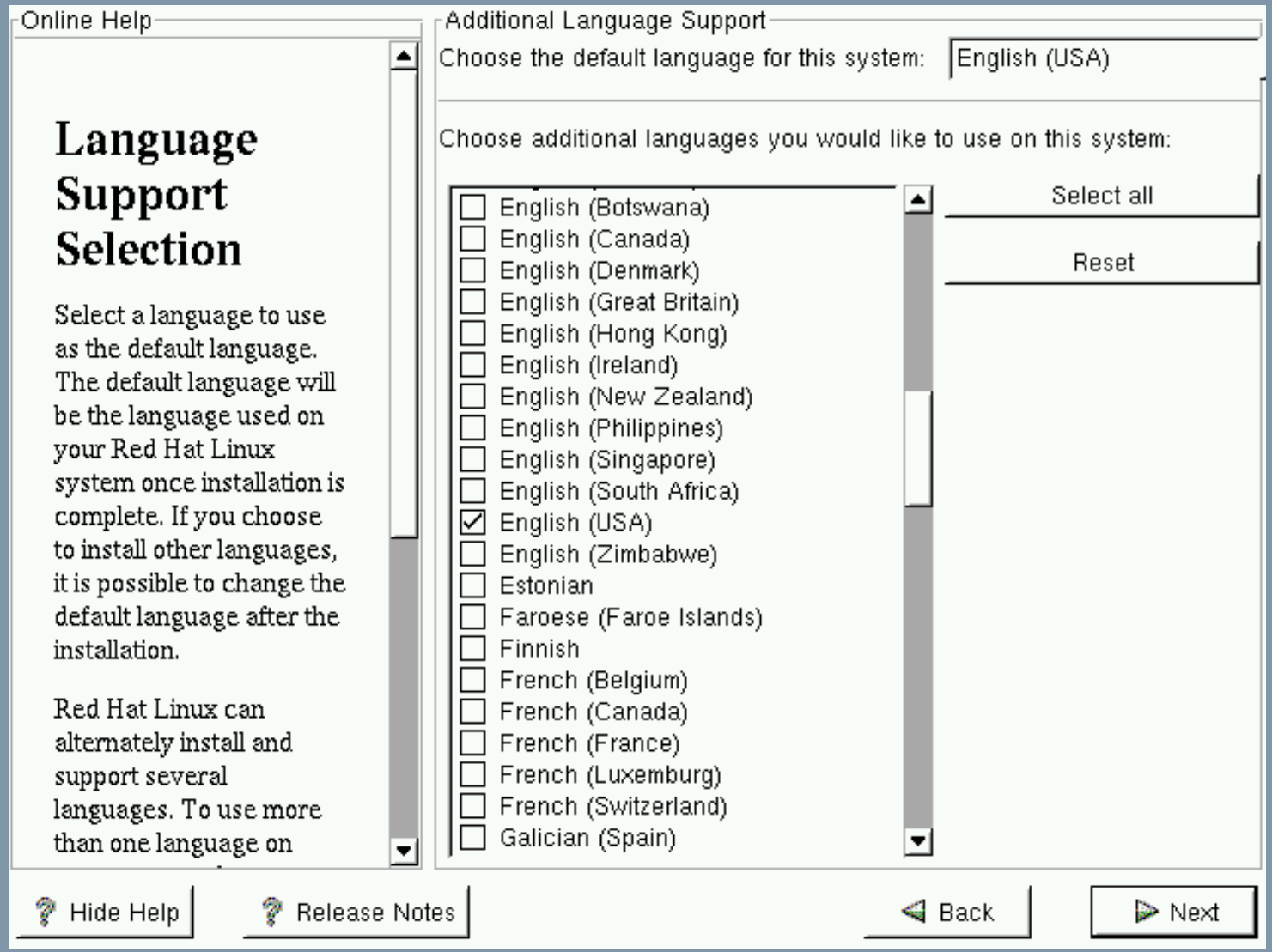
technical computing – invent. design. deliver.

Firewall Installation and Configuration



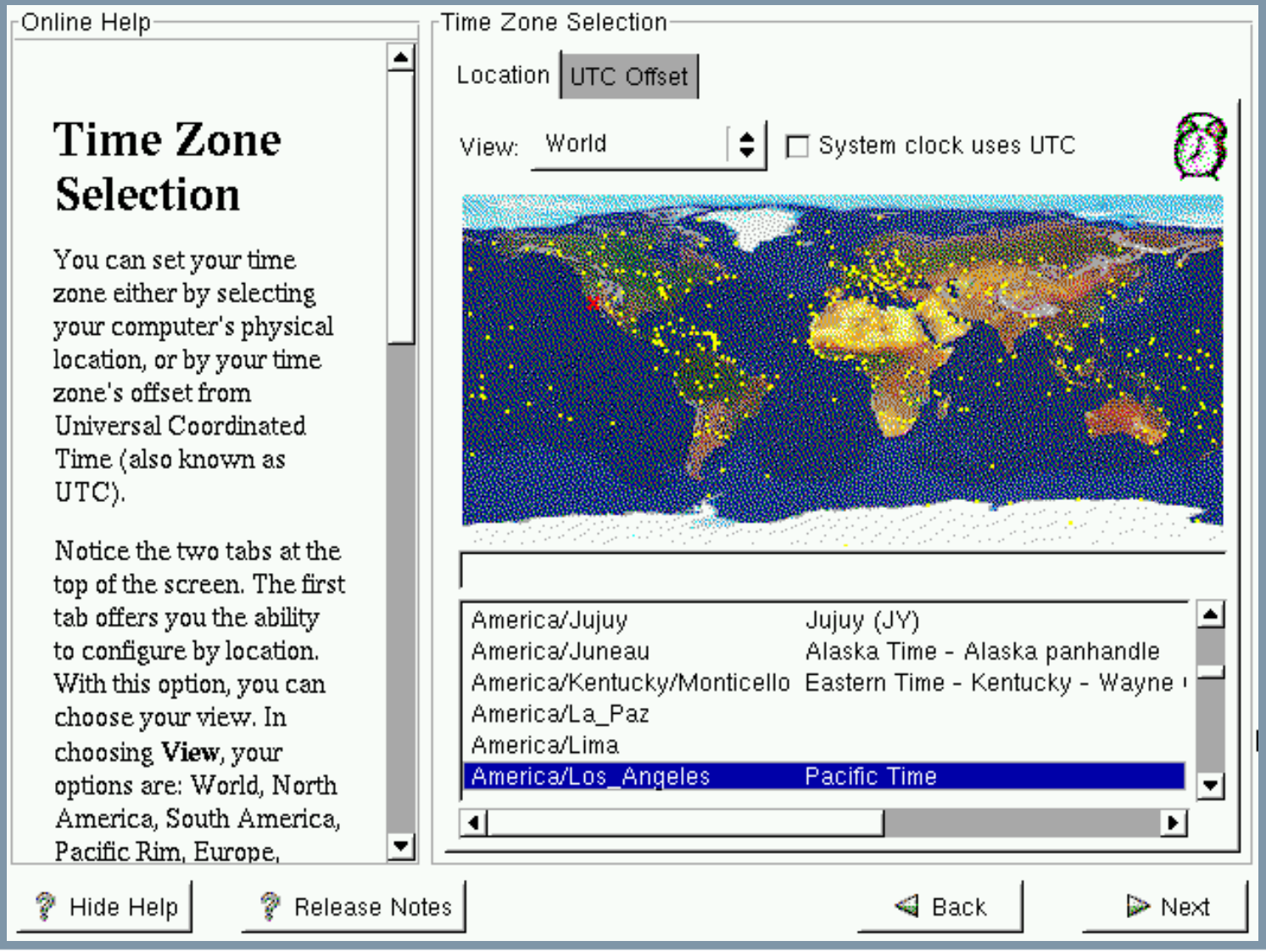
technical computing – invent. design. deliver.

Additional Language Installation



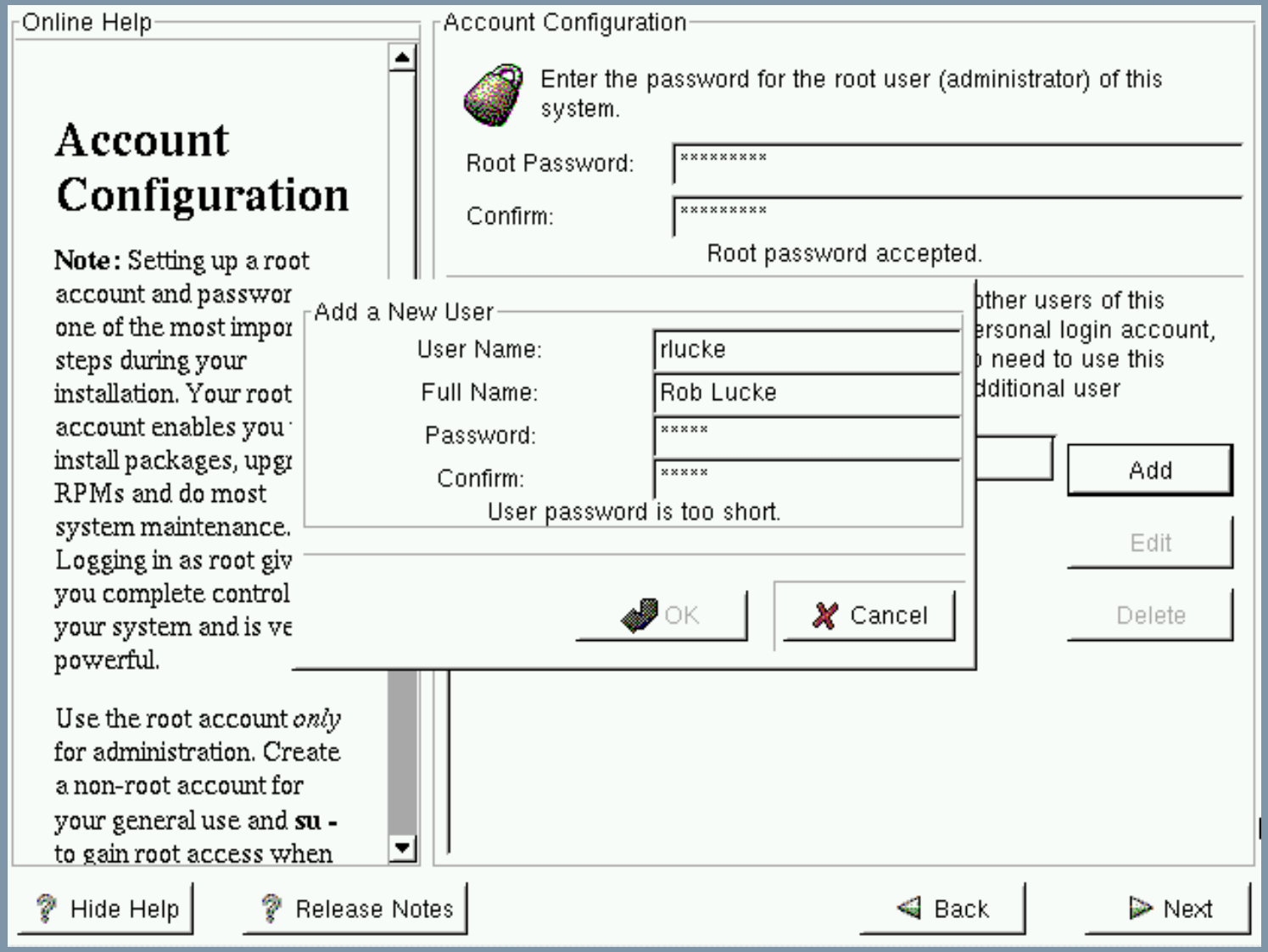
technical computing – invent. design. deliver.

Setting the Time Zone



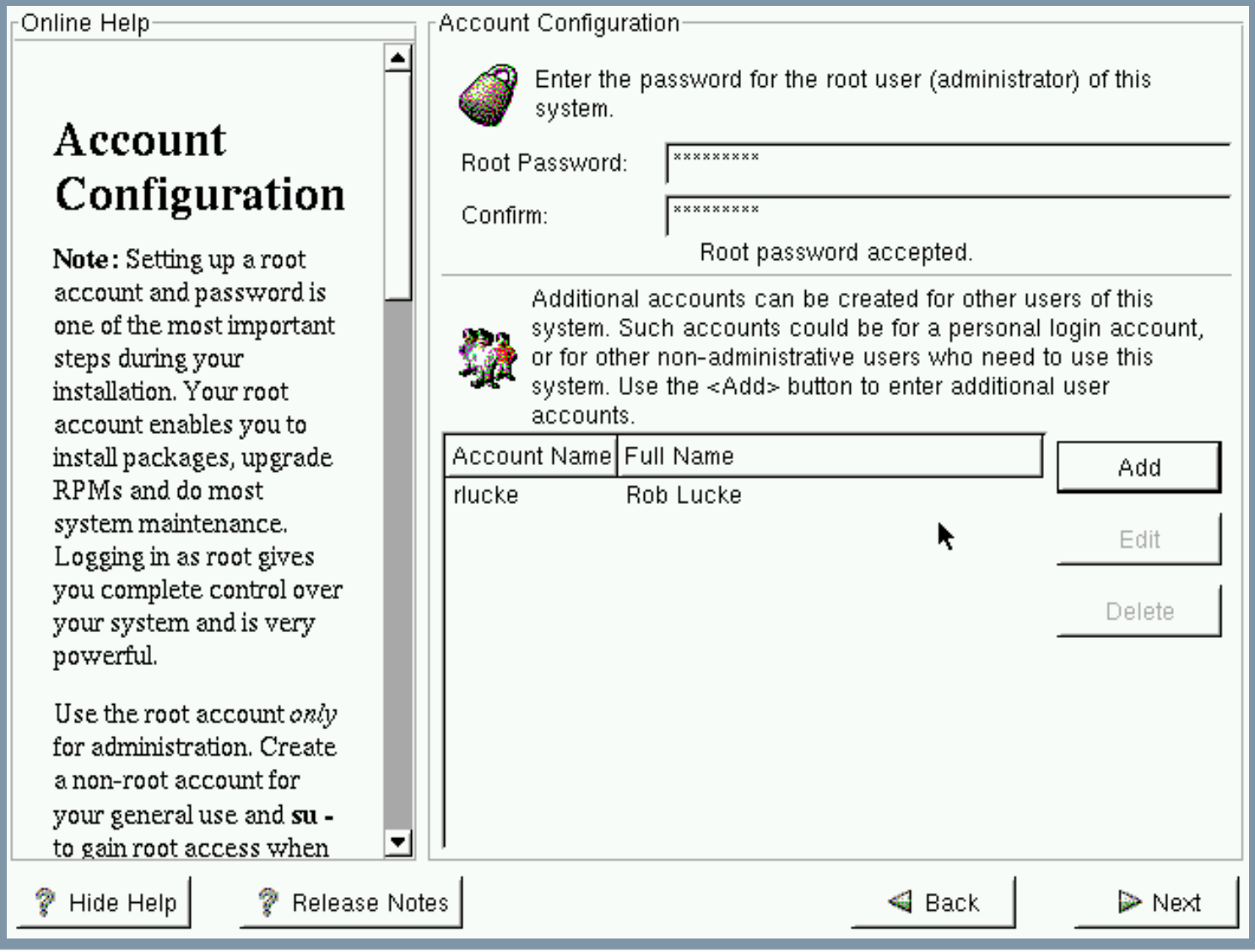
technical computing – invent. design. deliver.

Setting Initial Accounts



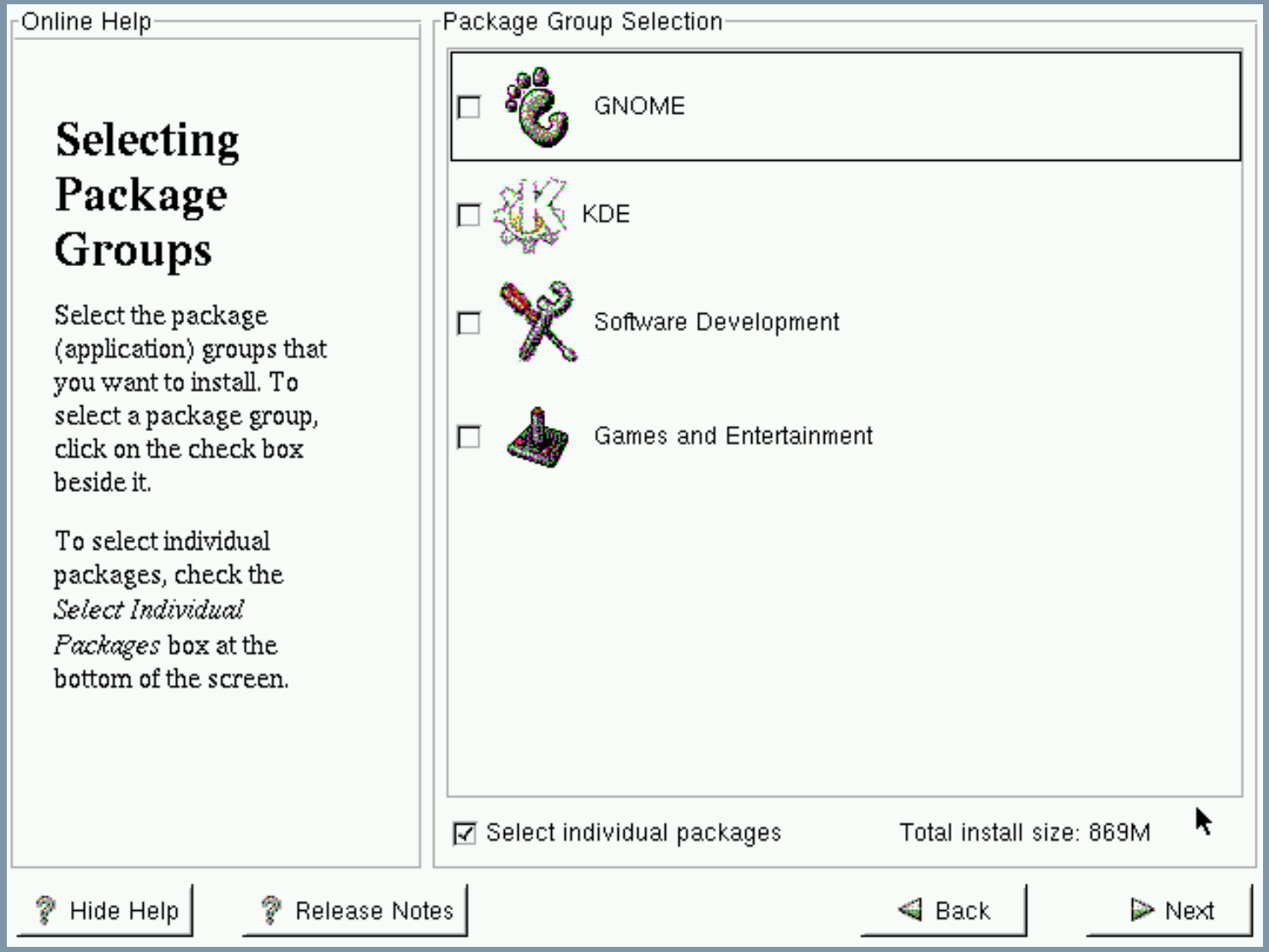
technical computing – invent. design. deliver.

Setting Initial Accounts Completed



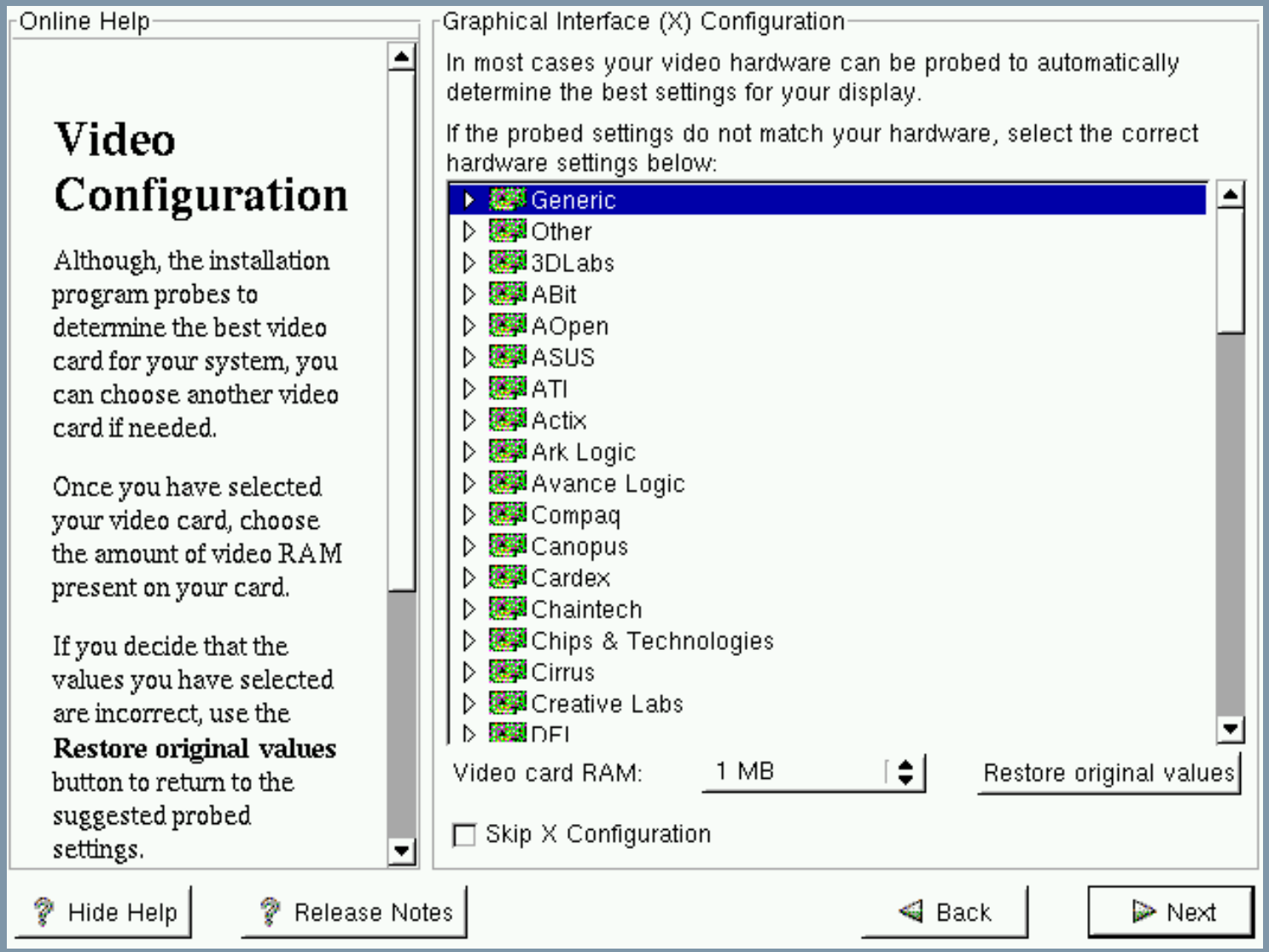
technical computing – invent. design. deliver.

Choosing Package Groups



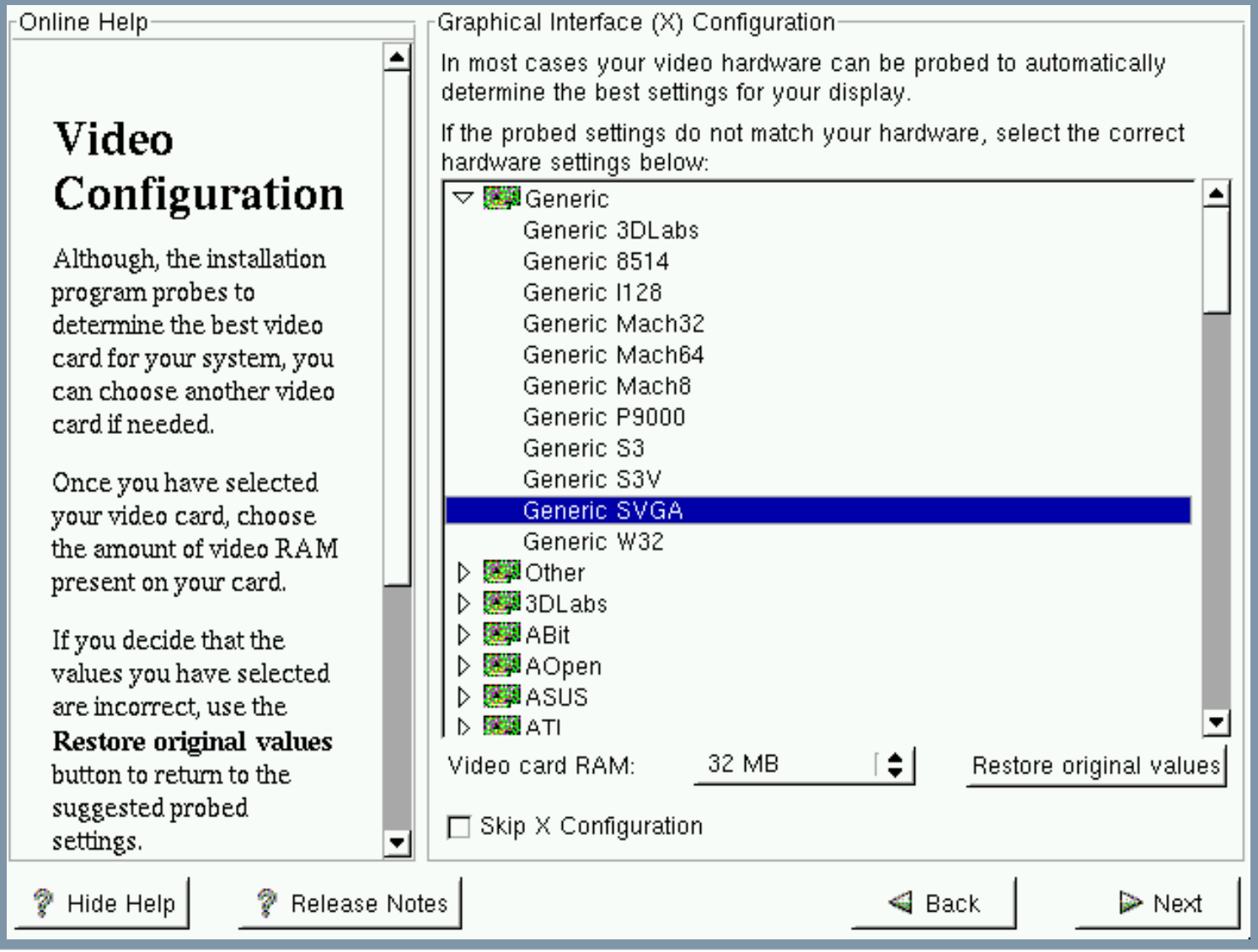
technical computing – invent. design. deliver.

Choosing the Video Interface



technical computing – invent. design. deliver.

Video Card Selected



technical computing – invent. design. deliver.

Installation Point of No Return

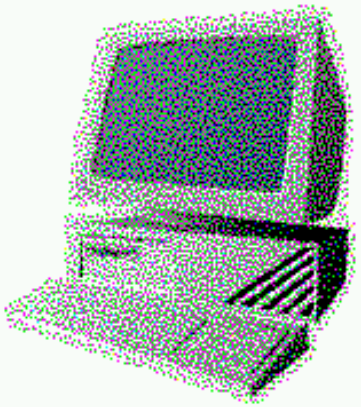
Online Help

About to Install

Caution: Once you click **Next**, the installation program will begin writing the Red Hat Linux operating system to your hard drive. This process cannot be undone. If you have decided not to install Red Hat Linux, this is the last point at which you can abort the installation process.

To abort this installation, remove all Red Hat Linux media, and press your computer's **Reset** button or reset using **Control-Alt-Delete**.

About to Install



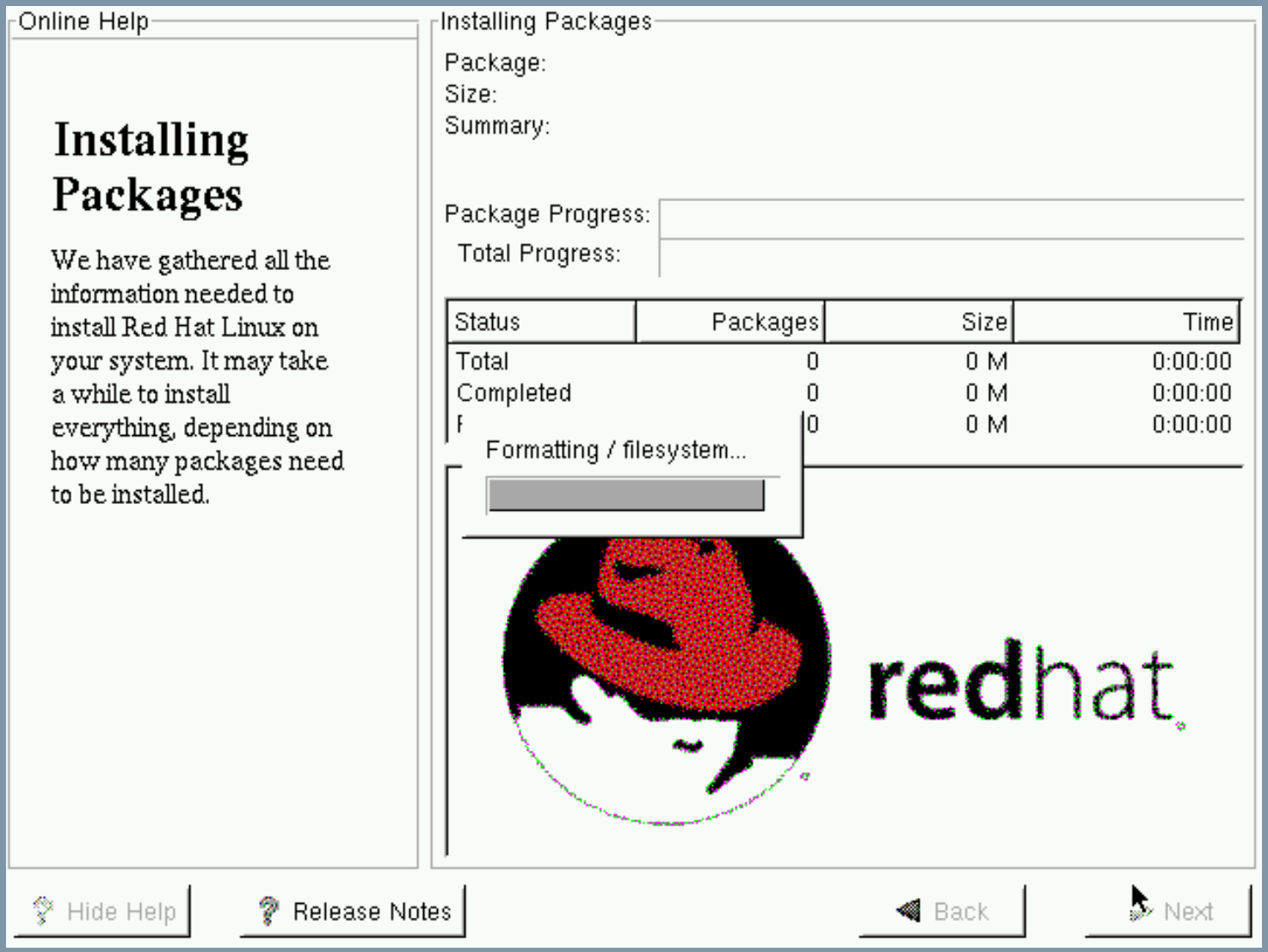
Click next to begin installation of Red Hat Linux.

A complete log of your installation will be in /tmp/install.log after rebooting your system. You may want to keep this file for later reference. A kickstart file representing the choices you have made will be in /root/anaconda-ks.cfg.

Hide Help | Release Notes | Back | Next

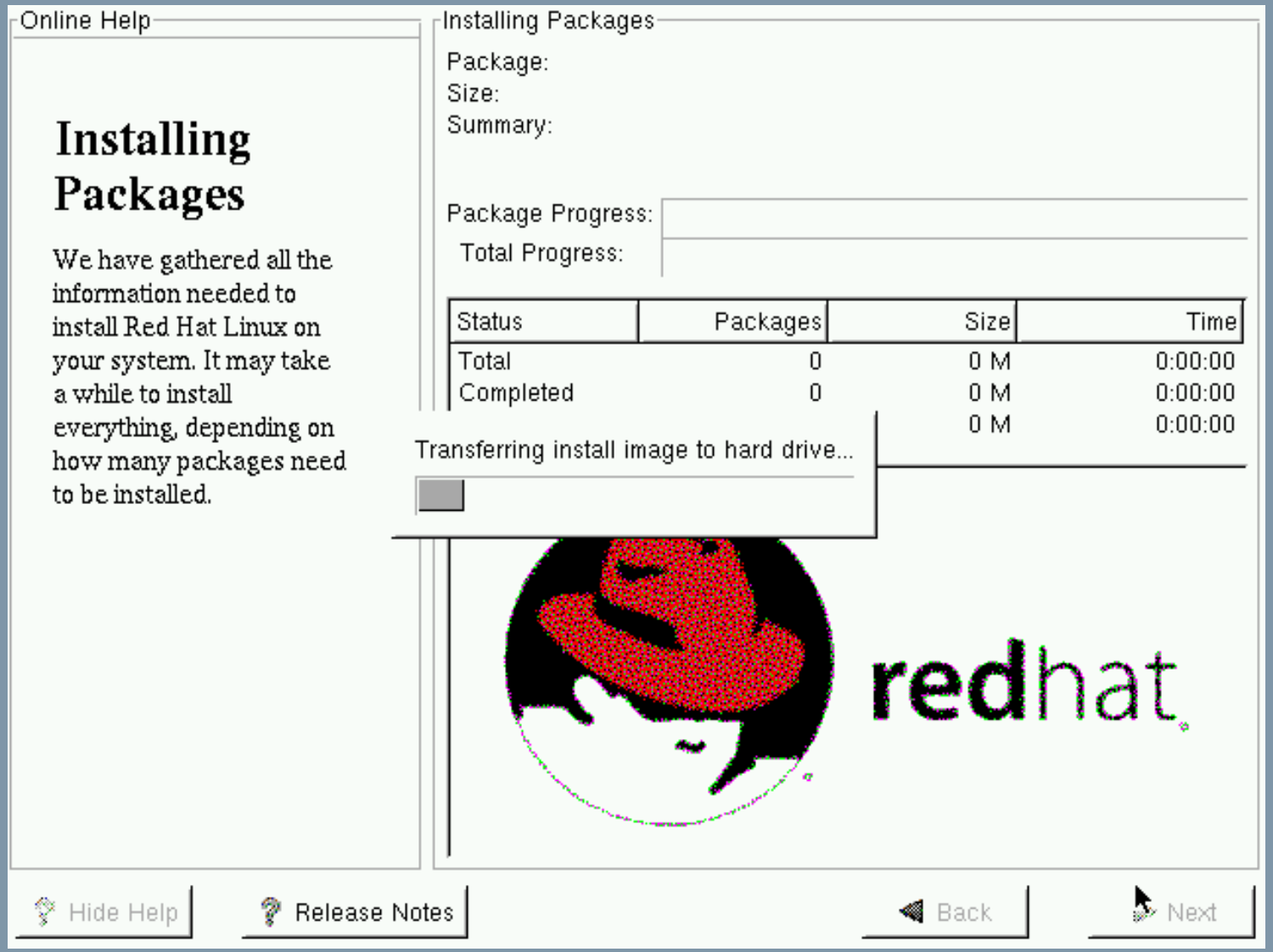
technical computing – invent. design. deliver.

File System Formatting



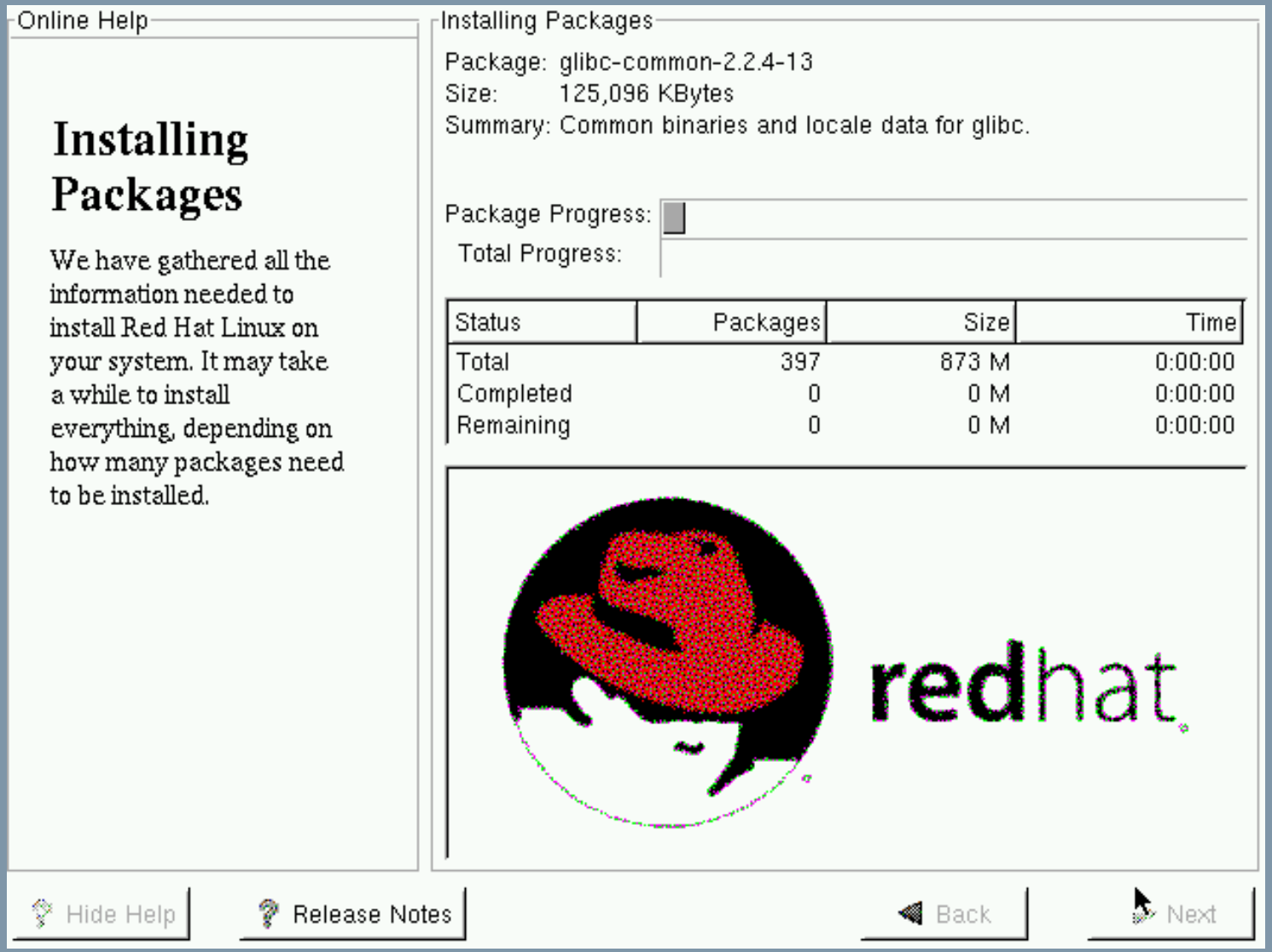
technical computing – invent. design. deliver.

Transfer Installation Image



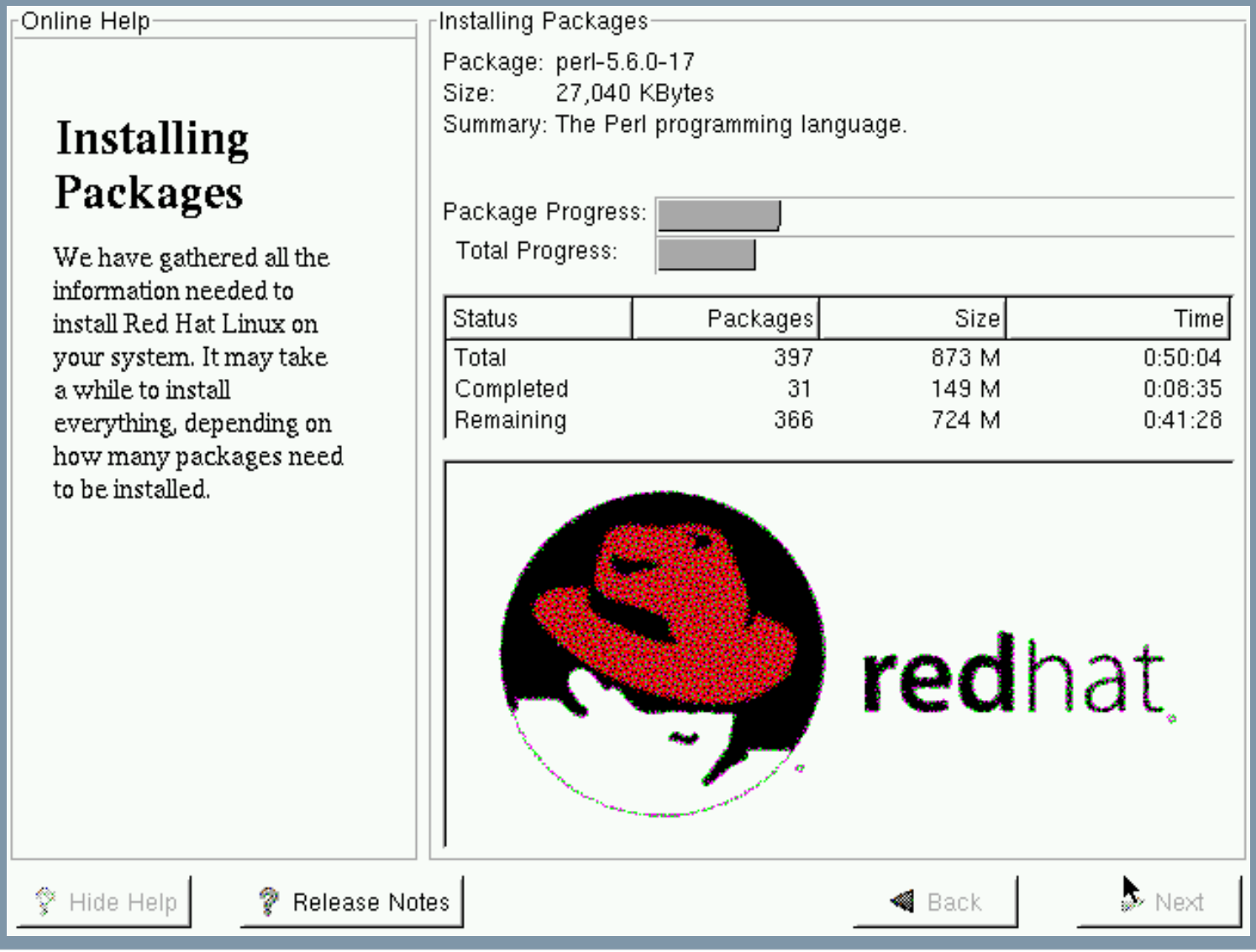
technical computing – invent. design. deliver.

Install Packages: Start



technical computing – invent. design. deliver.

Install Packages: Continuing

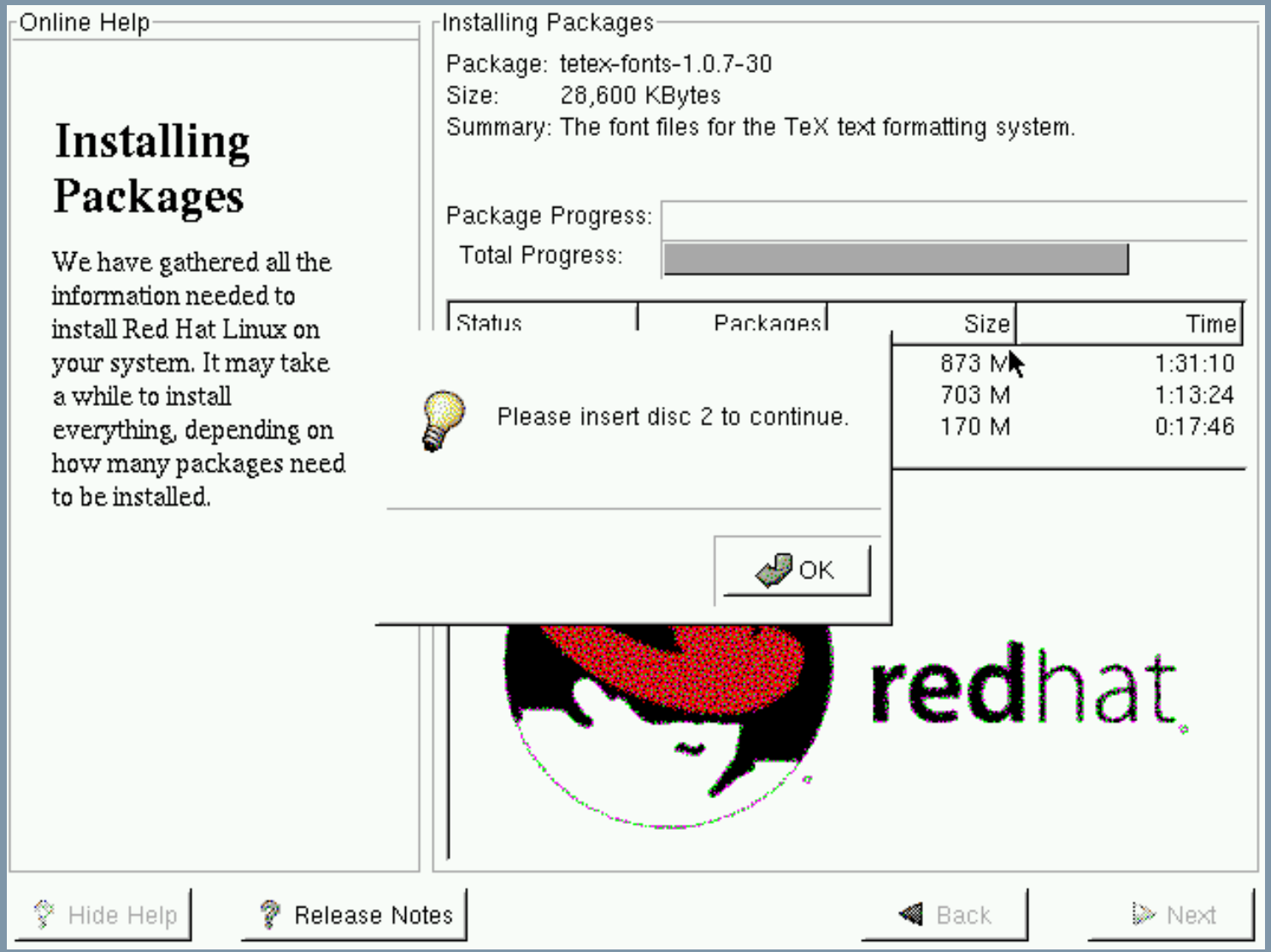


Installing Packages

We have gathered all the information needed to install Red Hat Linux on your system. It may take a while to install everything, depending on how many packages need to be installed.

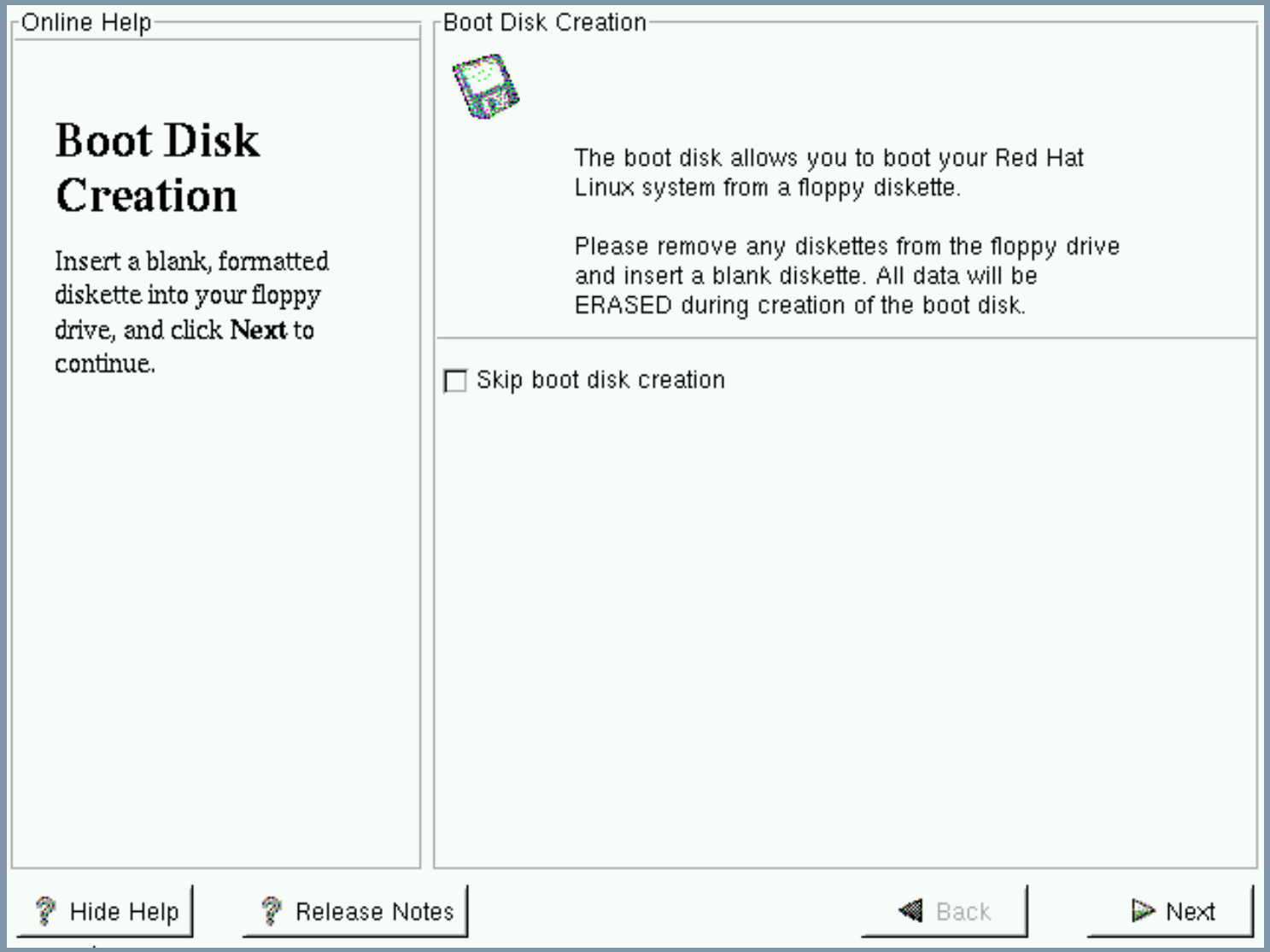
technical computing – invent. design. deliver.

Install Packages: CDROM Swap



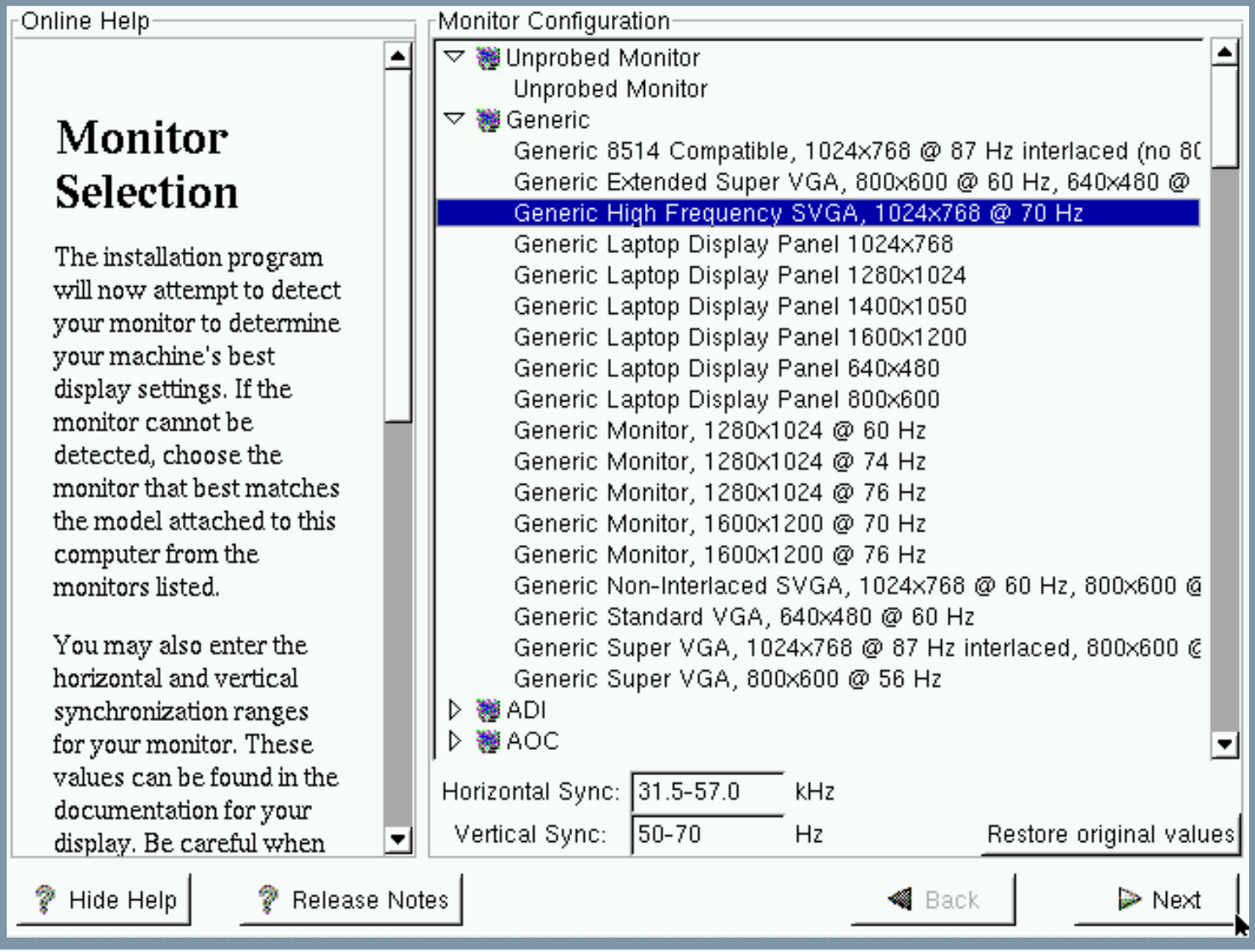
technical computing – invent. design. deliver.

Create Boot Floppy



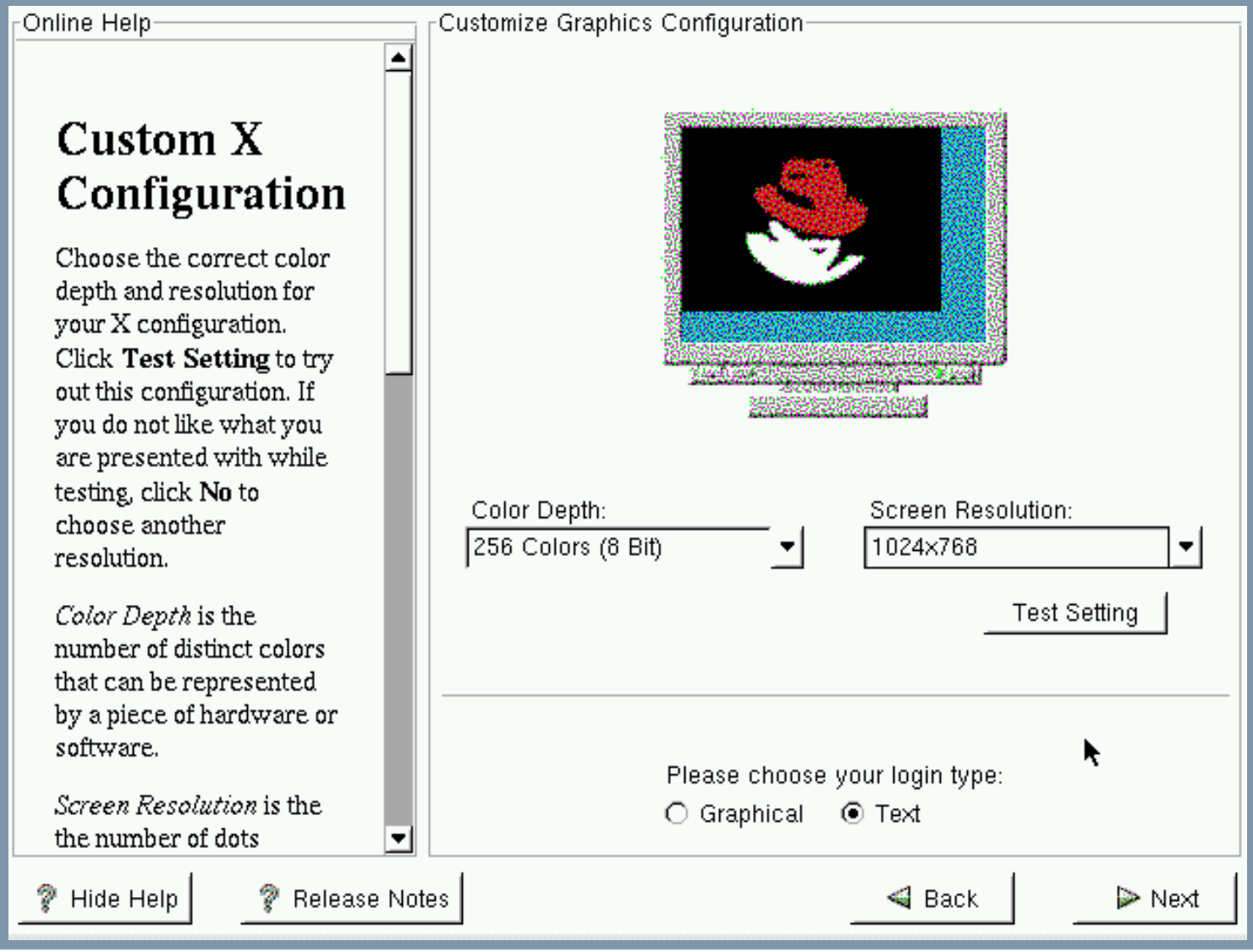
technical computing – invent. design. deliver.

Choose Video Display Parameters



technical computing – invent. design. deliver.

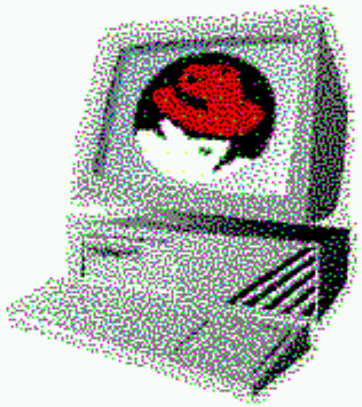
X-Windows Server Configuration



technical computing – invent. design. deliver.

Install Complete, Reboot

Congratulations



Congratulations, your Red Hat Linux installation is complete.

Remove any floppy diskettes you used during the installation process and press <Enter> to reboot your system.

If you created a boot disk to use to boot your Red Hat Linux system, insert it before you press <Enter> to reboot.

For information on errata (updates and bug fixes), visit http://www.redhat.com/errata.

Information on using and configuring your system is available in the Red Hat Linux manuals at http://www.redhat.com/support/manuals.

Show Help | Release Notes | Back | Exit

technical computing – invent. design. deliver.

Lessons Learned about Installation and Booting

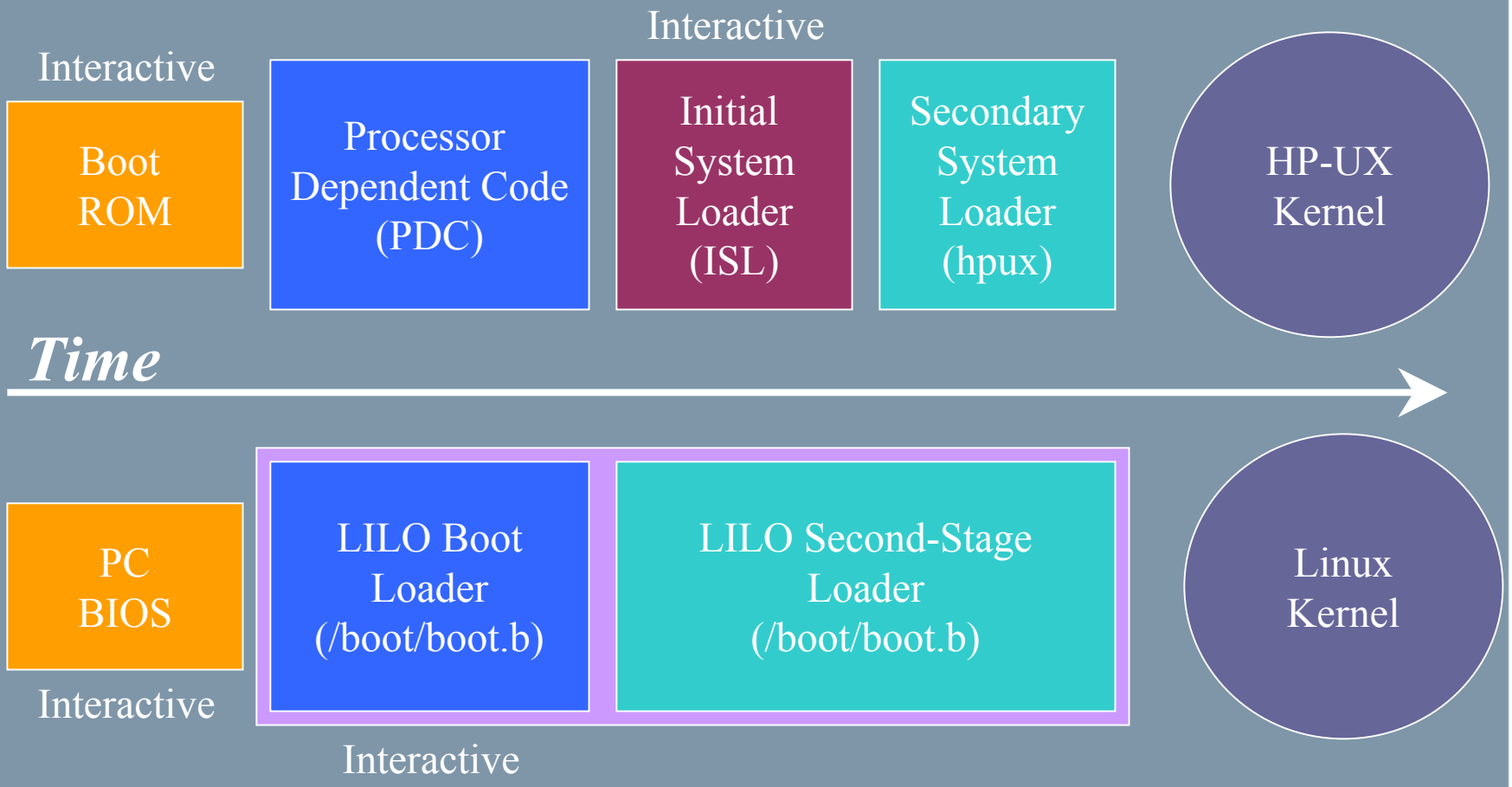
- Always make a boot disk!
- GRUB is a good boot manager and replaces LILO …
- GRUB understands file systems structure and is not dependent on offset like LILO (don't have to re-run it every change you make)
- Make a GRUB boot/installation disk, too!
- Always make a boot disk!
- You can place the /boot partition into a small partition on the first IDE/SCSI drive and still mount the Linux root partition from elsewhere
- Note that SCSI disk 0 has the LOWEST priority in the SCSI chain, so the PC BIOS has things a little backwards …
- ALWAYS MAKE A BOOT DISK!

technical computing – invent. design. deliver.

Boot Loaders

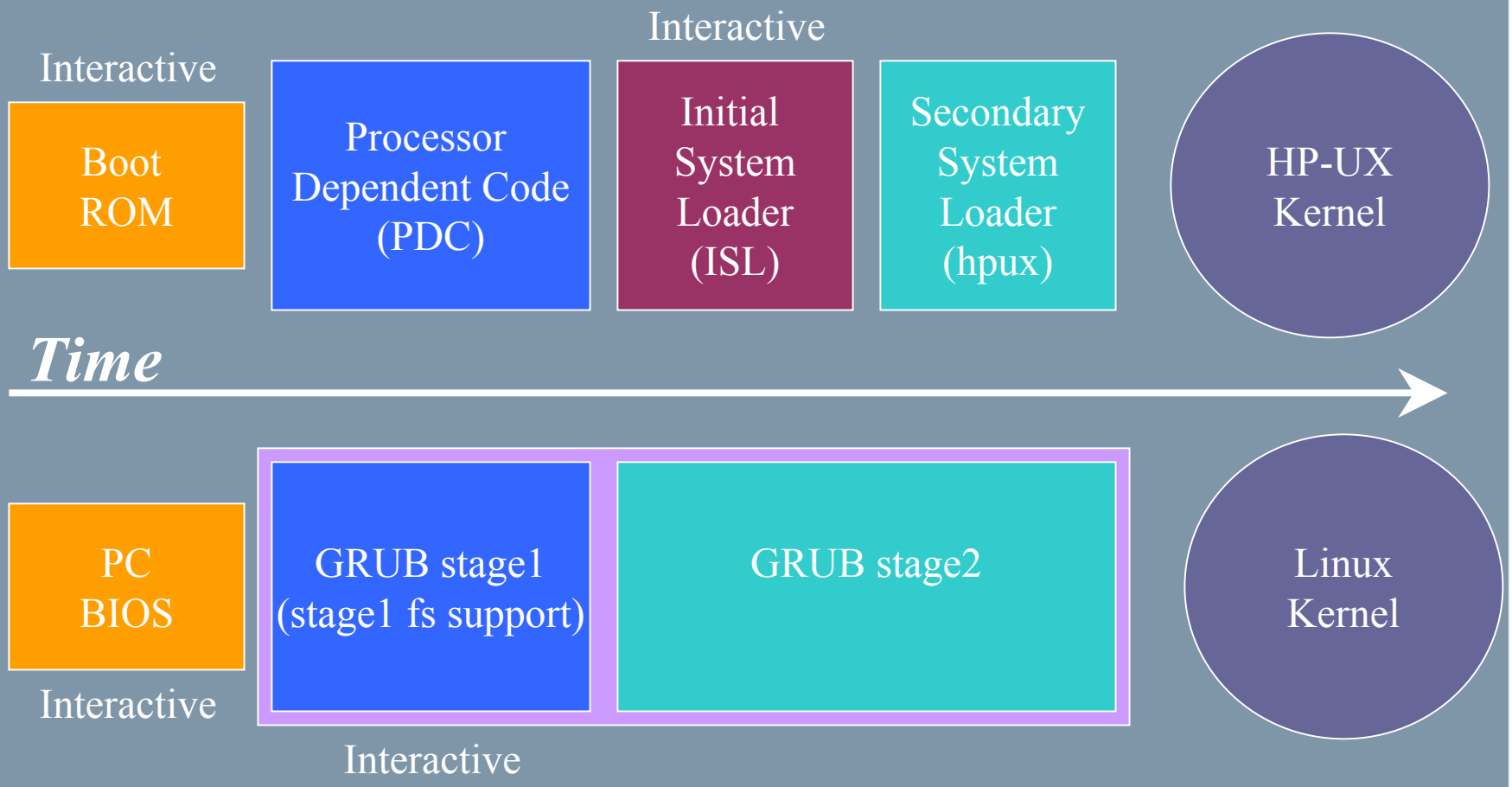
LILO
GRUB

Comparing the HP-UX and LILO Boot Sequences



technical computing – invent. design. deliver.

Comparing the HP-UX and GRUB Boot Sequences



technical computing – invent. design. deliver.

technical computing – invent. design. deliver.

Linux LILO Tips

- LILO is not the only way to boot a Linux system (thank goodness)
  - Other Linux boot managers (GRUB, bootactv, loadlin, etc.)
  - Commercially available boot managers (i.e. BootMagic, System Commander, etc.)
  - Windows NT/XP boot manager
    - "dd if=/dev/hda of=/tmp/linux.bs bs=512 count=1"
    - Move "linux.bs" to Windows NT "root" and configure into "boot.ini" file
- The important thing to remember: whatever boot manager is in control must be able to find and access the boot sector containing LILO's loader
- If your system's hard drive configuration does not meet the Linux booting requirements, then you can boot from a floppy disk
- If your system is ONLY Linux, then things are much simpler!

technical computing – invent. design. deliver.

GRand Unified Bootloader (GRUB) Tips

- /usr/share/grub/i386-redhat contains the “raw” GRUB installation files
- /boot/grub contains grub boot files, including grub.conf:

```
default=0
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz
title Red Hat Linux (2.4.18-3)
        root (hd0,0)
        kernel /vmlinuz-2.4.18-3 ro root=/dev/hda3
        initrd /initrd-2.4.18-3.img
```

- > “info grub”
  will invoke documentation for grub, including installation instructions and examples (info takes some getting used to)
- More on GRUB later (boot examples)
- GRUB supports ext2, ext3, xfs, reiserfs, fat, minix, jfs, vstafs, ffs

technical computing – invent. design. deliver.

Example LILO Configuration File

/boot/lilo.conf

```
boot      = /dev/fd0                    # Specify boot device
delay     = 10                          # Wait 10 seconds
message = bootmessage                   # Text prompt
read-only                               # Mount root RO

label = linux_up                        # Uniprocessor
        image = vmlinuz-2.2.12-20
        initrd = initrd-2.2.12-20.img
        root    = /dev/hda5
label = linux_smp                       # Multi-processor
        image=vmlinuz-2.2.12-20smp
        initrd = initrd-2.2.12-20smp.img
        root   = /dev/hda5
label = linux_old                       # Last week's kernel
        image = vmlinuz_old
        initrd = initrd-2.2.12-20old.img
        root   = /dev/hda5
```

technical computing – invent. design. deliver.

Example GRUB Configuration File

/boot/grub/grub.conf

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE:  You have a /boot partition.  This means that
#          all kernel and initrd paths are relative to /boot/, eg.
#          root (hd0,0)
#          kernel /vmlinuz-version ro root=/dev/hda3
#          initrd /initrd-version.img
#boot=/dev/hda
default=0
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz
title Red Hat Linux (2.4.18-3)
                root (hd0,0)
                kernel /vmlinuz-2.4.18-3 ro root=/dev/hda3
                initrd /initrd-2.4.18-3.img
```

technical computing – invent. design. deliver.

Booting the Kernel

- If properly configured, LILO will present you with a menu of kernels to boot
- Type Ctrl-X to get a text-mode boot prompt
- In text-mode, the TAB key will display possible boot targets
- LILO can boot Windows systems as well as Linux systems
- Run LILO “lilo –v –v –t “ to see all output and TEST the configuration, if something is wrong your system may not boot (time for the boot floppy you made…)
- Be CAREFUL when running LILO, you can inadvertently overwrite the MBR on multi-boot systems when all you want to do is change the BR in the /boot partition!

technical computing – invent. design. deliver.

More Lessons About LILO

- When you update a kernel, you must re-run LILO
- You should uninstall LILO before “decomissioning” a system, especially if LILO is installed in the MBR
- The default configuration file for LILO is ***/etc/lilo.conf***
- You can build a “mini” boot partition on a floppy, either by hand or using the “/sbin/mkbootdisk” command
- The LILO command may be run on a mounted disk by specifying the “-r <directory>” option, which will do a “chroot” to that directory
- Examine the “mkbootdisk” script for examples of what is needed to build a bootable floppy disk
- With the proper symbolic links and naming conventions, you can have a very flexible way to try new kernel configurations

technical computing – invent. design. deliver.

GRUB Boot Menu

technical computing – invent. design. deliver.

GRUB Boot Entries for Linux

technical computing – invent. design. deliver.

GRUB Editing (boot single user mode)

technical computing – invent. design. deliver.

GRUB Ready to Boot Single User

technical computing – invent. design. deliver.

Creating/Using GRUB Boot Floppy

- Being able to boot grub independently of the OS can sometimes save your system (especially multi-boot)
- To create a GRUB boot disk (will destroy floppy data!):
  - Insert floppy disk
  - "cd /usr/share/grub/i386-redhat"
  - "dd if=stage1 of=/dev/fd0 bs=512 count=1"
  - "dd if=stage2 of=/dev/fd0 bs=512 seek=1"
- You can then boot grub from the floppy and guess at the root device:
  - "root (hd0,0)"
  - "find /grub/grub.conf"
- GRUB will return the device it finds the file on, so you know that is /boot – set it to the root device
- "configfile /grub/grub.conf" will grab the menu from /boot/grub (and you may be off and running …)

technical computing – invent. design. deliver.

System Booting and Startup

technical computing – invent. design. deliver.

Passing Parameters to the Linux Kernel

- The Linux kernel (and modules), just like HP-UX, may be passed "command-line" parameters when they are started (remember the "hpux -is disk(;0)vmunix" command?)
- LILO or GRUB can pass the following parameters (and many more):
  - **read-only** — **mount the root read-only**
  - **vga=mode** — **normal(80x25), extended(80x50), ask**
  - **ramdisk=<size>** — **set size of initial RAM disk**
  - **root=<root_device>** — **device name or "current"**
  - **append=<string>** — **append <string> to options**
  - **literal=<string>** — **override ALL options with <string>**
- **LILO input: "<name> single" will boot into single-user mode**
- **See /usr/src/linux-2.4/Documentation/kernel-parameters.txt for a list of all (most?) kernel parameters**

technical computing – invent. design. deliver.

Modifying Kernel Parameters at Run Time

- The “sysctl” command may be used to set or examine the parameters that affect system behavior, either at boot or “real-time”
- “sysctl –a” — List all parameters
- “sysctl –w <parameter>=<value>” — Set a parameter
- “sysctl –p [<file>]” — Load from file or /etc/sysctl.conf
- Many drivers and modules also allow setting values through the /proc file system:

```
echo “213458” > /proc/sys/net/core/rmem_default
echo “546789” > /proc/sys/net/core/rmem_max
```

changes the default socket buffer memory pool size and maximum size, respectively, for all sockets on the system.

technical computing – invent. design. deliver.

Listing Kernel Parameters with “sysctl”

```
> sysctl –a

[…]
  net.core.rmem_default = 65535
  net.core.wmem_default = 65535
  net.core.rmem_max = 65535
  net.core.wmem_max = 65535
  vm.max-readahead = 127
  vm.min-readahead = 3
  vm.max_map_count = 65536
  vm.page-cluster = 3
  vm.pagetable_cache = 25          50
  vm.kswapd = 512          32      8
  vm.overcommit_memory = 0
  vm.bdflush = 40 0        0       0   500   3000    60      0       0
  vm.freepages = 1055      2304    3072
  kernel.overflowgid = 65534
  kernel.overflowuid = 65534

[…]
```

technical computing – invent. design. deliver.

Contents of the /boot Directory

System.map → System.map-2.4.18-3
System.map-2.4.18-3

vmlinuz → vmlinuz-2.4.18-3
vmlinuz-2.4.18-3

kernel.h

initrd.img → initrd-2.418-3.img
initrd-2.4.18-3.img

module-info → module-info-2.4.18-3
module-info-2.4.18-3

grub
- system.map
- e2fs_stage1_5
- fat_stage1_5
- ffs_stage1_5
- grub.conf
- jfs_stage1_5
- menu.lsg → ./grub.conf
- minix_stage1_5
- reiser_stage1_5
- spash.xpm.gz
- stage1
- stage2
- vstafs_stage1_5
- xfs_stage1_5

technical computing – invent. design. deliver.

HP-UX: UNIX for the Enterprise

Comparison of HP-UX and Linux SYSVR4 System Startup

- The world starts with “init”
- /etc/rc.config.d contains startup data
- /etc/rc script performs system startup
- /etc contains startup directories init.d and rc*.d
- /sbin/init.d contains scripts that are linked into /sbin/rc*.d
- Each run level has an associated directory that contains startup and shutdown links for each subsystem
- The directory for each intervening run-level is “executed” on run-level change

- The world starts with “init”
- /etc/sysconfig and scripts contain startup data
- /etc/rc, /etc/rc.local, and /etc/rc.sysinit perform startup
- /etc/init.d and /etc/rc*.d are linked into /etc/rc.d/
- /etc/rc.d/init.d contains scripts that are linked into /etc/rc*.d
- Each run level has an associated directory that contains startup and shutdown links for each subsystem
- Only the directory for the current run-level is “executed” when run-levels are changed

technical computing – invent. design. deliver.

Inittab and Run-Level Comparison

HP-UX: UNIX for the Enterprise

- HP-UX Init run levels
  - 0 halt
  - 1 Single-user mode
  - 2 Full multi-user
  - 3 X11
  - 4 Used?
- The “init:3:initdefault:” line in /etc/inittab controls the default level
- The “who –r” command returns the current value of init’s run level
- No shell information about run-level is available by default
- “init <level” will change the run level

- Linux Init run levels:
  - 0 halt
  - 1 Single-user mode
  - 2 Multi-user without NFS
  - 3 Full multi-user
  - 4 Unused (really?)
  - 5 X11
  - 6 Reboot
- The “id:5:initdefault:” line in /etc/inittab controls default level
- The “runlevel” command returns previous level and current level (N=none) “N 5”
- Run-level information is available in two shell environment variables: ${RUNLEVEL} and ${PREVLEVEL}
- “init <level>” will change the run level

technical computing – invent. design. deliver.

Enabling Linux Boot-Time Services

- Each file in /etc/init.d contains a special comment line that resembles:
  # chkconfig: 2345 80 30
  from the /etc/init.d/sendmail file.
- This comment tells the “chkconfig” command how to enable the associated service.
- “chkconfig” will create links in runlevels 2, 3, 4, and 5 with S80sendmail and K30sendmail (Start and Kill)
- “chkconfig sendmail on” will create the links but not start the service
- “chkconfig sendmail off” will remove the links but not stop the service
- “chkconfig –list” will show all services and their status, including services handled by xinetd
- “chkconfig –list sendmail” will show a status line like (on, off):

sendmail	0:off	1:off	2:on	3:off	4:off	5:off	6:off
sendmail	0:off	1:off	2:on	3:on	4:on	5:on	6:off

technical computing – invent. design. deliver.

Starting, Stopping, and Restarting Services

- The “service” command will start, stop, or restart a service.
- “service –status-all” will return the current status of all services (may take a while …)
- “service <service> stop” will stop the service
- “service <service> start” will start the service
  (Note: /var/run has files containing PIDs for running processes)
- “service <service> restart” will stop and then start the service
- Neat, huh? Wish that HP-UX did this? 8^)

technical computing – invent. design. deliver.

Hardware Configuration Troubleshooting

Output From Dmesg
Kudzu and Hardware Detection
Listing PCI Devices
The /proc Filesystem
XFree86 Logs
Network Interface Status

technical computing – invent. design. deliver.

Determining Hardware Configuration

- Linux supports quite a bit of common hardware, but not everything is -always- supported
- Whenever something does not get detected, the first step is to find out what is there
- For PCI cards and adapters, “lspci” will display devices
- For ISA cards, “pnpdump” will display plug-and-play devices on the system
- The “dmesg” command and /var/log/messages will display the hardware discovery process during boot
- The “kudzu” command is used at boot to discover new devices, you can also run it interactively (see /etc/sysconfig/hwconf)
- The /proc file system can also be a help in determining what Linux has and has not discovered (CPU, RAM, SCSI, USB, etc.)

technical computing – invent. design. deliver.

Output From /var/log/dmesg

(my laptop)

```
[…]
Serial driver version 5.05c (2001-07-08) with MANY_PORTS MULTIPORT SHARE_IRQ SERIAL_PCI ISAPNP enabled
ttyS00 at 0x03f8 (irq = 4) is a 16550A
Real Time Clock Driver v1.10e
block: 736 slots per queue, batch=184
Uniform Multi-Platform E-IDE driver Revision: 6.31
ide: Assuming 33MHz system bus speed for PIO modes; override with idebus=xx
PIIX4: IDE controller on PCI bus 00 dev 39
PIIX4: chipset revision 1
PIIX4: not 100% native mode: will probe irqs later
    ide0: BM-DMA at 0xfcf0-0xfcf7, BIOS settings: hda:DMA, hdb:pio
    ide1: BM-DMA at 0xfcf8-0xfcff, BIOS settings: hdc:DMA, hdd:pio
hda: IBM-DARA-212000, ATA DISK drive
hdc: TOSHIBA DVD-ROM SD-C2302, ATAPI CD/DVD-ROM drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
ide1 at 0x170-0x177,0x376 on irq 15
blk: queue c035e6a4, I/O limit 4095Mb (mask 0xffffffff)
hda: 23579136 sectors (12073 MB) w/418KiB Cache, CHS=1559/240/63, UDMA(33)
ide-floppy driver 0.99.newide
Partition check:
 hda: hda1 hda2 hda3
Floppy drive(s): fd0 is 1.44M
FDC 0 is a post-1991 82077
[…]
```

technical computing – invent. design. deliver.

Output From Kudzu: /etc/sysconfig/hwconf

(my laptop)

-
class: MOUSE
bus: PSAUX
detached: 0
device: psaux
driver: genericps/2
desc: "Generic Mouse (PS/2)"
-
class: AUDIO
bus: PCI
detached: 0
driver: maestro
desc: "ESS Technology|ES1978 Maestro 2E"
vendorId: 125d
deviceId: 1978
subVendorId: 103c
subDeviceId: 000a
pciType: 1
-

-
class: CDROM
bus: IDE
detached: 0
device: hdc
driver: ignore
desc: "TOSHIBA DVD-ROM SD-C2302"
-
class: VIDEO
bus: PCI
detached: 0
driver: Card:ATI Rage Mobility
desc: "ATI|Rage Mobility P/M AGP 2x"
vendorId: 1002
deviceId: 4c4d
subVendorId: 103c
subDeviceId: 000a
pciType: 1

technical computing – invent. design. deliver.

Output From “lspci”

(my laptop)

00:00.0 Host bridge: Intel Corporation 440BX/ZX - 82443BX/ZX Host bridge (rev 03)

00:01.0 PCI bridge: Intel Corporation 440BX/ZX - 82443BX/ZX AGP bridge (rev 03)

00:04.0 CardBus bridge: Texas Instruments PCI1225 (rev 01)

00:04.1 CardBus bridge: Texas Instruments PCI1225 (rev 01)

00:07.0 Bridge: Intel Corporation 82371AB PIIX4 ISA (rev 02)

00:07.1 IDE interface: Intel Corporation 82371AB PIIX4 IDE (rev 01)

00:07.2 USB Controller: Intel Corporation 82371AB PIIX4 USB (rev 01)

00:07.3 Bridge: Intel Corporation 82371AB PIIX4 ACPI (rev 03)

00:08.0 Multimedia audio controller: ESS Technology ES1978 Maestro 2E (rev 10)

00:11.0 PCI bridge: Intel Corporation 82380FB (rev 01)

01:00.0 VGA compatible controller: ATI Technologies Inc Rage Mobility P/M AGP 2x (rev 64)

02:07.0 Ethernet controller: 3Com Corporation 3cSOHO100-TX Hurricane (rev 30)

02:08.0 CardBus bridge: Texas Instruments PCI1220 (rev 02)

02:08.1 CardBus bridge: Texas Instruments PCI1220 (rev 02)

02:0c.0 IDE interface: CMD Technology Inc PCI0646 (rev 07)

02:0d.0 SCSI storage controller: Symbios Logic Inc. (formerly NCR) 53c810 (rev 23)

technical computing – invent. design. deliver.

The /proc Filesystem

(my laptop)

$ cat /proc/meminfo

```
   total:    used:    free:  shared: buffers:  cached:
   Mem:  394321920 382668800 11653120        0  6033408 285745152
   Swap: 805101568  5754880 799346688
   MemTotal:       385080 kB
   […]
```

$ cat /proc/interrupts

```
    CPU0
    0:    3033373        XT-PIC  timer
    1:      33173        XT-PIC  keyboard
    2:          0        XT-PIC  cascade
    3:          2        XT-PIC  serial
    4:          6        XT-PIC  serial
    8:    4530290        XT-PIC  rtc
   10:      47720        XT-PIC  usb-uhci, Texas Instruments PCI1225, Texas Instruments PCI1225 (#2),
   eth0, ESS Maestro 2E
   12:     222483        XT-PIC  PS/2 Mouse
   14:     124376        XT-PIC  ide0
   15:     236857        XT-PIC  ide1
   NMI:         0
   ERR:         3
```

technical computing – invent. design. deliver.

XFree86 Logs

(my laptop)

```
Module Loader present

Markers: (--) probed, (**) from config file, (==) default setting,
         (++) from command line, (!!) notice, (II) informational,
         (WW) warning, (EE) error, (NI) not implemented, (??) unknown.
  (==) Log file: "/var/log/XFree86.0.log", Time: Wed Jul 24 11:39:11 2002
  (==) Using config file: "/etc/X11/XF86Config-4“
  (==) ServerLayout "XFree86 Configured“
  (**) |-->Screen "Screen0" (0)
  (**) |   |-->Monitor "Generic High Frequency SVGA, 1024x768 @ 70 Hz“
  (**) |   |-->Device "ATI Rage 128"
  (**) |-->Input Device "Mouse0"
  (**) |-->Input Device "Keyboard0"
  (**) Option "XkbLayout" "us“
  (**) XKB: layout: "us“
  (==) Keyboard: CustomKeycode disabled
  (**) FontPath set to "unix/:7100"
  (==) RgbPath set to "/usr/X11R6/lib/X11/rgb“
  (==) ModulePath set to "/usr/X11R6/lib/modules“
  (--) using VT number 7

[…]
```

technical computing – invent. design. deliver.

Network Interface Status

(my laptop)

```
# mii-tool -v eth0
     eth0: negotiated 100baseTx-FD, link ok
     product info: TDK 78Q2120 rev 3
     basic mode:   autonegotiation enabled
     basic status: autonegotiation complete, link ok
     capabilities: 100baseTx-FD 100baseTx-HD 10baseT-FD 10baseT-HD
     advertising:  100baseTx-FD 100baseTx-HD 10baseT-FD 10baseT-HD
     link partner: 100baseTx-FD 100baseTx-HD 10baseT-FD 10baseT-HD flow-control
```

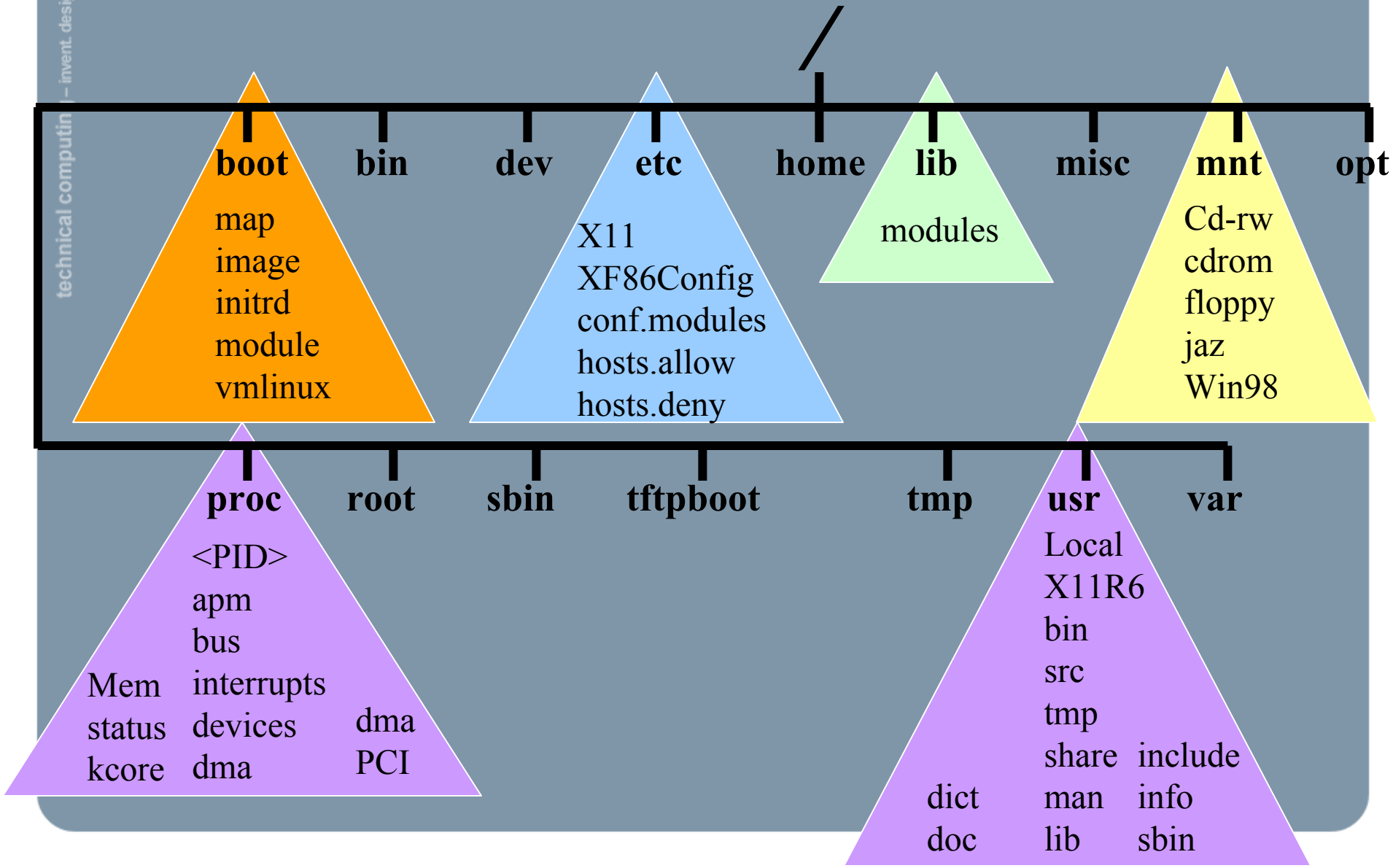
```
# cat /etc/sysconfig/network
   NETWORKING=yes
   GATEWAYDEV="eth0“
   DHCPCDARGS="-H -D -d -c /etc/firewall/gShield.rc“
   FORWARD_IPV4="no“
   HOSTNAME="gandolf.nsr.hp.com“
```

```
# cat /etc/sysconfig/network-scripts/ifcfg-eth0
   DEVICE="eth0“
   ONBOOT="yes“
   BOOTPROTO="dhcp“
   IPADDR=""“
   NETMASK=""
```

technical computing – invent. design. deliver.

File System Layout

Basic Linux File-system Structure



technical computing – invent. design. deliver.

Example Device Files

(My home server system)

Description	*Use*	*Device Name*
IDE Drive 1	Win98/Linux	/dev/hda
Windows partition	Windows 98	/dev/hda1
Linux partition	/boot	/dev/hda2
NEC CD-ROM stacker		/dev/hdc
HP R/W CD-ROM		/dev/hdd
Floppy drive		/dev/fd0
SCSI address 0	swap	/dev/sda1
SCSI address 0	Linux /	/dev/sda5
SCSI address 1	Linux /aux	/dev/sdb1
SCSI address 1	swap	/dev/sdb5

technical computing – invent. design. deliver.

Example Device Files
(Continued)

Description	*Use*	*Device Name*
SCSI address 2	Linux /vmdata1	/dev/sdc1
SCSI address 2	swap	/dev/sdc5
SCSI address 3	Linux /vmdata2	/dev/sdd1
SCSI address 3	swap	/dev/sdd5
SCSI address 4	HP Photo scanner	/dev/sge
SCSI address 5	HP 6200C scanner	/dev/sgf
SCSI address 6	JAZ 1 GB drive	/dev/sde4
COM1	V.90 modem	/dev/ttyS0
Parallel port	HP P1000 printer	/dev/lp0

technical computing – invent. design. deliver.

Example /etc/fstab File for Example Hardware

/dev/sda5	/	ext2	defaults	1	1
/dev/hda2	/boot	ext2	defaults	1	2
/dev/sdb1	/aux1	ext2	defaults	1	2
/dev/sdc1	/vmdata1	ext2	defaults	1	2
/dev/sdd1	/vmdata2	ext2	defaults	1	2
/dev/sda1	swap	swap	pri=1	0	0
/dev/sdb5	swap	swap	pri=1	0	0
/dev/sdc5	swap	swap	pri=1	0	0
/dev/sdd5	swap	swap	pri=1	0	0
/dev/fd0	/mnt/floppy	ext2	owner,noauto	0	0
/dev/cdrom	/mnt/cdrom	iso9660	owner,noauto,ro	0	0
/dev/hdd	/mnt/cd-rw	iso9660	noauto,ro	0	0
/dev/sde4	/mnt/jaz	vfat	fat=16	0	0
/dev/hda1	/mnt/Win98	vfat	fat=32	0	0
none	/proc	proc	defaults	0	0
none	/dev/pts	devpts	gid=5,mode=620	0	0

technical computing – invent. design. deliver.

Interesting File-system Side Trips

- /etc/pam.d — directory containing PAM config
- /etc/profile.d — directory containing shell profiles
- /etc/logrotate.d — directory controlling log rotation
  - /usr/sbin/logrotate command
  - /etc/logrotate.conf
- /etc/rc.d — directory containing startup scripts
- /proc — directory containing system/process information
  - /proc/bus/usb — directory containing usb device information
  - /proc/bus/pci — directory containing pci device information

technical computing – invent. design. deliver.

Example /proc File System Contents

```
1/     1402/  1475/  1587/  1756/  1781/  1792/  4/    763/  bus/         ide/         mdstat      slabinfo
1014/  1438/  1488/  1589/  1757/  1782/  1793/  5/    8/    cmdline      interrupts   meminfo     stat
1056/  1461/  1543/  1591/  1758/  1783/  1794/  536/  846/  cpuinfo      iomem        misc        swaps
1078/  1462/  1557/  1592/  1760/  1784/  1825/  541/  879/  devices      ioports      modules     sys/
1099/  1463/  1569/  1594/  1762/  1785/  1828/  561/  90/   dma          irq/         mounts@     sysvipc/
1167/  1464/  1571/  1595/  1764/  1786/  183/   590/  900/  driver/      kcore        mtrr        tty/
12/    1465/  1579/  1598/  1766/  1787/  1971/  6/    950/  execdomains  kmsg         net/        uptime
1241/  1466/  1581/  1599/  1768/  1789/  2/     7/    968/  fb           ksyms        partitions  version
1354/  1467/  1583/  1749/  1770/  1790/  2014/  702/  986/  filesystems  loadavg      pci         vmnet/
1371/  1474/  1585/  1755/  1772/  1791/  3/     722/  apm   fs/          locks        self@
```

PROC(5) **Linux Programmer's Manual** **PROC(5)**

NAME

proc - process information pseudo-filesystem

DESCRIPTION

/proc is a pseudo-filesystem which is used as an interface to kernel data structures rather than reading and interpreting /dev/kmem. Most of it is read-only, but some files allow kernel variables to be changed.

technical computing – invent. design. deliver.

Example /proc/1 (PID 1) File System Contents

```
-r--r--r--    1 root     root            0 Jul 29 22:46 cmdline
lrwxrwxrwx    1 root     root            0 Jul 29 22:46 cwd -> /
-r--------    1 root     root            0 Jul 29 22:46 environ
lrwxrwxrwx    1 root     root            0 Jul 29 22:46 exe -> /sbin/init
dr-x------    2 root     root            0 Jul 29 22:46 fd
-r--r--r--    1 root     root            0 Jul 29 22:46 maps
-rw-------    1 root     root            0 Jul 29 22:46 mem
-r--r--r--    1 root     root            0 Jul 29 22:46 mounts
lrwxrwxrwx    1 root     root            0 Jul 29 22:46 root -> /
-r--r--r--    1 root     root            0 Jul 29 22:46 stat
-r--r--r--    1 root     root            0 Jul 29 22:46 statm
-r--r--r--    1 root     root            0 Jul 29 22:46 status
```

technical computing – invent. design. deliver.

An Interesting Redhat-ism

- **/etc/pam.d** — **directory containing PAM config**
  /etc/profile.d — **directory containing shell profiles**
  /etc/logrotate.d — **directory controlling log rotation**
  /etc/rc.d — **directory containing runlevel definitions**
  /etc/init.d — **directory containing service scripts**
  /etc/xinetd.d — **directory containing inetd service definitions**
- **These directories make it easy for packages to install and remove themselves from the system without affecting other packages**
- **Typical service installation might:**
  - Add service script to /etc/init.d
  - Run chkconfig to create links in /etc/rc*.d
  - Add service information to /etc/xinetd.d
  - Add log processing to /etc/logrotate.d
- This is *very* manageable once you catch the paradigm!

technical computing – invent. design. deliver.

Software Installation and Update

technical computing – invent. design. deliver.

The RedHat Package Manager (RPM)

- Installation and update of software on Linux (at least the RedHat distributions) is done via the “rpm” command
- Packages contain binaries, configuration information and scripts, and PGP signatures for verification
- The system keeps a database of all installed packages, their revisions, and their dependencies
- To find out what packages are installed, “rpm –qa”
- To install and/or update a package,
  “rpm –Uvh <package>”
- The kernel, libraries, and applications may be updated LIVE! (including glibc)

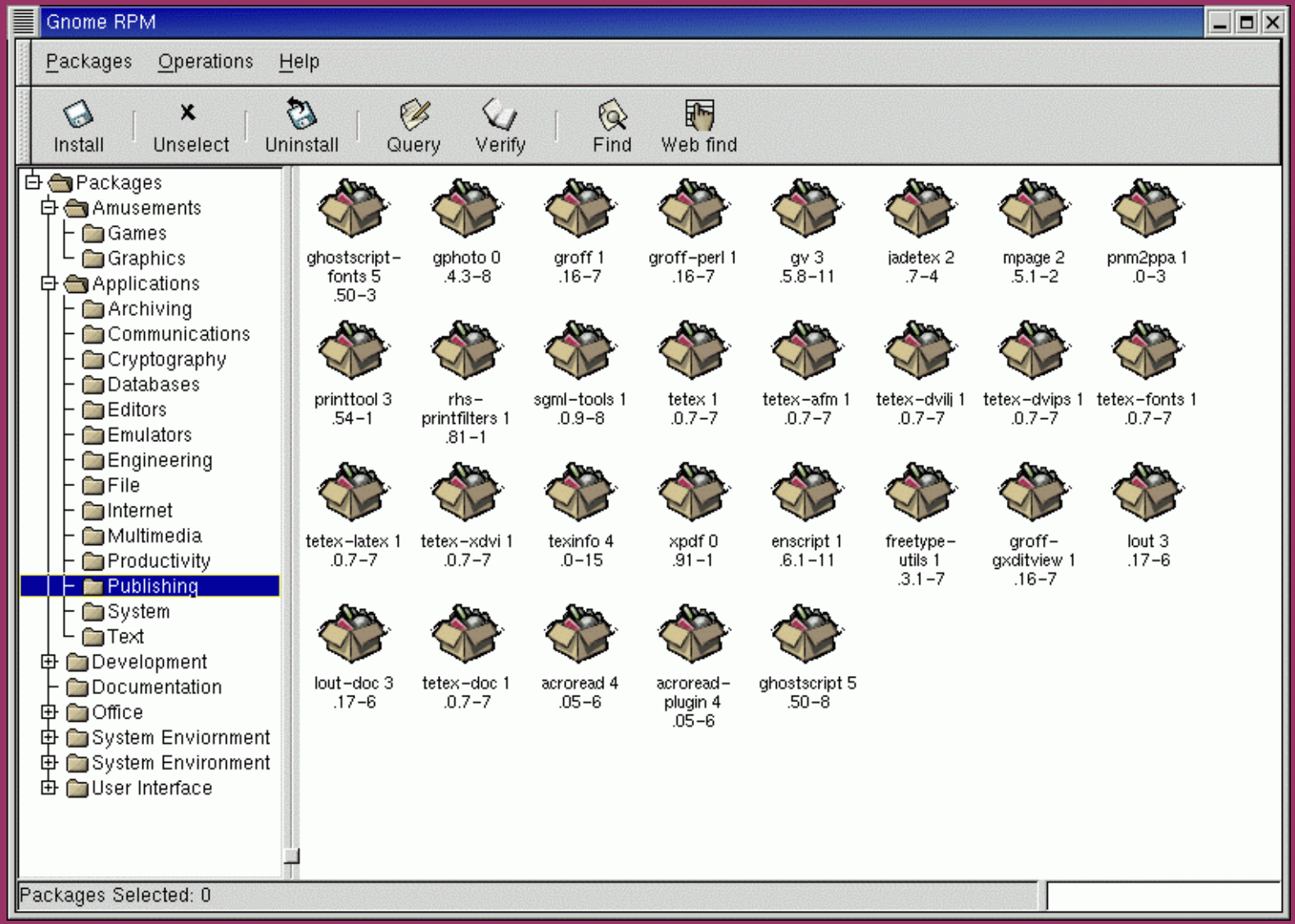
technical computing – invent. design. deliver.

The RedHat Package Manager (RPM)

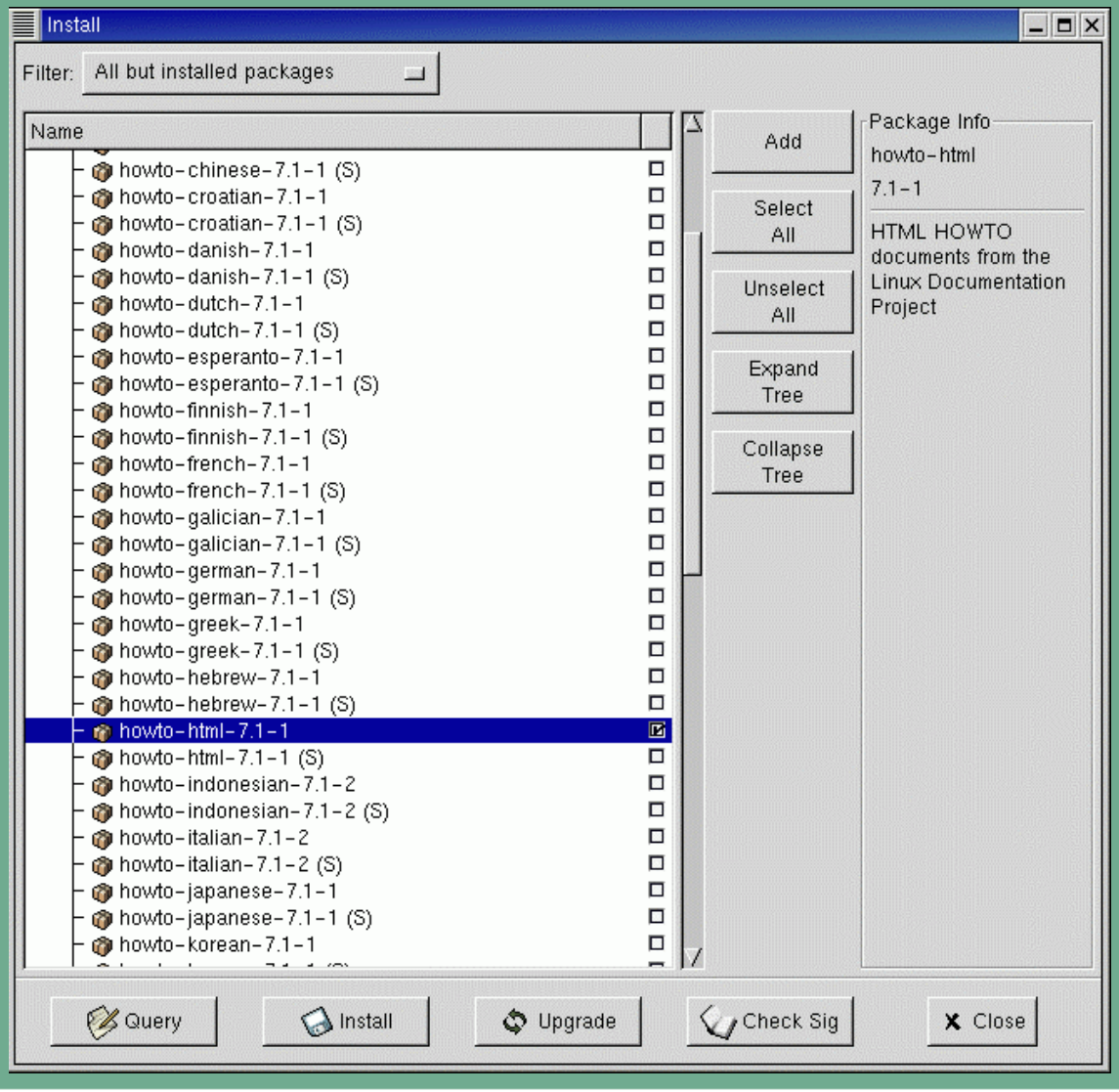
```
# rpm -q --whatprovides /bin/ls
        fileutils-4.1-10

# rpm -qa | grep real
        ethereal-0.9.4-0.7.3.0
        ethereal-gnome-0.9.4-0.7.3.0

# rpm --checksig RealPlayer-8.0-1.i386.rpm
        RealPlayer-8.0-1.i386.rpm: md5 OK

# rpm -q --filesbypkg ethtool
        ethtool         /usr/sbin/ethtool
        ethtool         /usr/share/doc/ethtool-1.5
        ethtool         /usr/share/doc/ethtool-1.5/AUTHORS
        ethtool         /usr/share/doc/ethtool-1.5/COPYING
        ethtool         /usr/share/doc/ethtool-1.5/ChangeLog
        ethtool         /usr/share/doc/ethtool-1.5/INSTALL
        ethtool         /usr/share/doc/ethtool-1.5/NEWS
        ethtool         /usr/share/doc/ethtool-1.5/README
        ethtool         /usr/share/man/man8/ethtool.8.gz
```

technical computing – invent. design. deliver.

The GNOME Interface to RPM, “gnorpm”



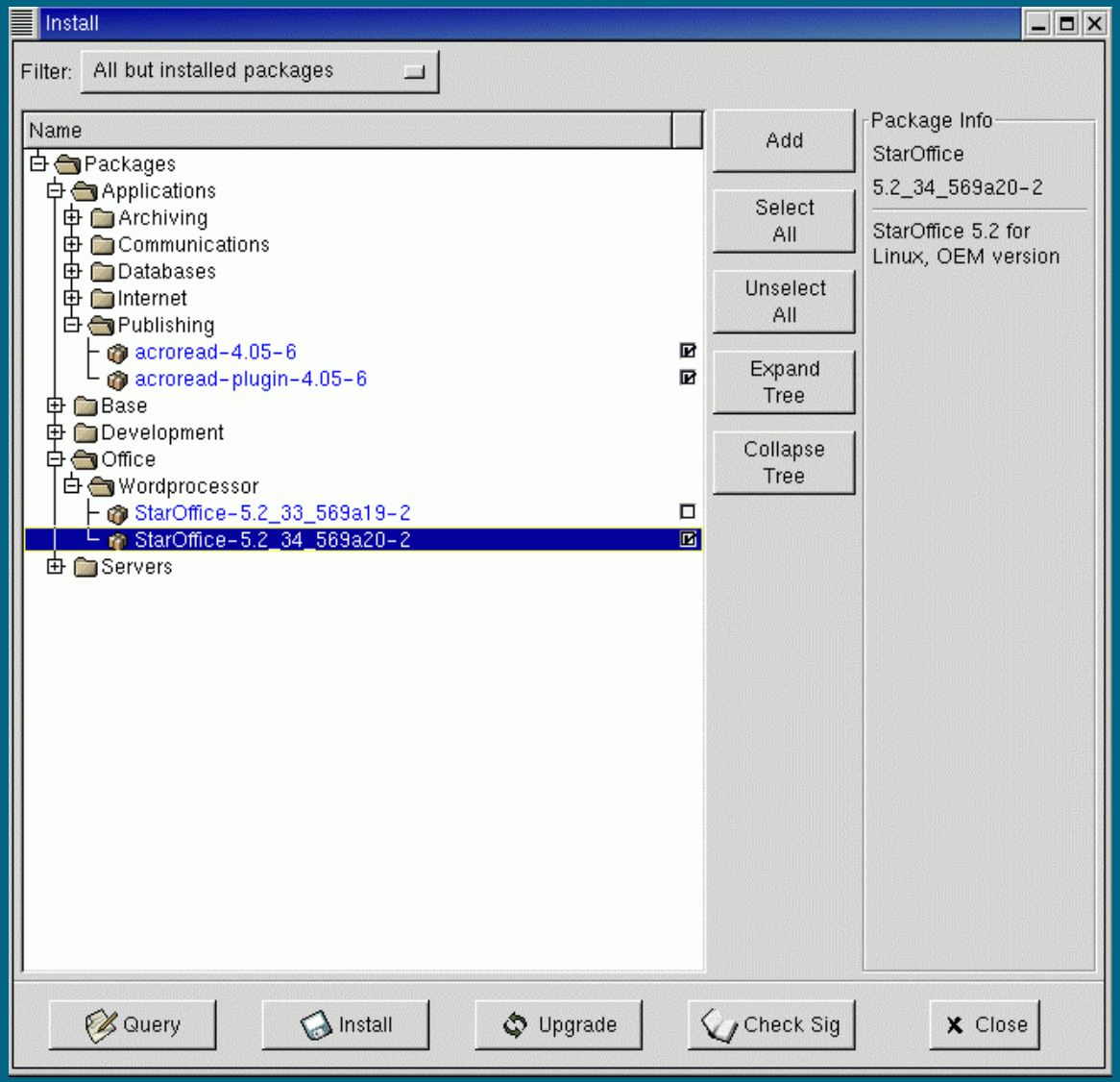
Installation With “gnorpm”



technical computing – invent. design. deliver.

technical computing – invent. design. deliver.

Update With “gnorpm”



technical computing – invent. design. deliver.

RedHat Update Manager, “up2date”

- RedHat provides an automated way of keeping your system packages up to date, the “up2date” command
- For 90 days after purchase, or if you buy support, you may connect to the RedHat FTP server with this tool
- All updated packages are presented in a graphical interface, and you may select the ones you want
- The packages are downloaded and installed
- Packages are stored in /var/spool/up2date if you select the “keep after installation” option and are removed otherwise
- “up2date –update” will download all matching packages and update your system

technical computing – invent. design. deliver.

Interesting Software for Linux

- JAVA at http://www.blackdown.org
- StarOffice from Sun Microsystems on RedHat Deluxe Workstation CD-ROM
- Real Player on RedHat Deluxe Workstation CD-ROM
- Vmware at http://www.vmware.com
- Netscape (4.76 and 6.0) at http://www.netscape.com/netscape
- Iomega drivers at http://www.iomega.com
- XV X-windows image viewer at http://www.trilon.com/xv/xv.html
- Other applications at http://freshmeat.net

technical computing – invent. design. deliver.

General System Administration Tasks

technical computing – invent. design. deliver.

Linux Password Management

- Shadow password file is the default, no passwords are available to casual users
  - Too much password data kept to go into here …
  - rob:x:1000:100:/home/rob:/bin/bash
  - chage — Change user expiration information
  - pwck — Check password integrity
  - grpck — Check group integrity
- /etc/passwd /etc/passwd- — /etc/group /etc/group-
- /etc/shadow — (mode 400, root:root)
- /etc/gshadow — (mode 400, root:root)
- passwd -> pwconv -> shadow
- shadow -> pwunconv -> passwd
- group -> grconv -> gshadow
- gshadow -> grunconv -> group

technical computing – invent. design. deliver.

Linux Security

- Linux, by default is set up with more security features enabled than HP-UX “out of the box”
- You will not be able to log in as “root” except at the console
- Telnet in as a “normal” user, then “su -” to “root”, but you must HAVE a user configured besides “root” … Remember this at system installation time!
- Check out /etc/hosts.allow and /etc/hosts.deny to configure machines that can access your Linux box
- Check out /etc/xinetd.d for configuration files for FTP, telnet, and other services
- For NFS to function, any firewall must permit access to the portmap service
- FTP is disabled by default

technical computing – invent. design. deliver.

Linux Security Ipchains/Iptables

- Linux has two TCP/IP firewall services, Iptables and Ipchains, named after the commands that control the rule sets
- The installation process allows you to choose a firewall configuration, but uses the older Ipchains
- Iptables is much more flexible
- I use a firewall generation tool named “gShield” that builds rule sets for Iptables – answer some questions and you’re off and dropping
- The gShield firewall is updated each time the DHCP client daemon, “dhcpcd” is run
- Note that you must use either Iptables or Ipchains, but not both -- they are mutually exclusive (and you will get error messages)
- The “Linux Firewalls” book explains the operation of both services, with a concentration on the newer Iptables

technical computing – invent. design. deliver.

Linux Security Ipchains/Iptables

- “service ipchains stop”
- “chkconfig ipchains off”
- “chkconfig iptables on”
- “service iptables start”
- You are now ready to define firewall rules with the “ipchains” command
- See next slide for example fragment.

technical computing – invent. design. deliver.

Linux Iptables Command Example

```
#!/bin/bash

   IPTABLES=$( which iptables )
   # Create a new rule chain to handle key rotation requests that
   # come in on port 500 udp/tcp (isakmp) from the VPN system
   # Log any packets that are either accepted or rejected.
   ${IPTABLES} -N ISAKMP
   ${IPTABLES} -F ISAKMP
   ${IPTABLES} -A ISAKMP --source ${NAIATLGW1} -j ACCEPTnLOG
   ${IPTABLES} -A ISAKMP --source ${NAIATLGW2} -j ACCEPTnLOG
   ${IPTABLES} -A ISAKMP                      -j DROPnLOG
   ${IPTABLES} -N IPSEC
   ${IPTABLES} -F IPSEC
   ${IPTABLES} -A IPSEC  --source ${NAIATLGW1} -j ACCEPTnLOG
   ${IPTABLES} -A IPSEC  --source ${NAIATLGW2} -j ACCEPTnLOG
   ${IPTABLES} -A IPSEC                       -j DROPnLOG
   # Insert rules in the INPUT chain to intercept port 500 (isakmp)
   # key rotation packets and port 50/51 IPSEC packets
   ${IPTABLES} -I INPUT 21 --protocol tcp --dport isakmp -j ISAKMP
   ${IPTABLES} -I INPUT 21 --protocol udp --dport isakmp -j ISAKMP
   ${IPTABLES} -I INPUT 21 --protocol tcp --dport 50 -j ISAKMP
   ${IPTABLES} -I INPUT 21 --protocol tcp --dport 51 -j ISAKMP
   ${IPTABLES} -I INPUT 21 --protocol udp --dport 50 -j ISAKMP
   ${IPTABLES} -I INPUT 21 --protocol udp --dport 51 -j ISAKMP
   ###
   ###
```

technical computing – invent. design. deliver.

Linux Security

- The “inetd” process is replaced by “xinetd”
- The configuration file for xinetd is in /etc/xinetd.conf
- The /etc/xinetd.d directory (specified in xinetd.conf) contains individual files that enable or disable services.

```
# default: on
# description: The telnet server serves telnet sessions; it uses
# unencrypted username/password pairs for authentication.
service telnet
{
        flags           = REUSE
        socket_type = stream
        wait            = no
        user            = root
        server          = /usr/sbin/in.telnetd
        log_on_failure  += USERID
}
```

technical computing – invent. design. deliver.

Making Boot Disks Manually

(if you foolishly skipped the install step)

- Boot floppies can save your system if something goes wrong.
- Use the “mkbootdisk” command to create a bootable floppy from your kernel and modules.
- The boot floppy will enable you to mount the root directory in single use mode and possibly repair problems.
- An example command to make a boot floppy for my laptop is:

```
mkbootdisk –device /dev/fd0 2.4.18-3
```

technical computing – invent. design. deliver.

Examining the “initrd.img” File

- The initrd.img file contains a file system image with modules needed by the kernel to access the root file system (this solves a chicken and egg problem)
- The initrd.img file might contain SCSI modules, RAID modules, ext3 journaling modules, or any other module essential to booting (I.e. not built into the kernel)
- The initrd.img file also contains a linuxrc file to load modules for the kernel
- The initrd.img may be examined by:
  - “zcat /boot/initrd-2.4.18-3.img > /tmp/myimg”
  - “mkdir /tmp/image”
  - “mount –o loop /tmp/myimg /tmp/image”
  - “cd /tmp/image”

technical computing – invent. design. deliver.

Examining the “initrd.img” File

- The initrd.img file system contains:

  bin/ dev/ etc/ lib/ linuxrc* loopfs/ proc/ sbin@ sysroot/

- Under the lib directory, my initrd.img contains the “jbd.o” and “ext3.o” modules
- The “linuxrc” script contains:

```
#!/bin/nash
echo "Loading jbd module“
insmod /lib/jbd.o
echo "Loading ext3 module“
insmod /lib/ext3.o
echo Mounting /proc filesystem
mount -t proc /proc /proc
echo Creating root device
mkrootdev /dev/root
echo 0x0100 > /proc/sys/kernel/real-root-dev
echo Mounting root filesystem
mount --ro -t ext3 /dev/root /sysroot
umount /proc
pivot_root /sysroot /sysroot/initrd
```

technical computing – invent. design. deliver.

Building Device Files

- Linux comes pre-configured with “standard” device files, but sometimes more are needed
- To build device files, use “/dev/MAKEDEV”, which will handle major and minor numbers properly

• ttys{0-63}	TTY devices
• fd{0-7}	Floppy disks
• loop	Loopback devices
• sd[a-z]	SCSI disks
• sg[a-h], sg[0-7]	Generic SCSI devices
• Etc.	See man page for MAKEDEV

technical computing – invent. design. deliver.

Miscellaneous System Administration Tasks

- Make initial RAM-disk images for your system with the "mkinitrd –fv initrd-<kernel_version>.img <kernel_version>" command
- Examine DHCP parameters in the /etc/dhcpcd directory
  /etc/dhcpcd/dhcpcd-eth0.info
- Using "ls –color=auto" in a shell alias will cause the ls command to display files in color, based on their mode and type
- You can cut and paste from text-mode and VGA graphics windows
- Disk striping is easy with configuration data in /etc/raidtab and the "mkraid" command. You can select a number of RAID modes and behaviors that are automatically enabled at system boot. The "raidstart" and "raidstop" commands enable and disable the multiple disk (MD) device. Once the devices are configured, you may build the file system of your choice.
- Don't forget SAMBA and SWAT for exporting SMB/CIFS file systems!
- Linux can mount lots of different file systems and even has LVM!

hnical computing – invent. design. deliver.

Building The Linux Kernel

technical computing – invent. design. deliver.

Building The Linux Kernel

- Note: Redhat Linux is compiled for maximum compatibility, for an i386 processor! Later distributions *may* have CPU optimizations available
- To build the kernel:
  - cd /usr/src/linux2.4 (a link to the current version)
  - Build the configurator:
    - make xconfig (X-windows configurator tool)
    - make menuconfig (VGA mode configurator tool)
  - Save the current configuration to a file!
    - The current configuration is kept in /usr/src/linux2.4/.config
    - The default Redhat configurations are kept in /usr/src/linux/configs
  - Make changes using the configurator
  - Save new configuration to a file!
  - Build the kernel (Be prepared for a wait …)
    - make deps (build dependencies)
    - make bzImage (a compressed kernel)
    - make modules (if you are using them)

technical computing – invent. design. deliver.

Building the Linux Kernel

(continued)

- Back up the current module information!
- Backup the current kernel and system files!
- Always make a boot disk! (/sbin/mkbootdisk)
- make modules_install (install new modules)
- Move kernel and other system files from /usr/src/linux/arch/i386/boot, this will be called “bzImage”
  - Can use /sbin/installkernel or /sbin/new-kernel-package
- Run LILO to update map information (offset into partition for kernel file) in boot record (not necessary with GRUB, just modify the /boot/grub/grub.conf file)
- Reboot and test

- Naming conventions are important
  - Add a “test” label to the boot configuration file?
  - With links and “generic” boot configuration labels, you can implement a relatively fail-safe way of installing new kernels

technical computing – invent. design. deliver.

Building The Linux Kernel

(continued)

- Tip: The /usr/src/linux2.4/Makefile contains lines like the following:

  VERSION = 2
  PATCHLEVEL = 4
  SUBLEVEL = 18
  EXTRAVERSION = -3

- To keep from screwing up your current installation when building a new kernel and associated modules, change the EXTRAVERSION variable to something like

  "EXTRAVERSION=-3custom"

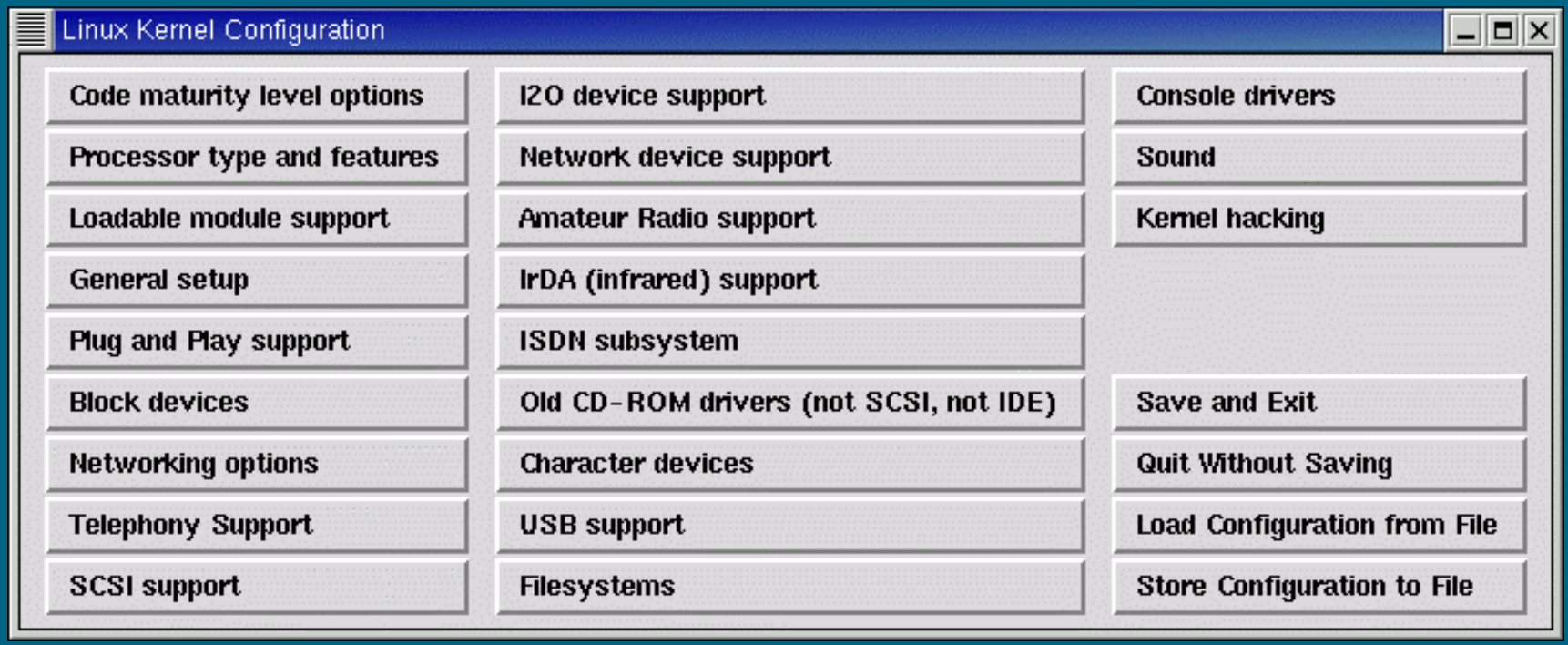
- This will allow you to keep your modules, initrd, and kernel separate from your working version (/usr/lib/modules-2.4.18-3custom, /boot/vmlinux-2.4.18-3custom, /boot/initrd-2.4.18-3custom, etc.)

- Tip: If you have more than one CPU, you can type "make –j <N> bzimage modules" to run N simultaneous compile jobs

technical computing – invent. design. deliver.

Linux Kernel Configuration Tool

- "cd /usr/src/linux2.4"; make xconfig
- Remember to load/save the configuration changes if you want to keep them across rebuilds



Linux Kernel Configuration Tool

technical computing – invent. design. deliver.

hnical computing – invent. design. deliver.

Dynamic Kernel Modules

technical computing – invent. design. deliver.

Linux Dynamic Kernel Modules

- When configuring a kernel, you have the choice to build things in (monolithic) or to dynamically load modules (slower) but much easier ...
- The “lsmod” command lists all loaded modules and their current state
- The startup operation (kudzu) will detect hardware and load the proper module (usually) for any *supported* hardware
- At boot time, the startup process runs “depmod -a”, which builds a dependency list for all modules
- The /etc/modules.conf file contains commands for the module commands “insmod” and “modprobe”
- Unused modules are unloaded by a cron job which executes “rmmod -a”
- Modules are located under “/lib/modules/<kernel_rev>”
- Modules are generally *not* compatible across kernel revisions!
- The kernel is smart enough to locate the proper modules to match its version, provided they are in the standard location (/lib/modules)

technical computing – invent. design. deliver.

Linux Dynamic Kernel Modules Commands

```
# lsmod
Module                  Size  Used by    Tainted: PF
maestro                30144   1  (autoclean)
soundcore               6692   2  (autoclean) [maestro]
vmnet                  23616   8
parport_pc             18724   0
parport                34208   0  [parport_pc]
vmmon                  22836   6
ipt_TOS                 1952  17  (autoclean)
ipt_state               1536   2  (autoclean)
ipt_REJECT              4096   8  (autoclean)
ipt_LOG                 4736   9  (autoclean)
ipt_limit               1952   3  (autoclean)
iptable_nat            21012   0  (autoclean) (unused)
ip_conntrack           21164   2  (autoclean) [ipt_state iptable_nat]
iptable_mangle          3136   1  (autoclean)
iptable_filter          2752   1  (autoclean)
ip_tables              13984  10  [ipt_TOS ipt_state ipt_REJECT ipt_LOG ipt_limit iptable_nat iptable_mangle iptable_filter]
serial_cs               5344   0  (unused)
3c59x                  28520   1
ds                      8608   2  [serial_cs]
yenta_socket           12384   2
pcmcia_core            50752   0  [serial_cs ds yenta_socket]
ide-cd                 30272   1  (autoclean)
cdrom                  32192   0  (autoclean) [ide-cd]
usb-uhci               24484   0  (unused)
usbcore                73152   1  [usb-uhci]
ext3                   67136   2
jbd                    49400   2  [ext3]
```

technical computing – invent. design. deliver.

/etc/modules.conf

cat /etc/modules.conf

```
alias                parport_lowlevel        parport_pc
alias                sound-slot-0            maestro
post-install         sound-slot-0            /bin/aumix-minimal -f /etc/.aumixrc -L >/dev/null 2>&1 || :
pre-remove           sound-slot-0            /bin/aumix-minimal -f /etc/.aumixrc -S >/dev/null 2>&1 || :
alias                usb-controller          usb-uhci
alias                eth0                    3c59x
```

ls –al /lib/modules/2.4.18-3

```
total 300
  drwxr-xr-x    5 root     root         4096   Jul  29 17:02 .
  drwxr-xr-x    3 root     root         4096   Jul  29 16:58 ..
lrwxrwxrwx    1 root     root           31 May 19 23:25 build -> ../../../usr/src/linux-2.4.18-3
  drwxr-xr-x    8 root     root         4096 May 19 23:25 kernel
  drwxr-xr-x    2 root     root         4096 Jun  27 04:51 misc
    -rw-r--r--    1 root     root        89026 Jul  15 09:15 modules.dep
    -rw-r--r--    1 root     root           31 Jul  15 09:15 modules.generic_string
    -rw-r--r--    1 root     root          147 Jul  15 09:15 modules.ieee1394map
    -rw-r--r--    1 root     root         8257 Jul  15 09:15 modules.isapnpmap
    -rw-r--r--    1 root     root           29 Jul  15 09:15 modules.parportmap
    -rw-r--r--    1 root     root        60859 Jul  15 09:15 modules.pcimap
    -rw-r--r--    1 root     root           24 Jul  15 09:15 modules.pnpbiosmap
    -rw-r--r--    1 root     root        91181 Jul  15 09:15 modules.usbmap
 drwxr-xr-x    2 root     root         4096 May 19 23:25 pcmcia
```

technical computing – invent. design. deliver.

Other Dynamic Module Commands

- “ksyms” List exported module symbols
- “insmod” install module (low level)
- “modprobe” install module and dependencies (high level)
- “rmmod” remove module
- “depmod” create module dependencies
- “lsmod” list installed modules

technical computing – invent. design. deliver.

HP-UX and Linux Commands

Basic HP-UX and Linux Command Differences

• swapinfo	swapon -s
• chown root:root /root_home	chown root.root /dev/fd0
• rm	rm -f (defaults to “safe”)
• ll	alias ll=‘ls -al’
• bdf	df
• lanscan	ifconfig

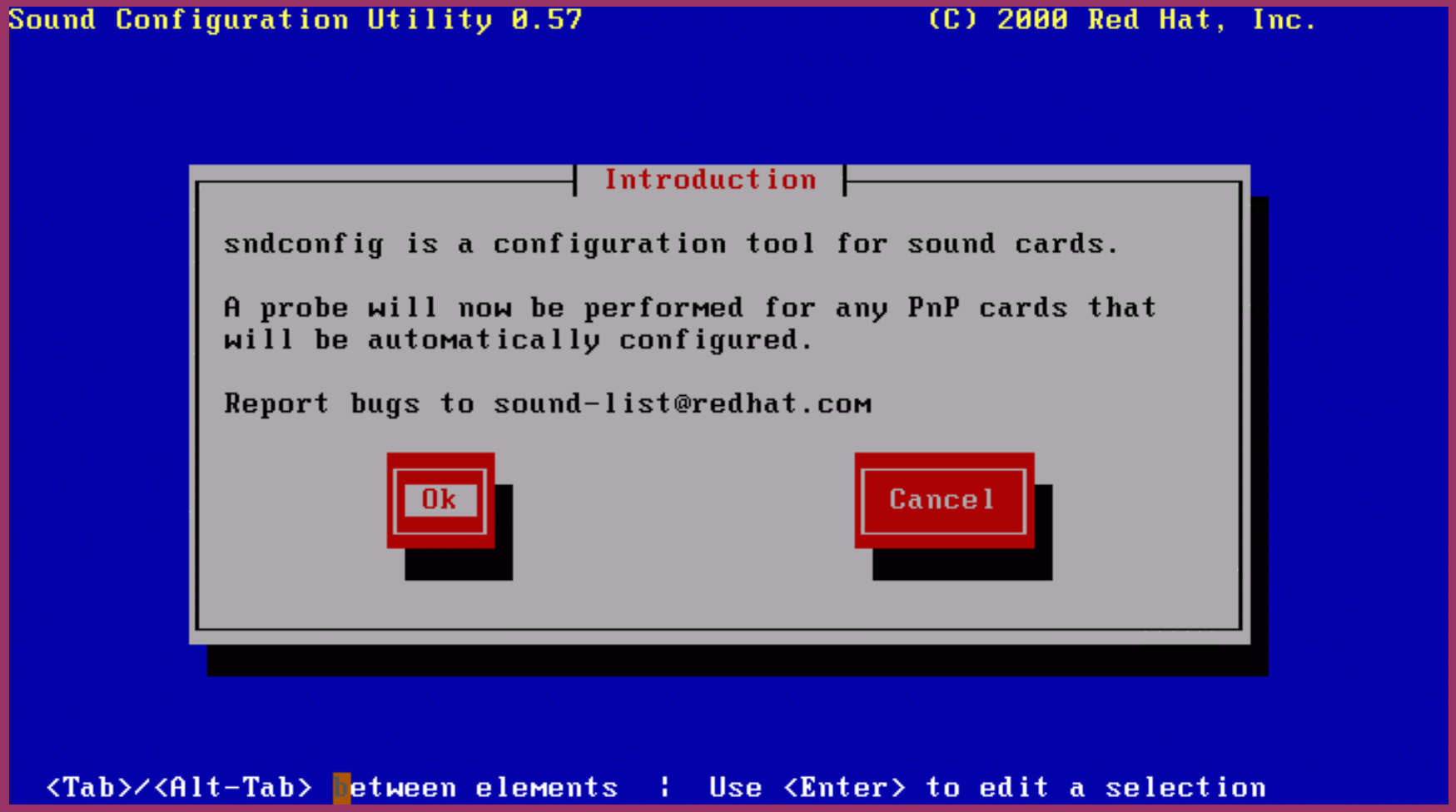
- Note: Many Linux commands support both the “-v” option style and the GNU “--verbose” option style. Check the man page for the command for details

technical computing – invent. design. deliver.

chnical computing – invent. design. deliver.

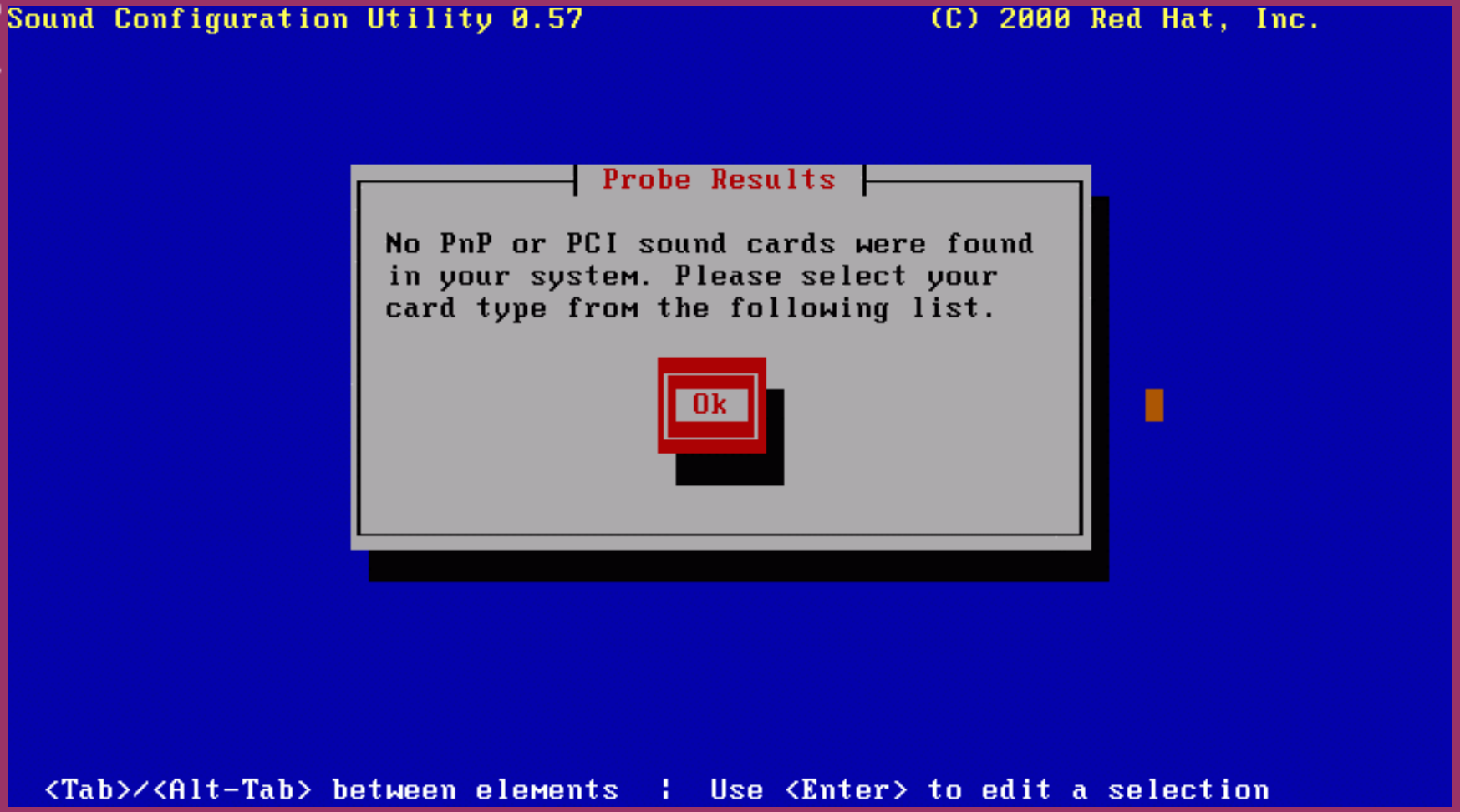
Configuring Your Sound Card

Sound Configuration Utility "sndconfig"



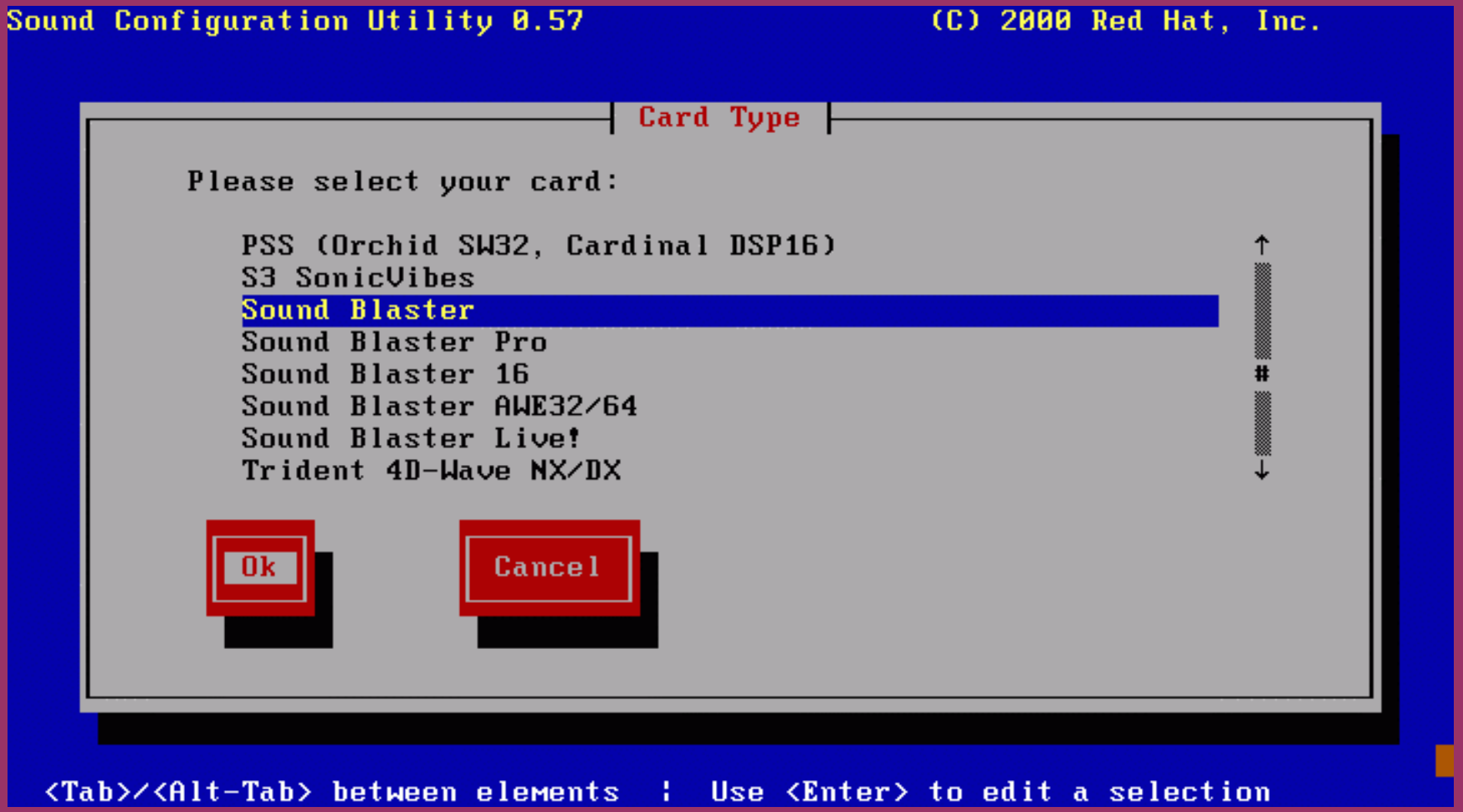
technical computing – invent. design. deliver.

Sound Configuration: Probing for Sound Hardware



technical computing – invent. design. deliver.

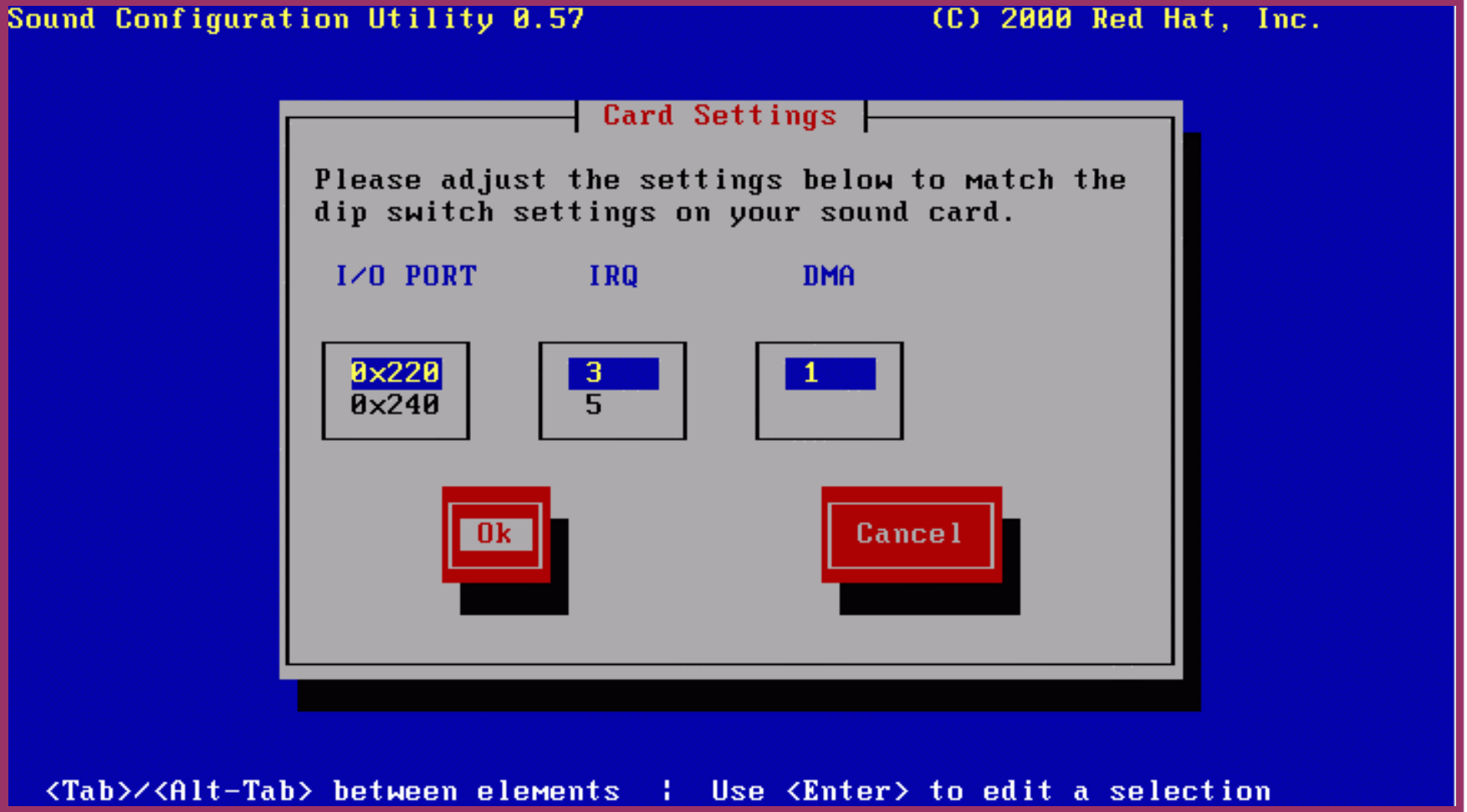
Sound Configuration: Selecting the Card Type



technical computing – invent. design. deliver.

technical computing – invent. design. deliver.

Sound Configuration: Device Parameters



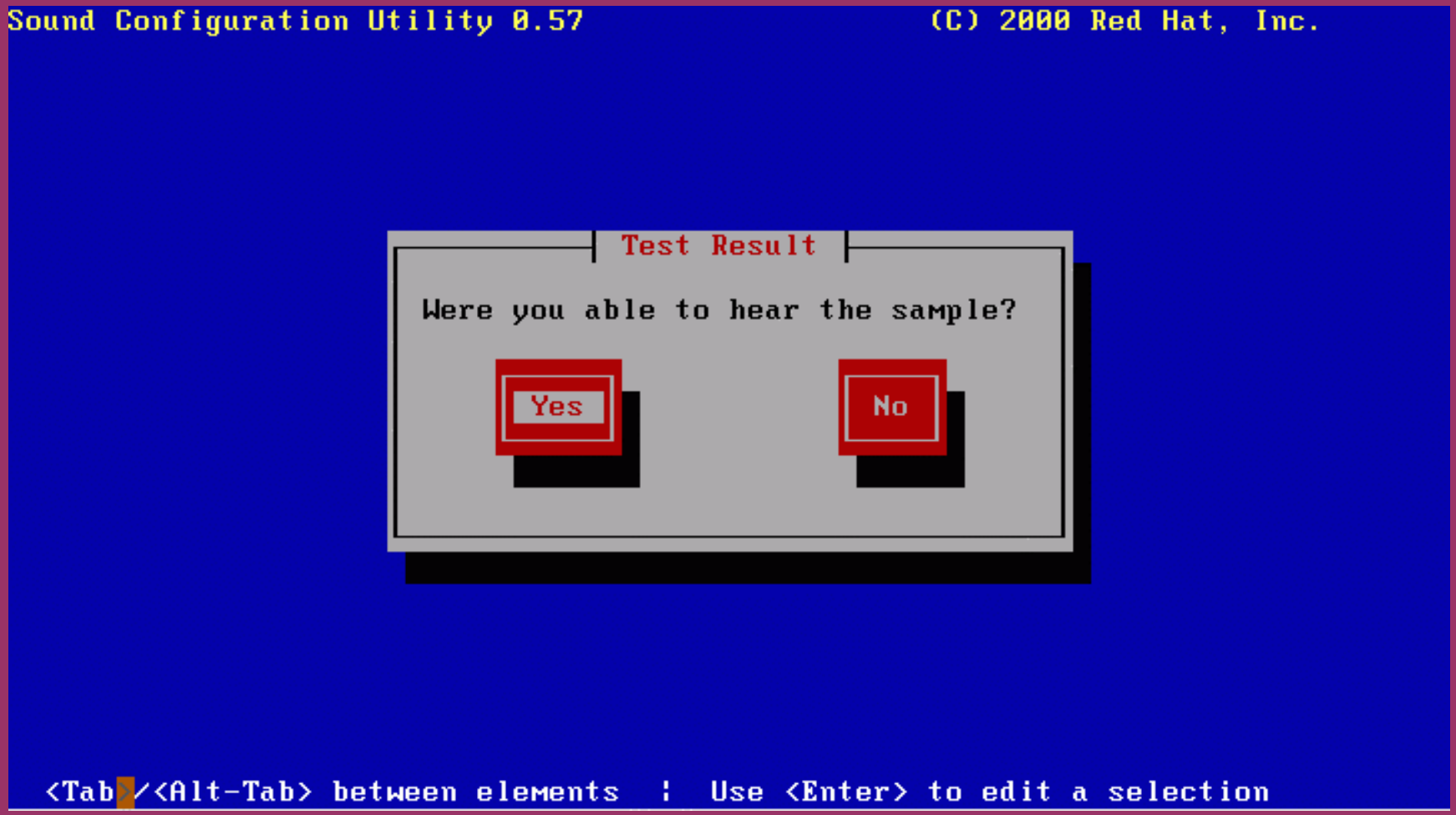
Sound Configuration: Loading the Module

technical computing – invent. design. deliver.

Sound Configuration: Playing Sample Sound

technical computing – invent. design. deliver.

Sound Configuration: Verifying Operation



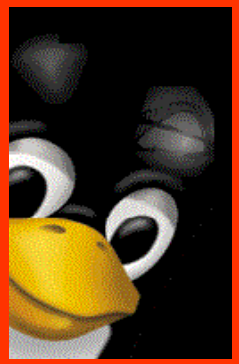
technical computing – invent. design. deliver.

Sound Configuration: Verifying the MIDI Interface

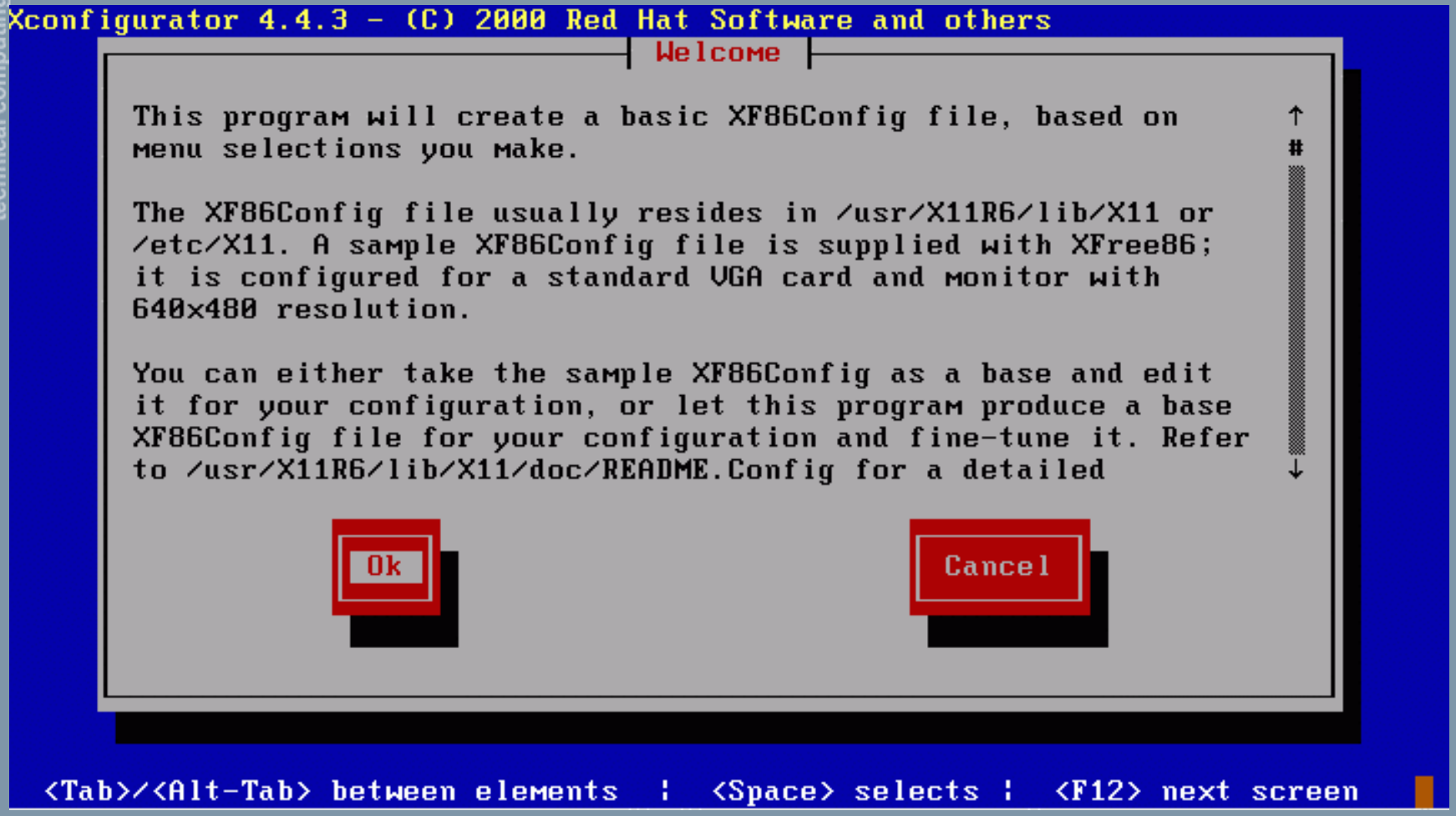
technical computing – invent. design. deliver.

hnical computing – invent. design. deliver.

Configuring X-Windows



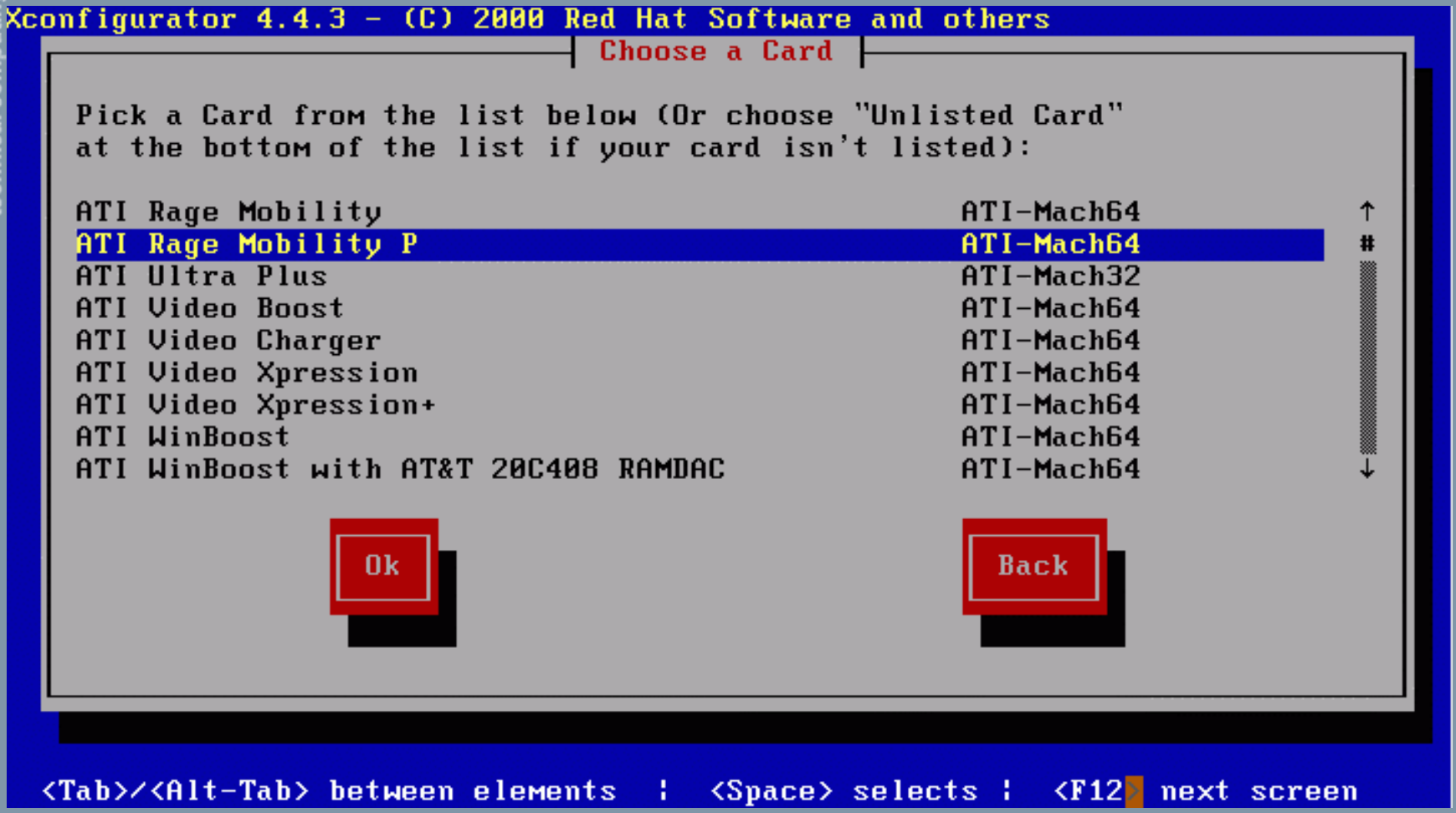
X Configuration: The “Xconfigurator” Utility



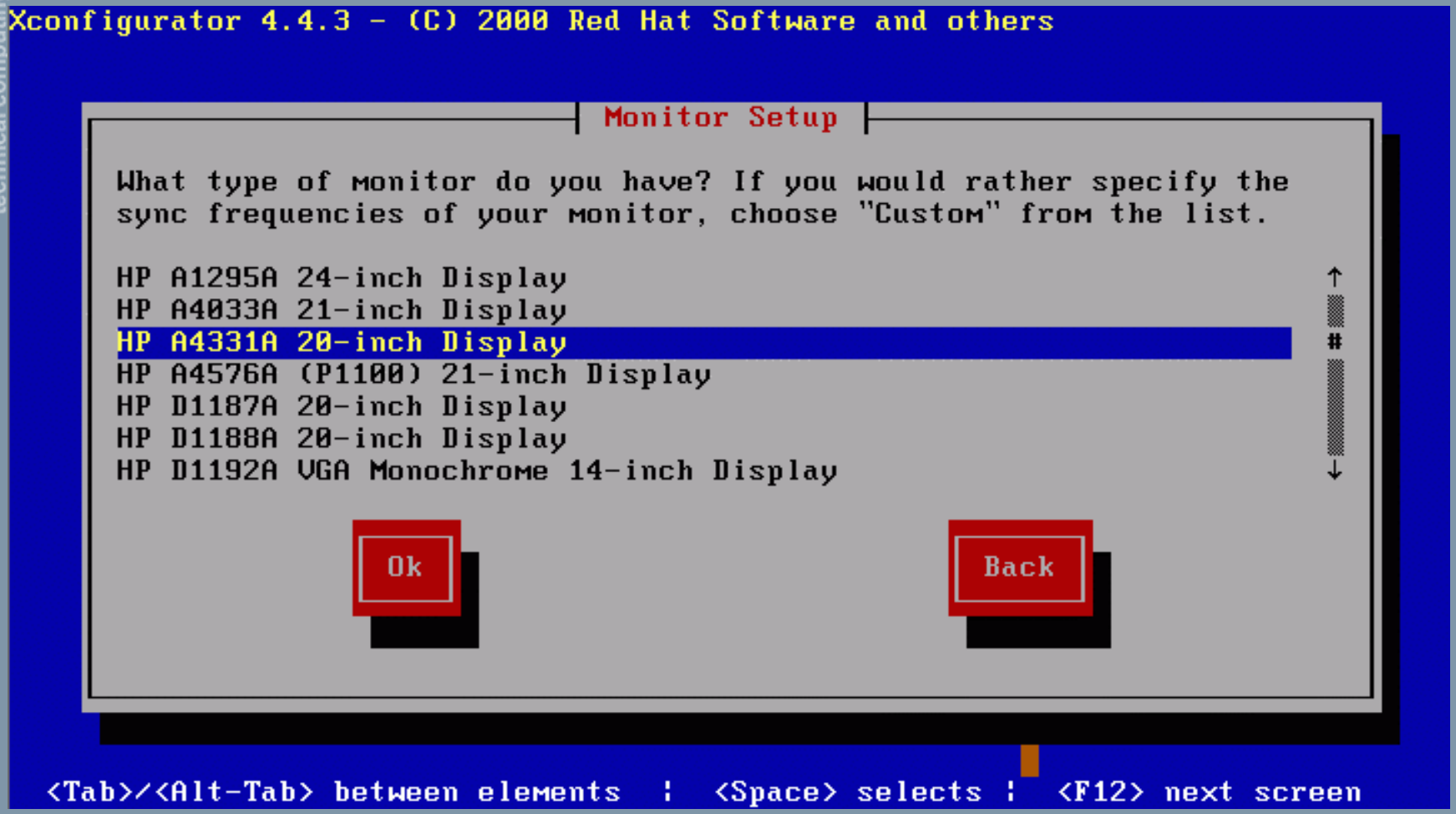
technical computing – invent. design. deliver.

technical computing – invent. design. deliver.

X Configuration: Select Adapter Type

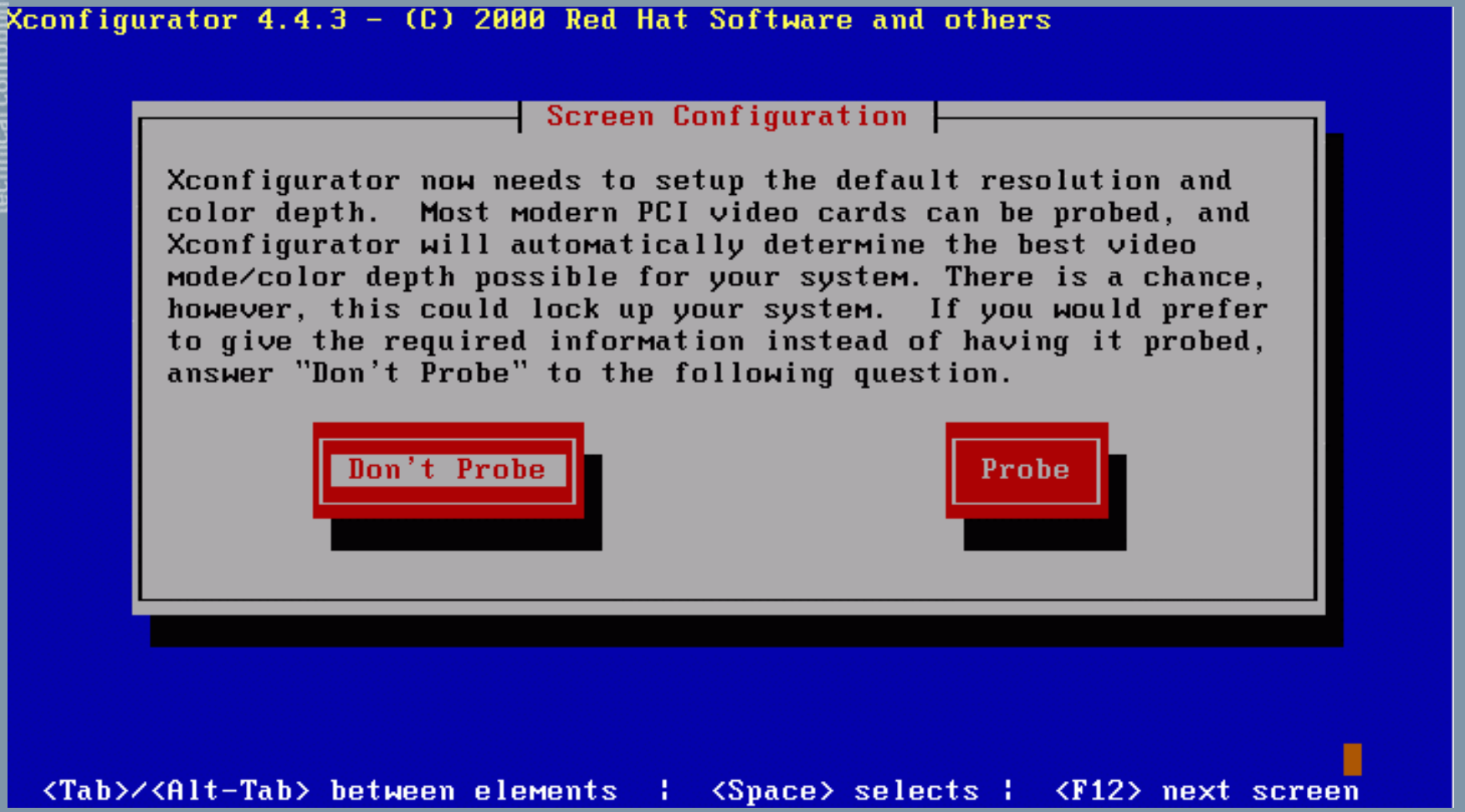


X Configuration: Select Monitor Type

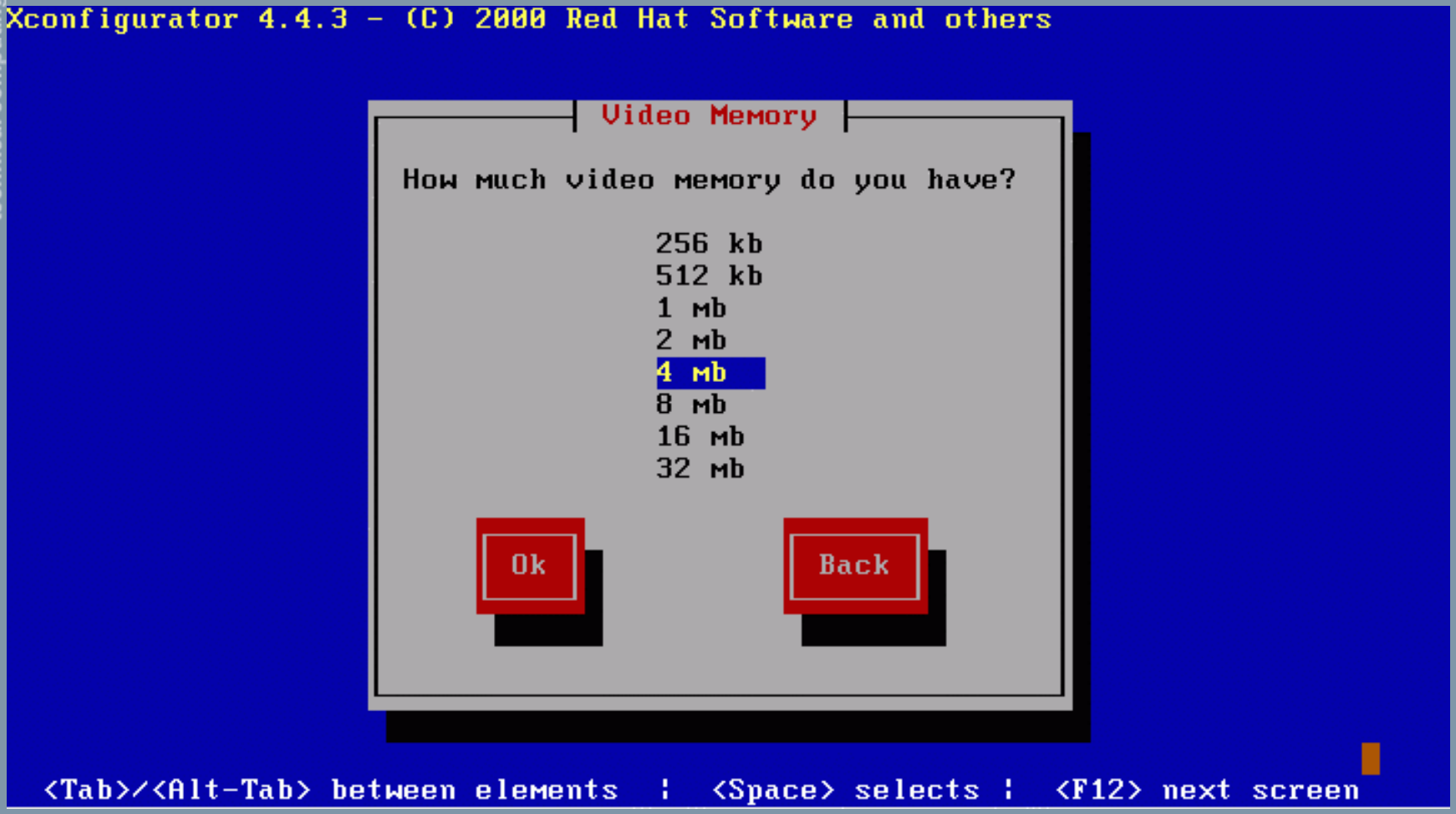


technical computing – invent. design. deliver.

X Configuration: Probe for Display Memory

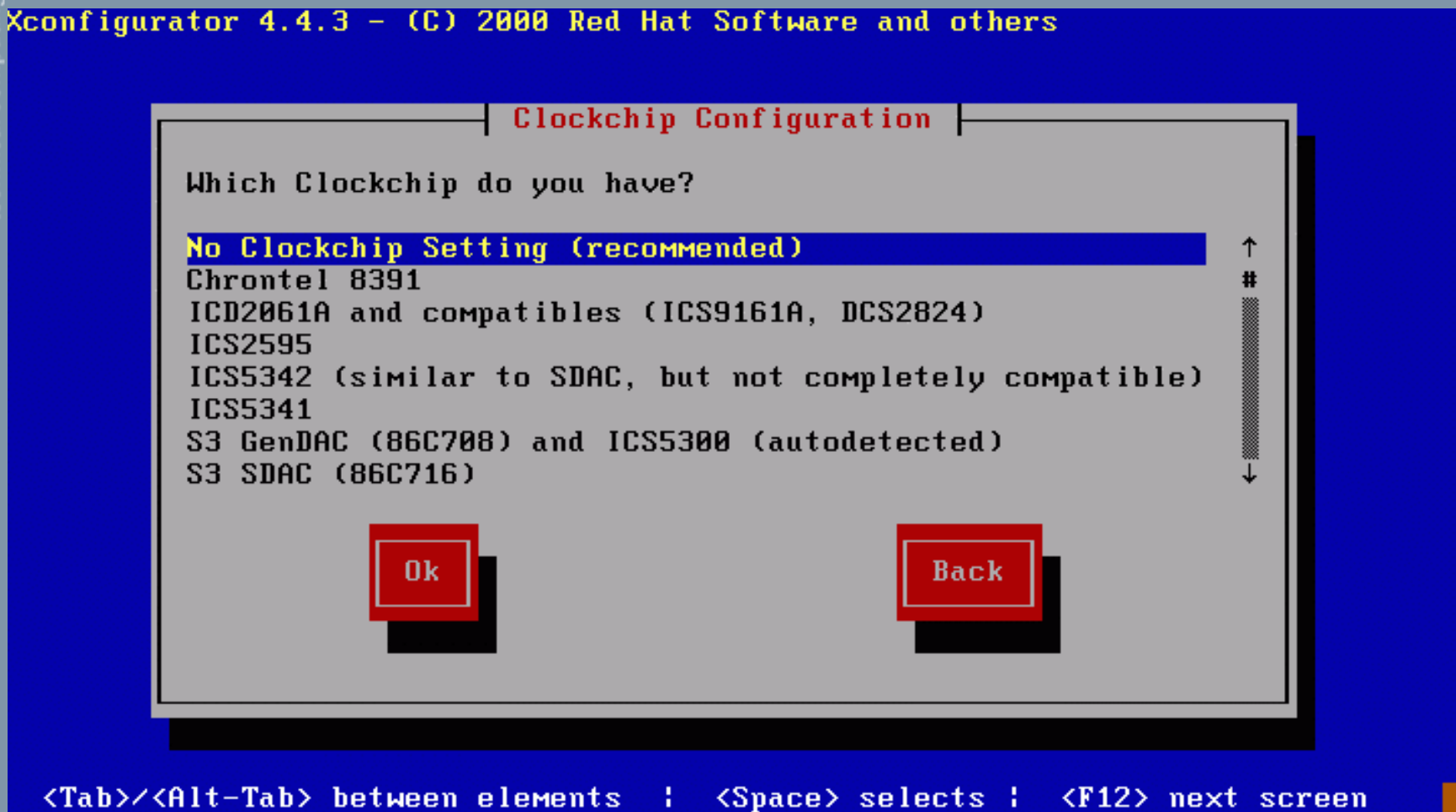


X Configuration: Verifying Display Memory



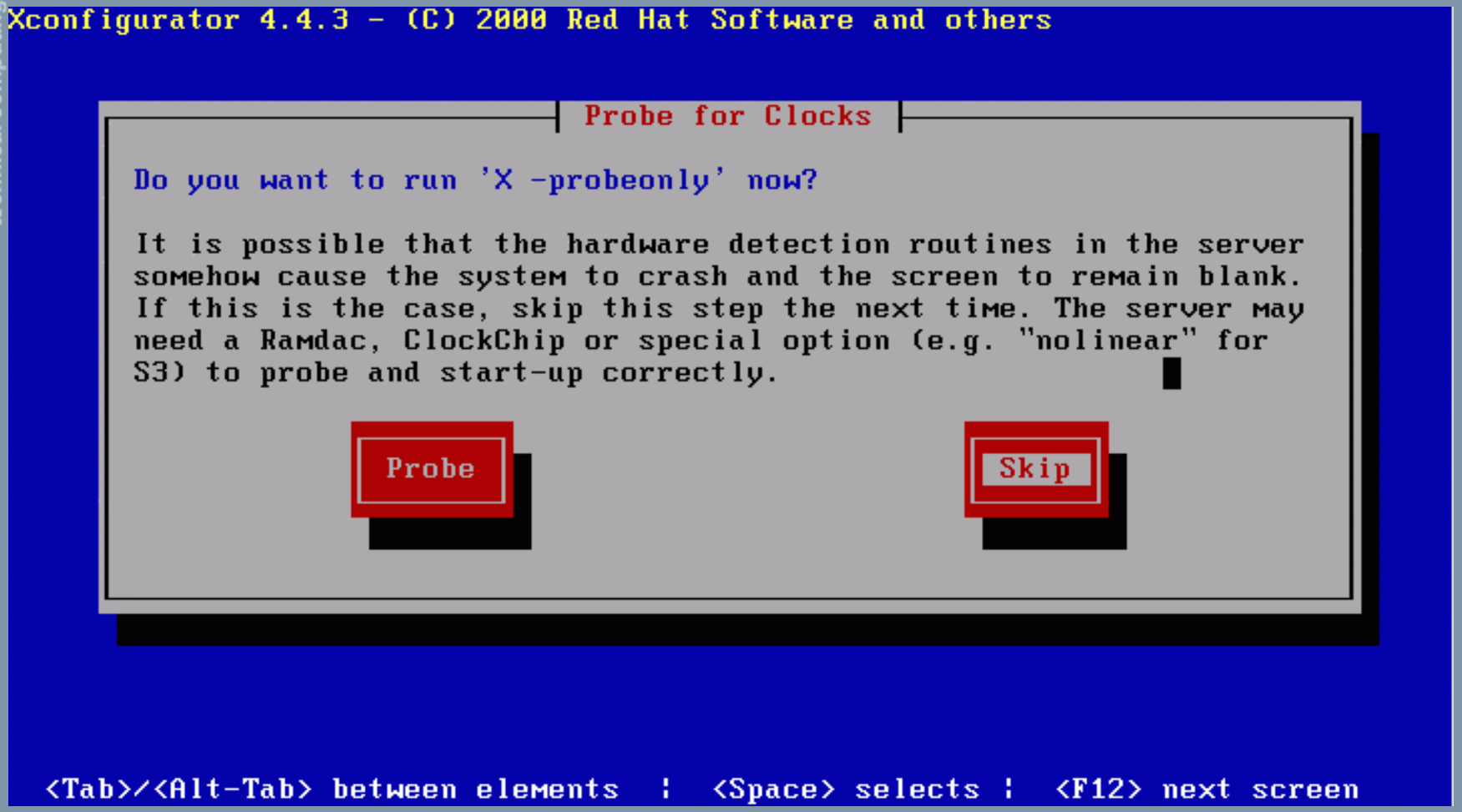
technical computing – invent. design. deliver.

X Configuration: Select Clock Chip

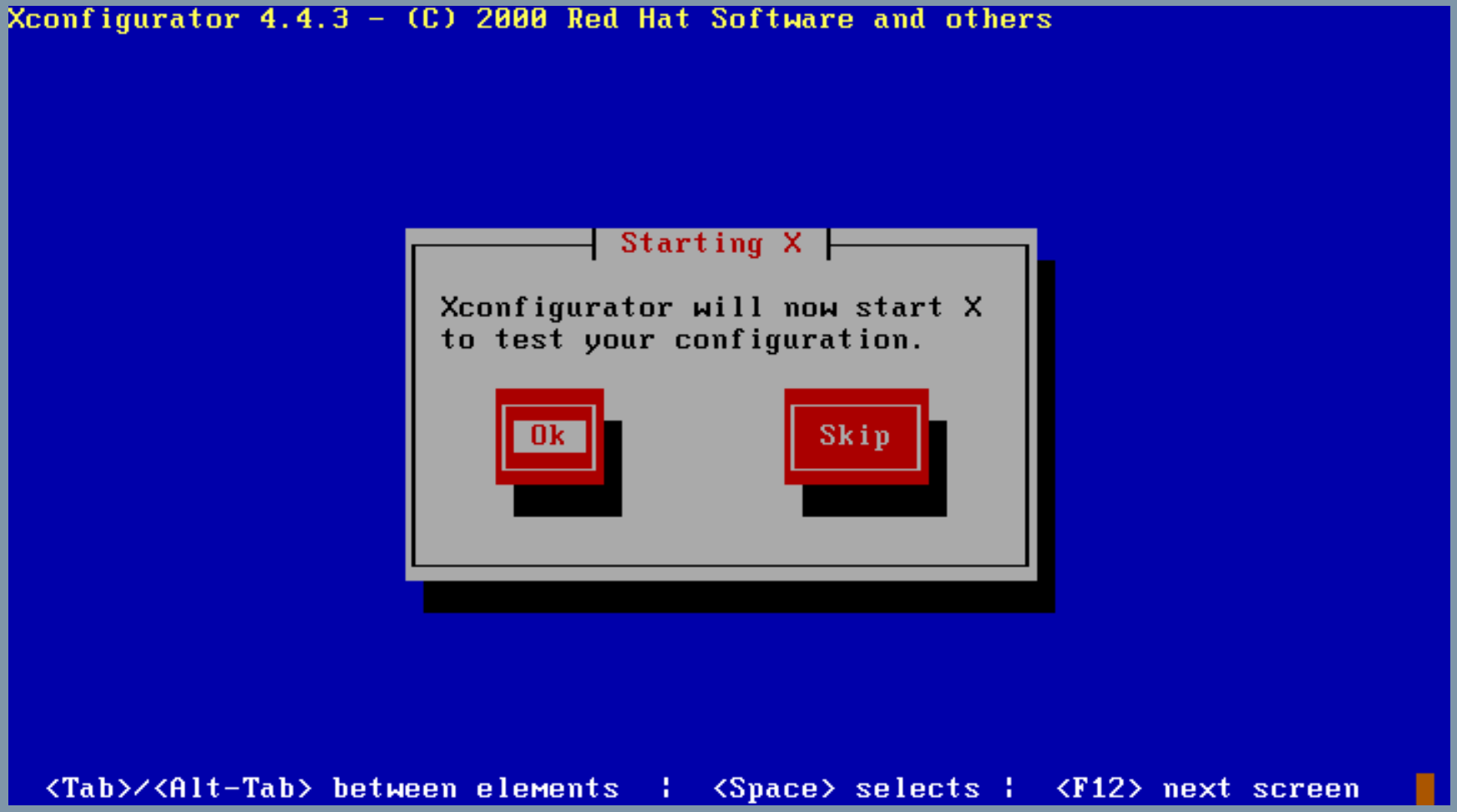


technical computing – invent. design. deliver.

X Configuration: Probe for Clock Settings



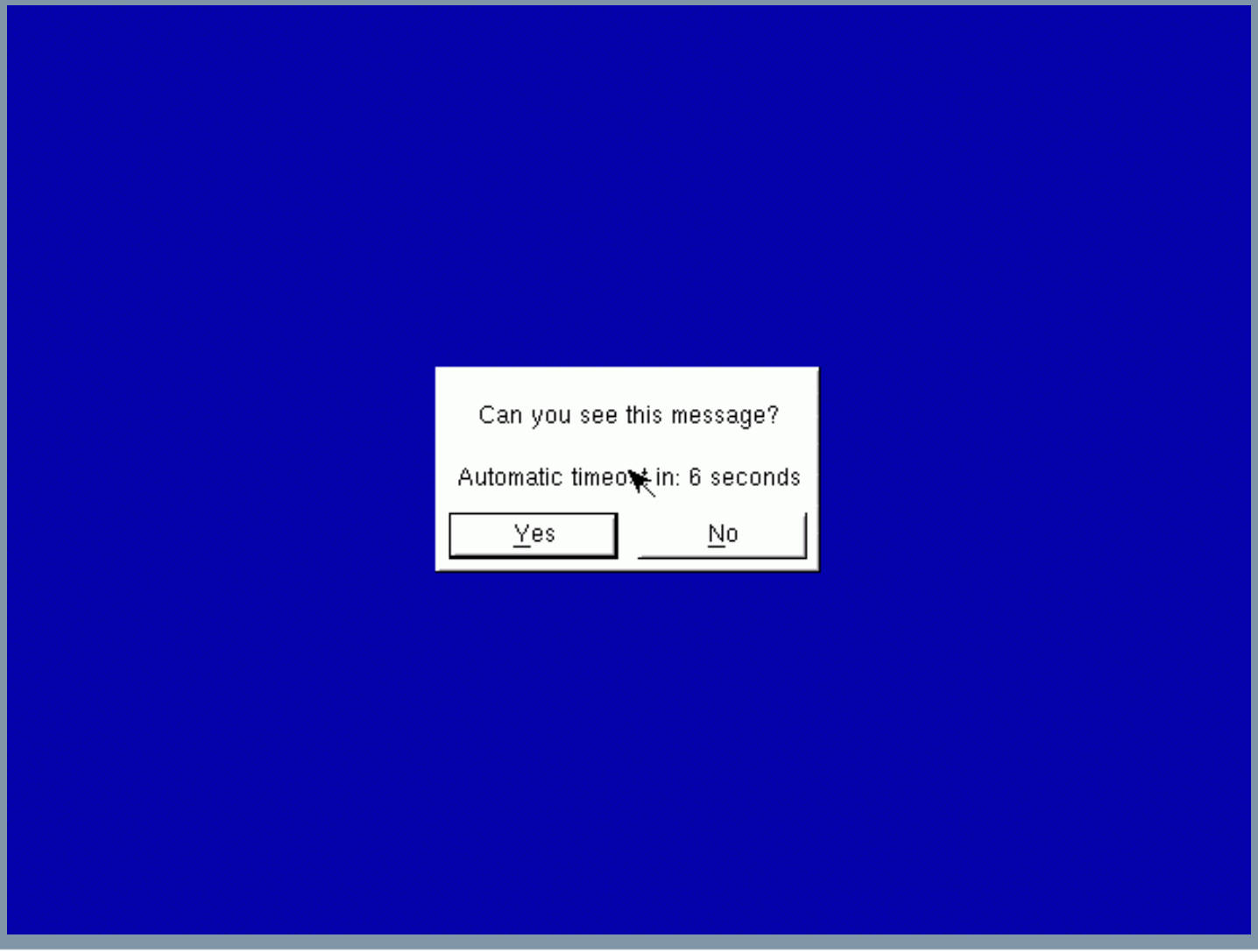
X Configuration: Testing the Settings



technical computing – invent. design. deliver.

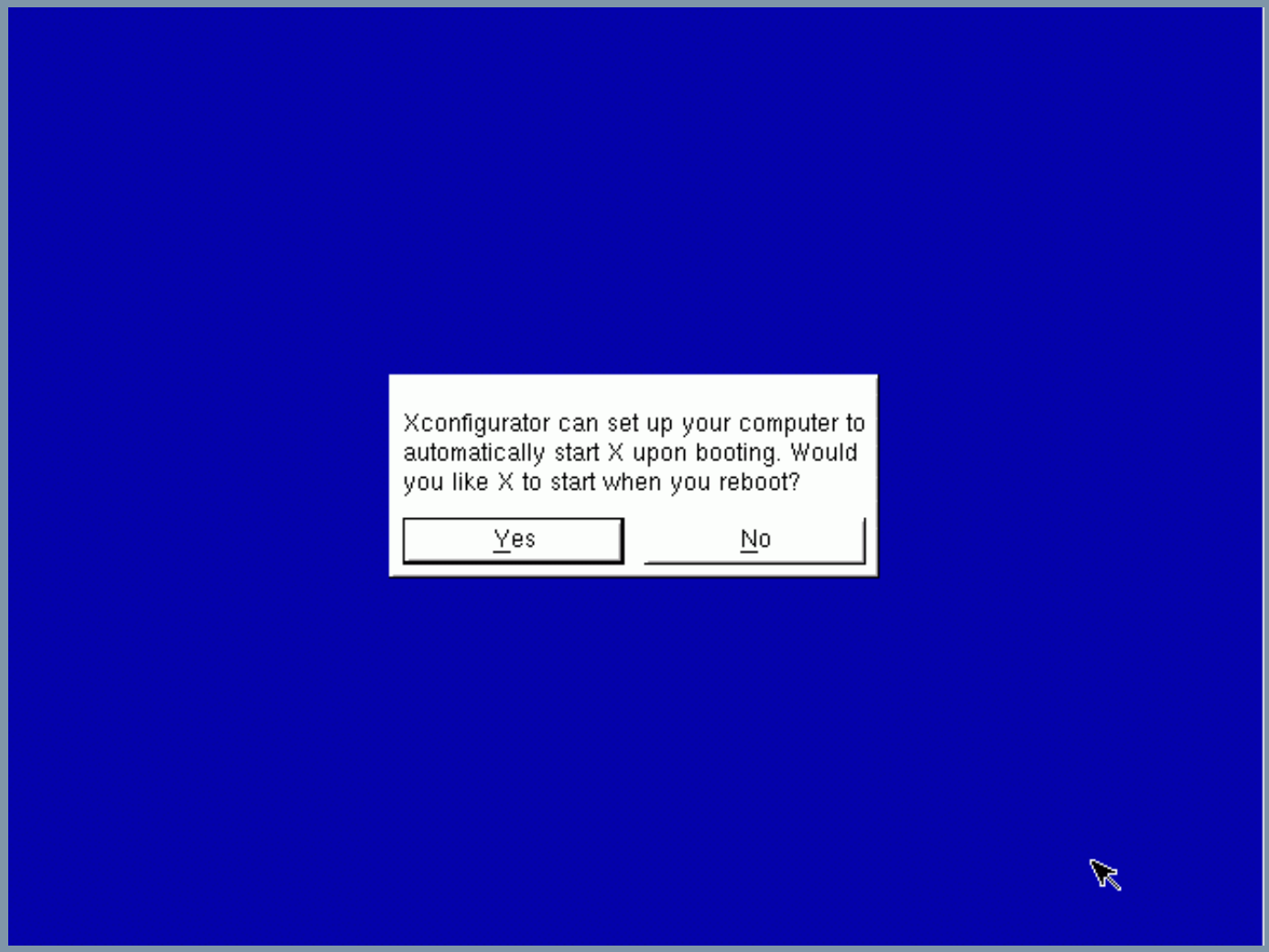
technical computing – invent. design. deliver.

X Configuration: Successful Startup



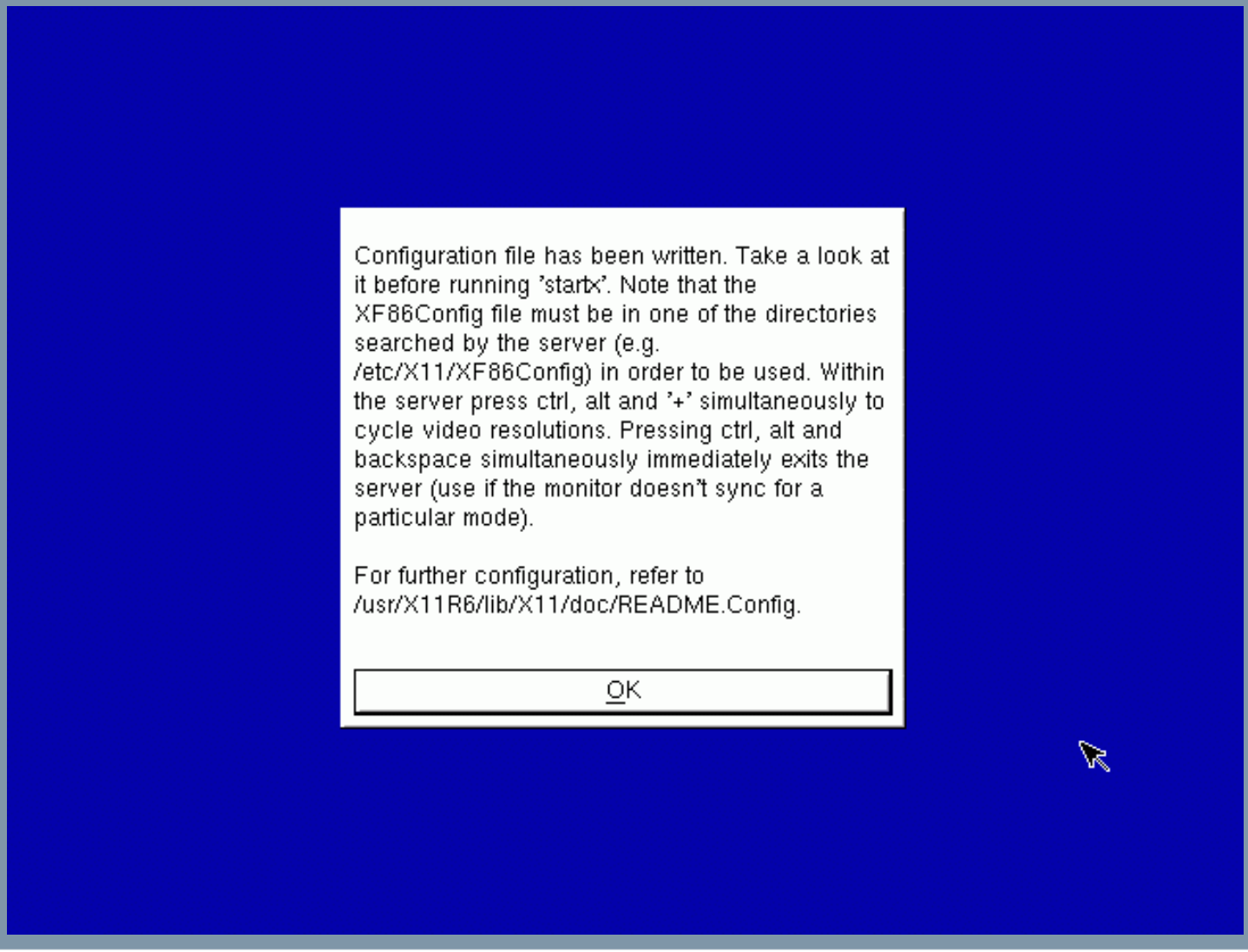
technical computing – invent. design. deliver.

X Configuration: Automatic X Startup?



technical computing – invent. design. deliver.

X Configuration: Configuration File Written



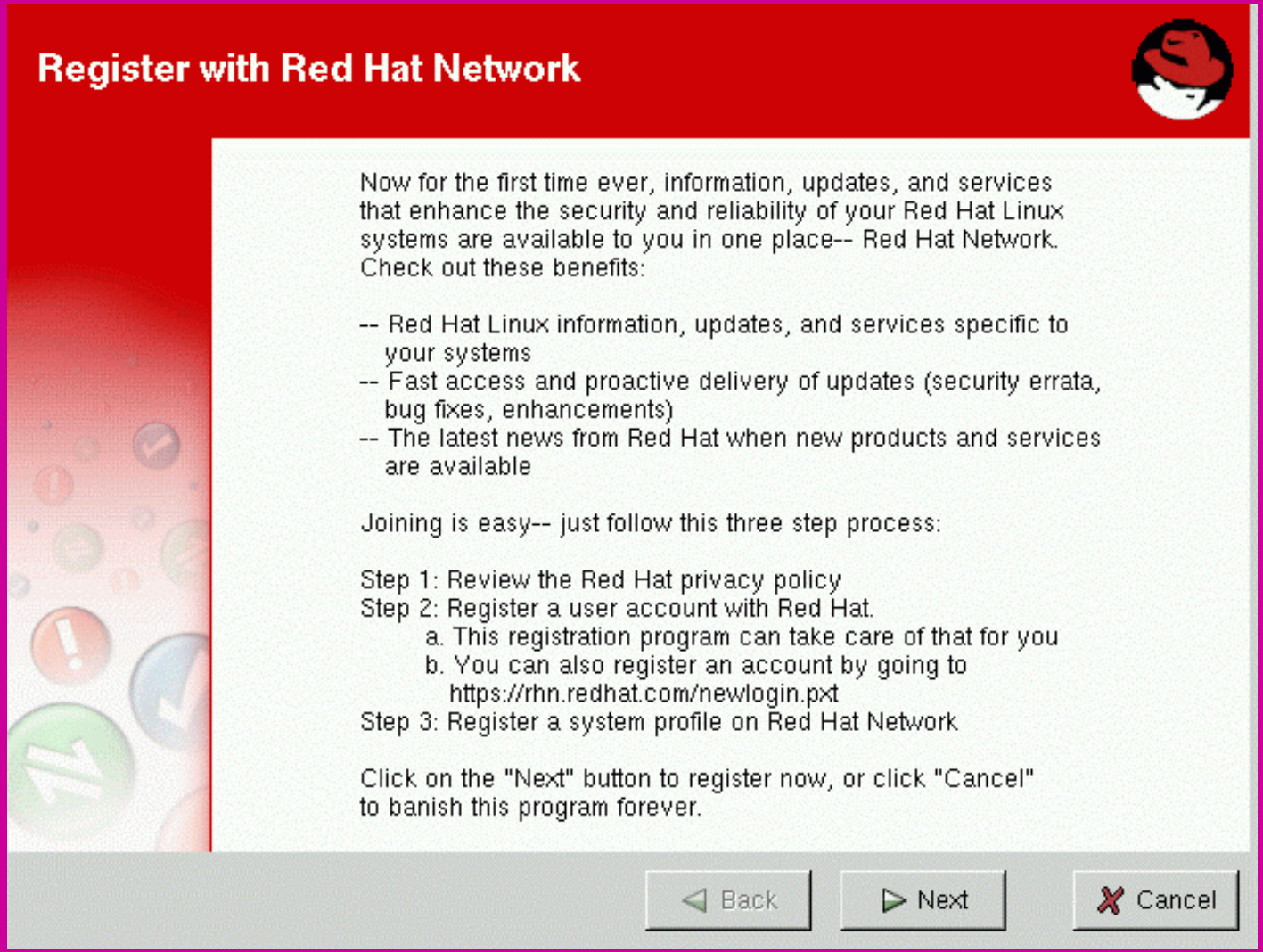
technical computing – invent. design. deliver.

X Configuration Tips

- Find out as much as you can about your display and monitor hardware before starting the configuration
- Wait until *after* you have successfully verified X-Windows operation to make graphical login the default
- Boot in run-level 3, then use “startx” to test your configuration
- Modify /etc/inittab to make the X run-level the default after verification
- To kill the X-server, Ctrl-Alt-Backspace
- If you selected more than one display depth, you can cycle between them with Ctrl-Alt-+
- There are more than one virtual displays on the X-server, you can get to them with Ctrl-Alt-F[1-8]
- The configuration file is /etc/X11/XF86Config (version 3.3.6) or /etc/X11/XF86Config-4 (version 4.X.X)

technical computing – invent. design. deliver.

up2date

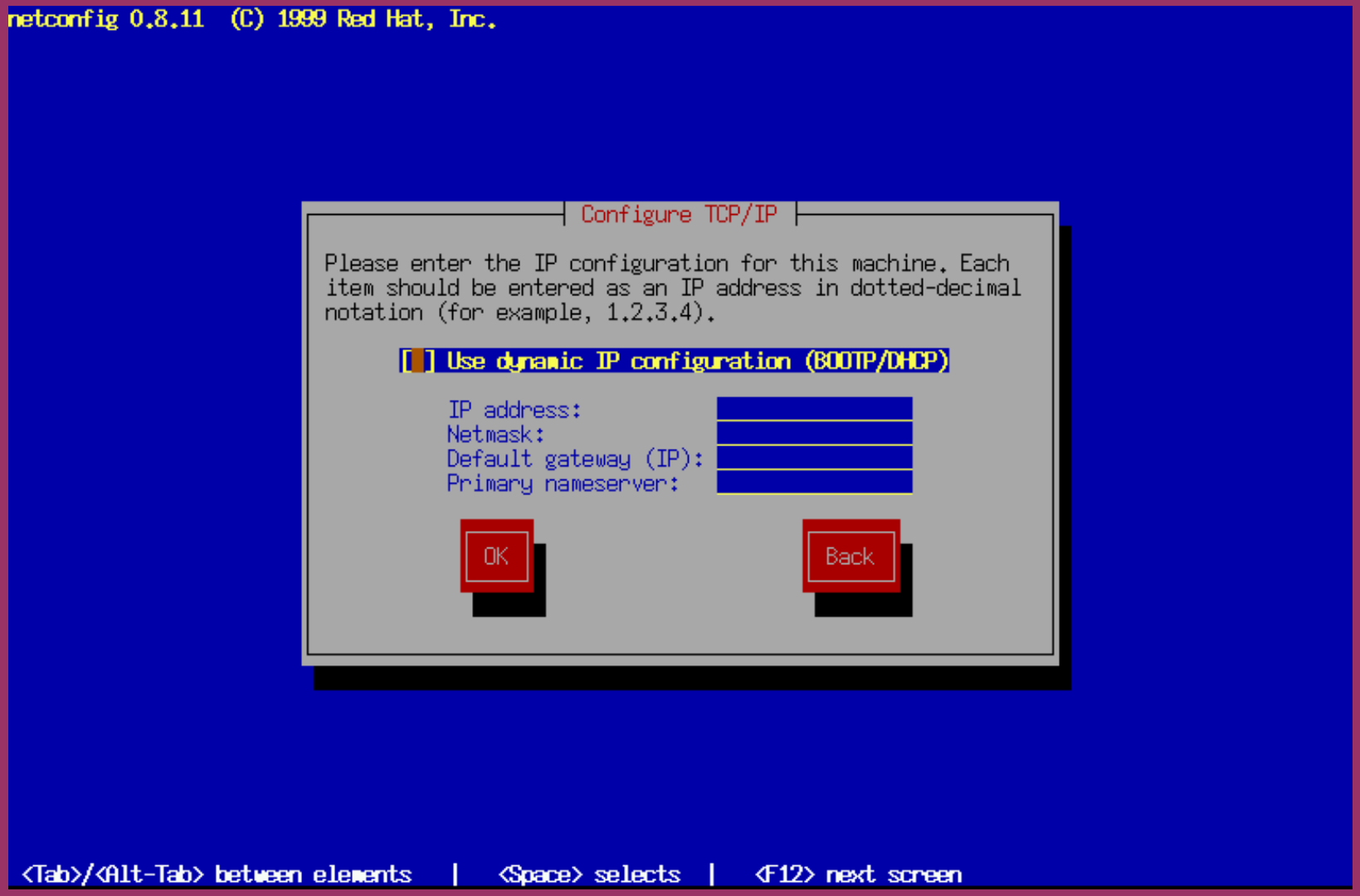


technology computing – invent. design. deliver.

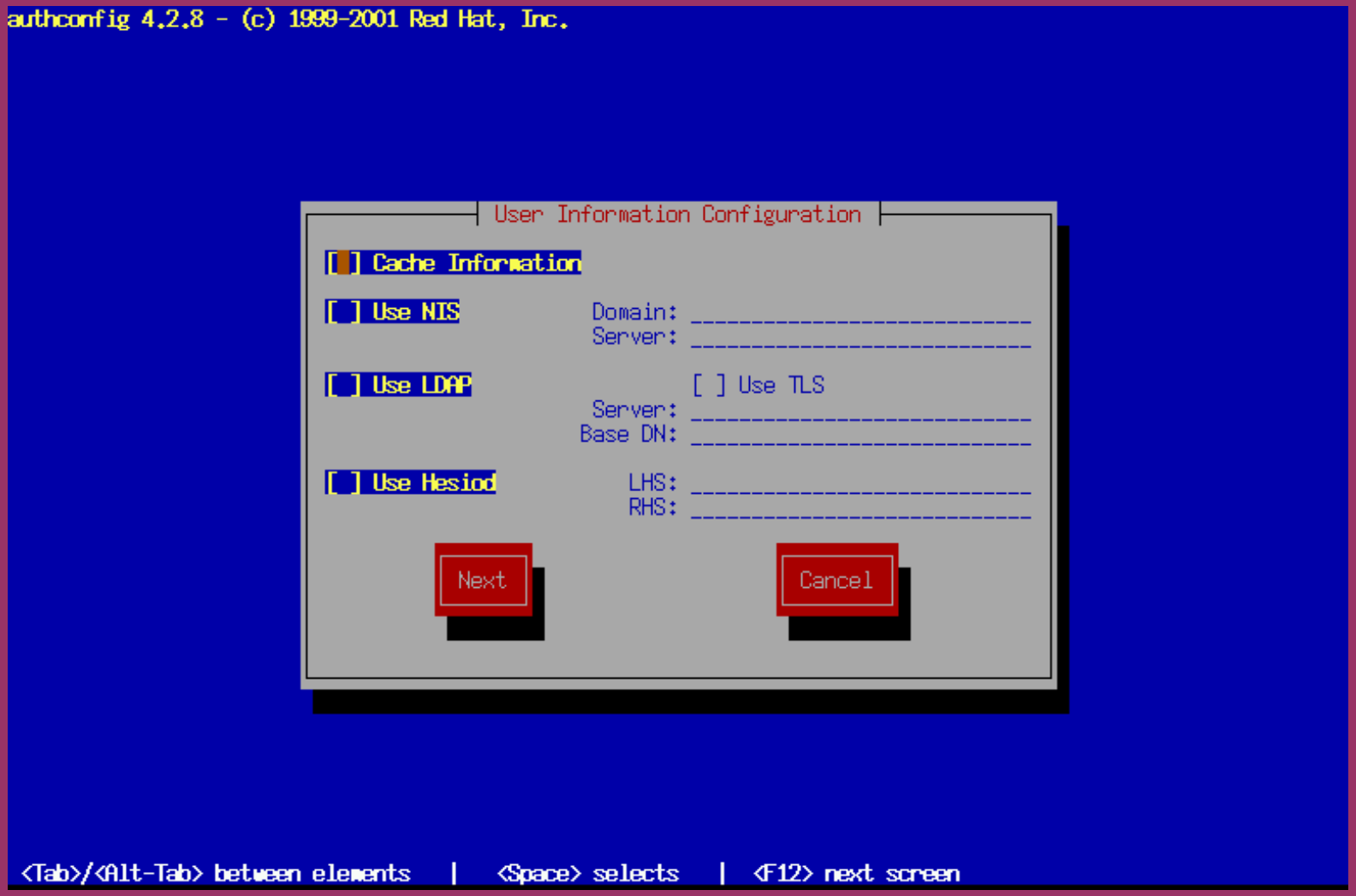
Miscellaneous Redhat Linux Configuration Tools

technical computing – invent. design. deliver.

“netconfig”



“authconfig”



technical computing – invent. design. deliver.

technical computing – invent. design. deliver.

“authconfig”

authconfig 4.2.8 - (c) 1999-2001 Red Hat, Inc.

Authentication Configuration

[*] Use Shadow Passwords

[*] Use MD5 Passwords

[ ] Use LDAP Authentication [ ] Use TLS

Server: _____

Base DN: _____

[ ] Use Kerberos 5

Realm: _____

KDC: _____

Admin Server: _____

[ ] Use SMB Authentication

Workgroup: _____

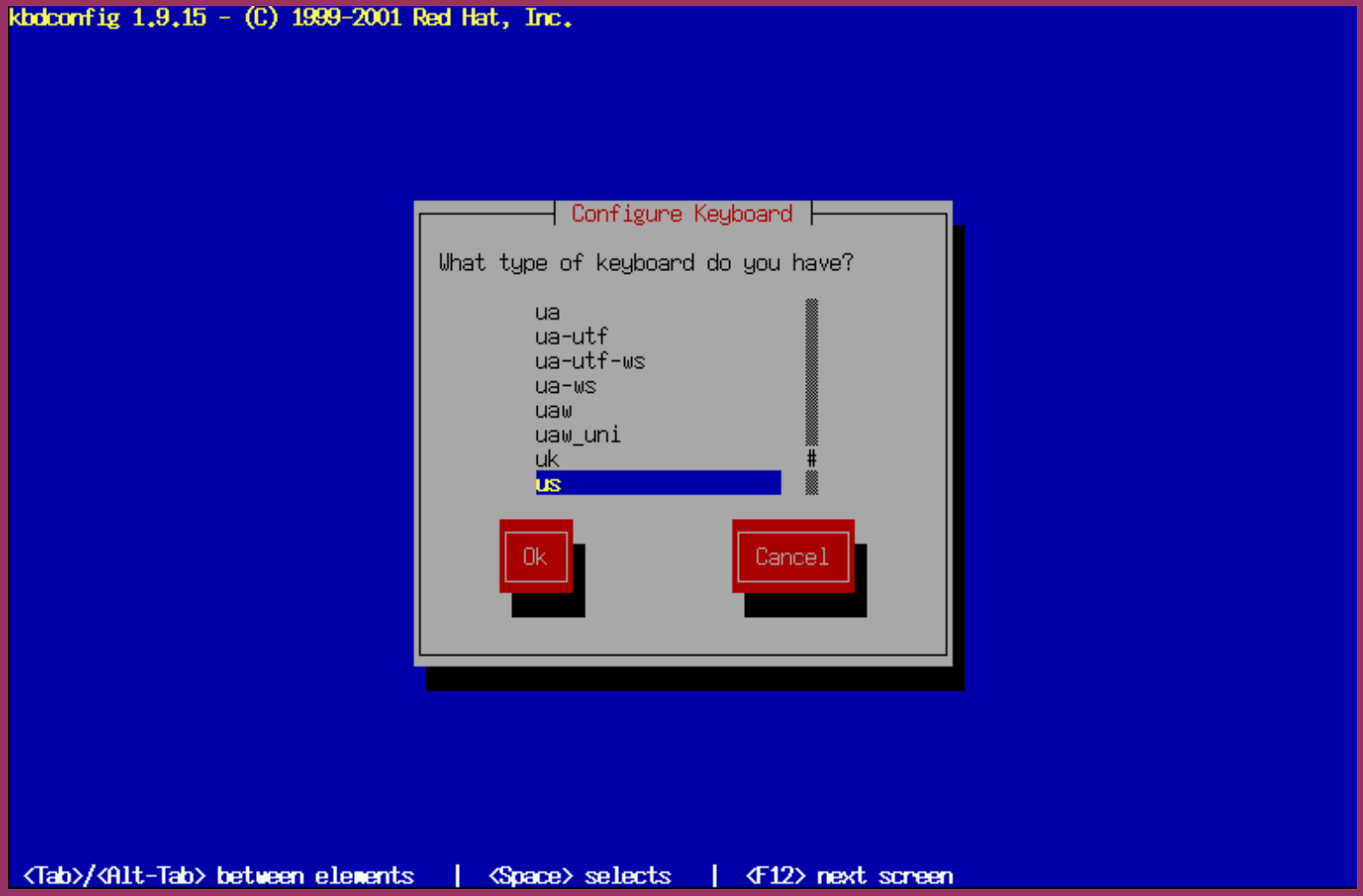
Servers: _____

Ok Back Cancel

<Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen

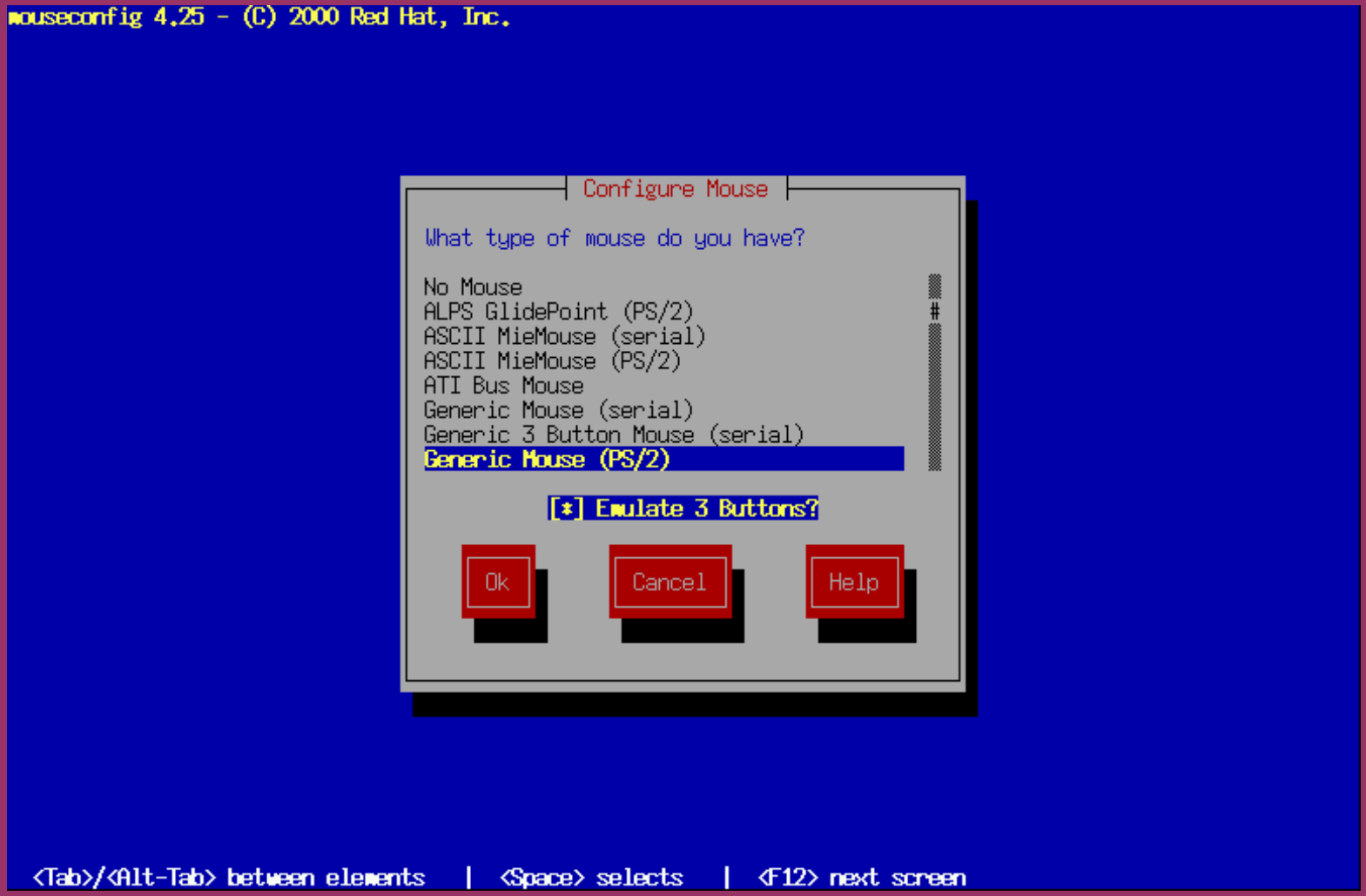
technical computing – invent. design. deliver.

“kbdconfig”



technical computing – invent. design. deliver.

“mouseconfig”



technical computing – invent. design. deliver.

“timeconfig”

timeconfig 3.2.7 - (C) 2002 Red Hat, Inc.

Configure Timezones

Format machine time is stored in:

[ ] Hardware clock set to GMT

America/Kentucky/Louisville
America/Kentucky/Monticello
America/Knox_IN
America/La_Paz
America/Lima
America/Los_Angeles

Ok

Cancel

<Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen

technology computing – invent. design. deliver.

Network Installation with Kickstart

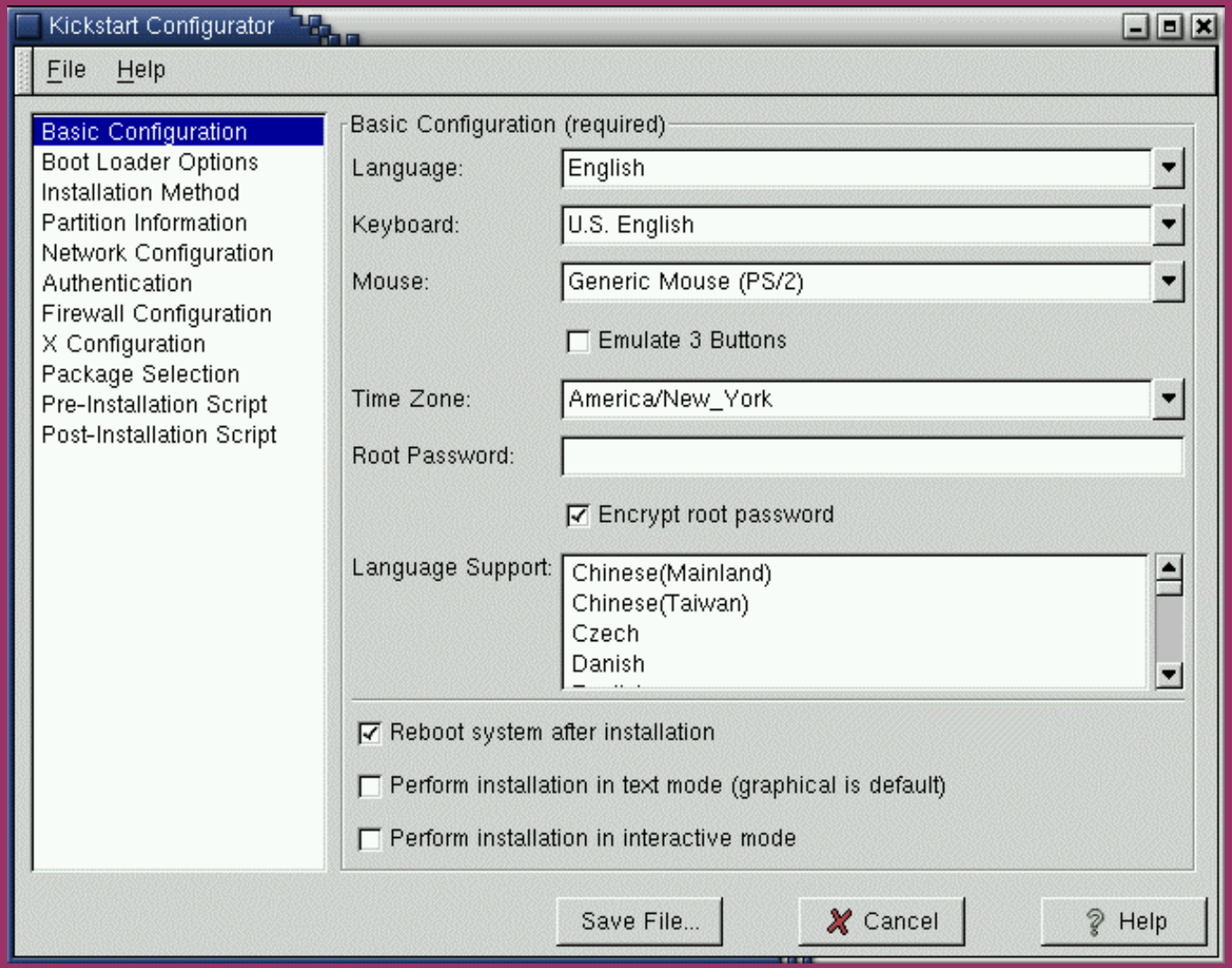
technical computing – invent. design. deliver.

Kickstart

- Kickstart can allow you to do interactive network installs via FTP, HTTP, or NFS
- Systems may boot from floppy, DHCP, or other network boot protocols.
- Normal installation writes a /root/anaconda-ks.cfg file that contains all of the choices made during system installation
- Use that file or “ksconfig” to make the Kickstart control file
- Copy the Redhat CD-ROM installation disks to the Kickstart server and make them available to Kickstart
- A complete Kickstart tutorial is beyond the scope of this session.
- There are “HowTos” available for Kickstart on Linux community web sites.

technical computing – invent. design. deliver.

“ksconfig”

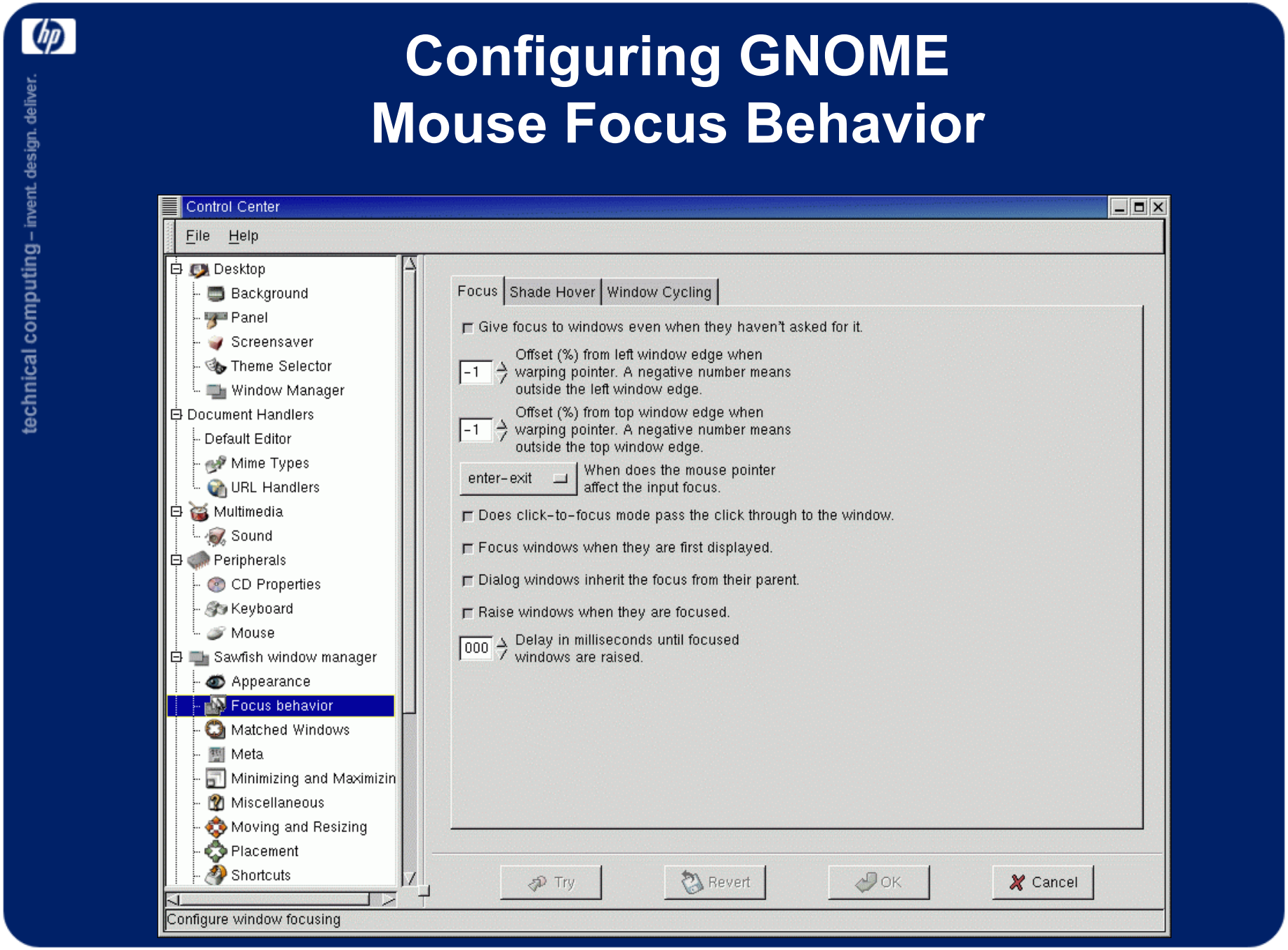


hnical computing – invent. design. deliver.

Miscellaneous Linux Stuff

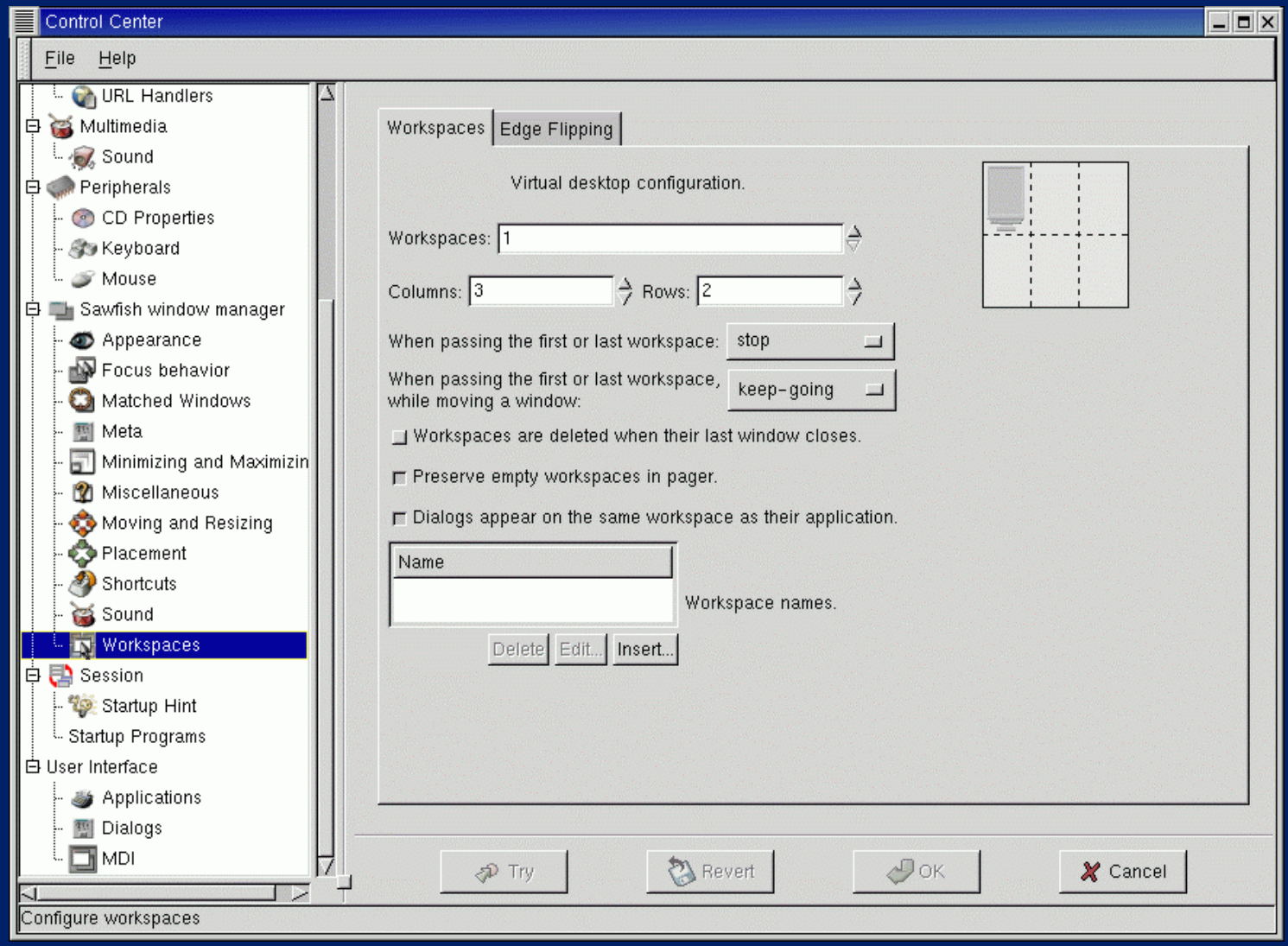
Linux Running Windows via VMware

technical computing – invent. design. deliver.

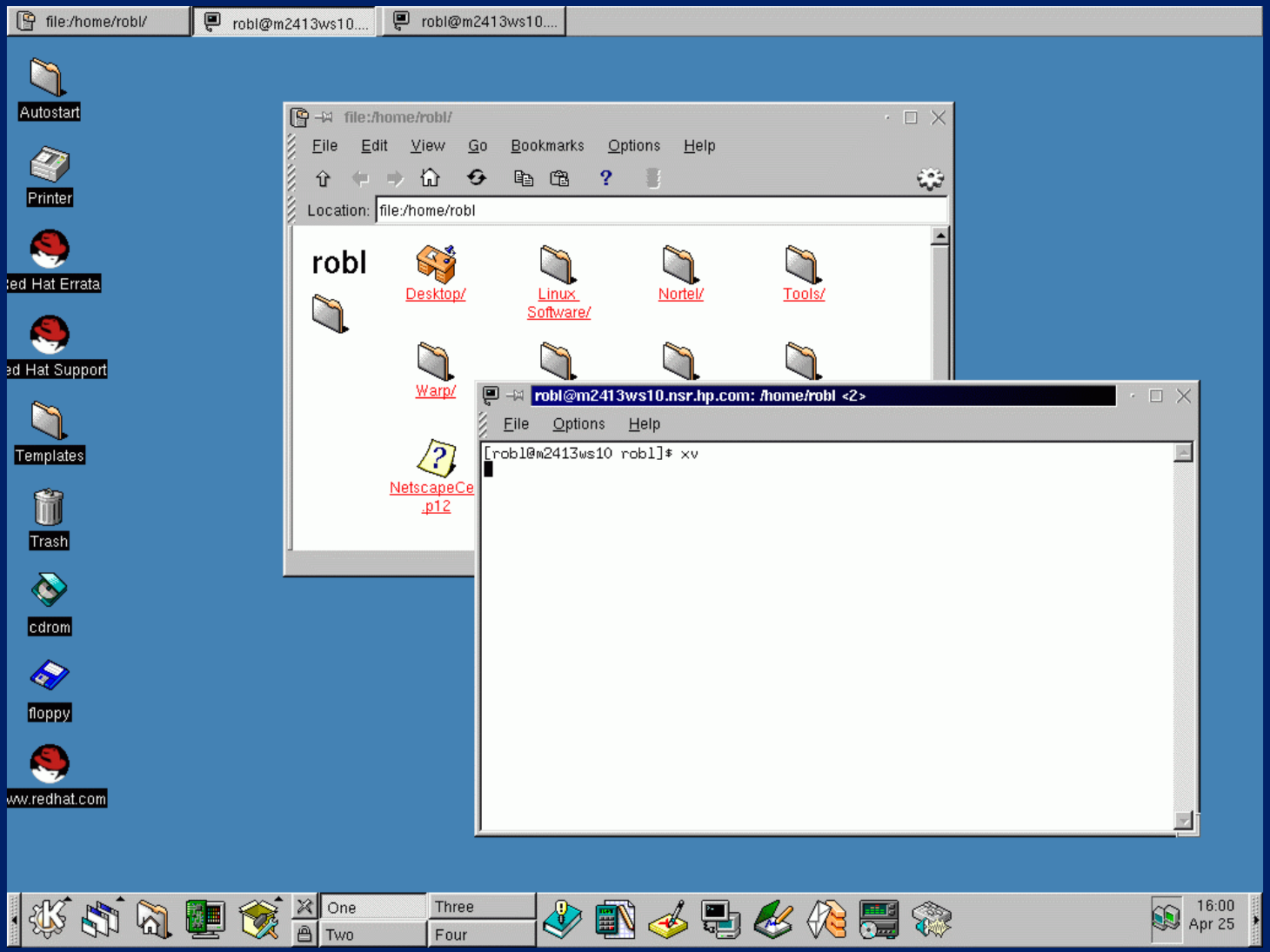

technical computing – invent. design. deliver.
Configuring GNOME
Mouse Focus Behavior
Control Center
File Help
Desktop
Background
Panel
Screensaver
Theme Selector
Window Manager
Document Handlers
Default Editor
Mime Types
URL Handlers
Multimedia
Sound
Peripherals
CD Properties
Keyboard
Mouse
Sawfish window manager
Appearance
Focus behavior
Matched Windows
Meta
Minimizing and Maximizin
Miscellaneous
Moving and Resizing
Placement
Shortcuts
Focus
Shade Hover
Window Cycling
Give focus to windows even when they haven't asked for it.
-1
Offset (%) from left window edge when warping pointer. A negative number means outside the left window edge.
-1
Offset (%) from top window edge when warping pointer. A negative number means outside the top window edge.
enter-exit
When does the mouse pointer affect the input focus.
Does click-to-focus mode pass the click through to the window.
Focus windows when they are first displayed.
Dialog windows inherit the focus from their parent.
Raise windows when they are focused.
000
Delay in milliseconds until focused windows are raised.
Try
Revert
OK
Cancel
Configure window focusing

technical computing – invent. design. deliver.

Configuring GNOME Workspaces

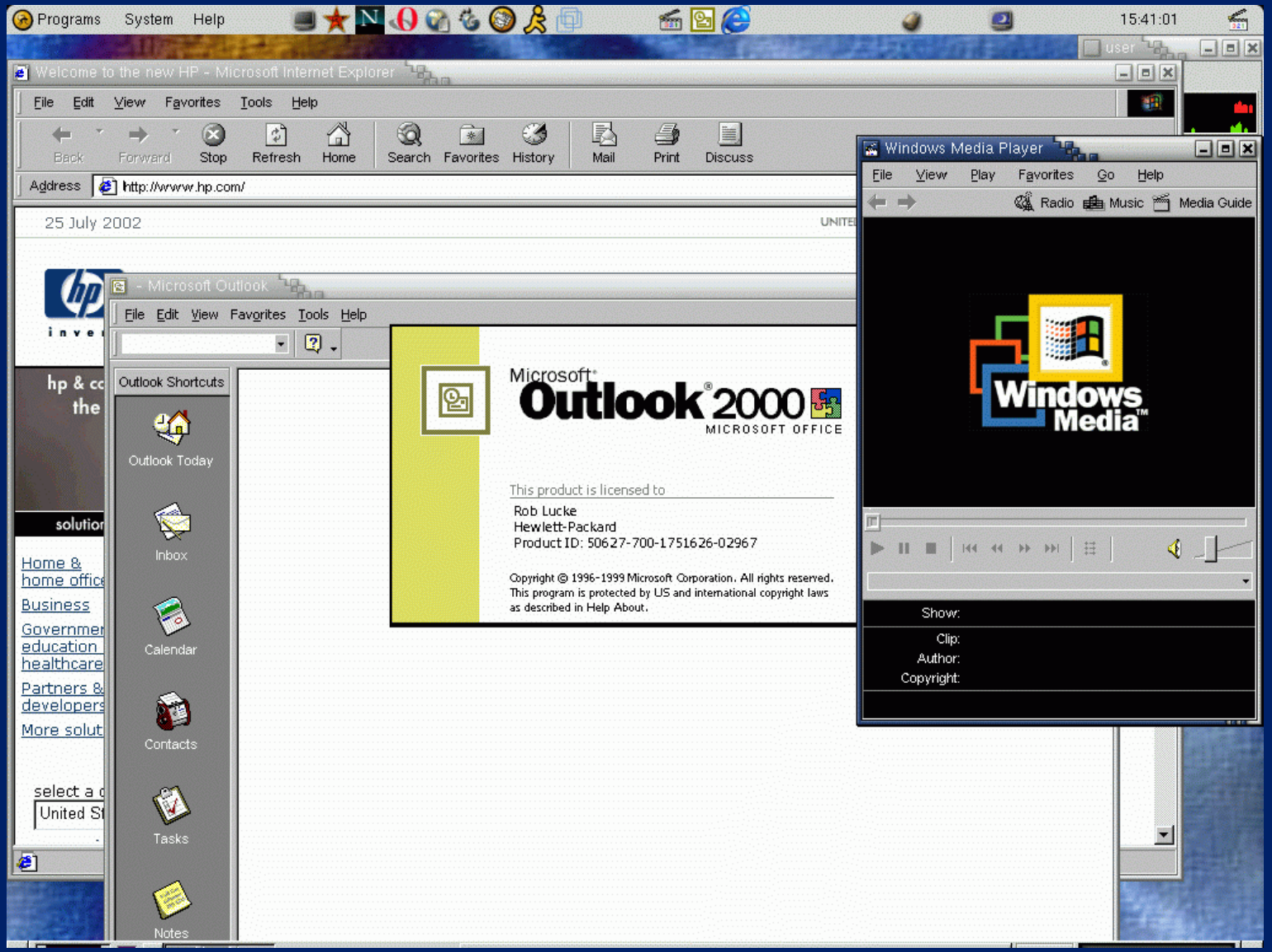


KDE Desktop



technical computing – invent. design. deliver.

CrossOver Office on Ximian GNOME



hp
invent