# 4mb−Laptops

# Table of Contents

# Table of Contents

# 4mb Laptop HOWTO


## Bruce Richardson <brichardson@lineone.net>


25 March 2000

---


*How to put a "grown−up" Linux on a small−spec (4mb RAM, <=200mb hard disk) laptop.*


---

## 1.Introduction



## 2.The Laptops



## 3.Choices Made

# 4.The Pre−installation Procedure



# 5.The Installation



# 6.Conclusion

# 7.Appendix A:



# 8.Appendix B: Resources relevant to this HOWTO

# 1.Introduction

## 1.1 Why this document was written.

I got my hands on two elderly laptops, both with just 4mb RAM and small (<=200mb) hard drives. I wanted to install Linux on them. The documentation for this kind of laptop all recommends installing either a mini−Linux or an old (and therefor compact) version of one of the professional distributions. I wanted to install an up−to−date professional distribution.

## 1.2 What use is a small laptop?

Plenty. It isn't going to run X or be a development box (see Which components to install?) but if you are happy at the console you have a machine that can do e−mail, networking, writing etc. Laptops also make excellent diagnostic/repair tools and the utilities for that will easily fit onto small laptops.

## 1.3 Why not just upgrade the laptop?

Upgrading old laptops is not much cheaper than upgrading new ones. That's a lot to spend on an old machine, especially considering that the manufacturer isn't supporting it any more and spare parts are hard to find.

## 1.4 What about 4mb desktop machines?

The procedure described in this document will work perfectly well on a desktop PC. On the other hand, upgrading a desktop machine is far easier and cheaper than upgrading a laptop. Even if you don't upgrade it, there are still simpler options. You could take out the hard disk, put it in a more powerful machine, install Linux, trim it to fit and then put the disk back in the old machine.

## 1.5 What this document doesn't do.

This document is not a general HOWTO about installing Linux on laptops or even a specific HOWTO for either of the two machines mentioned here. It simply describes a way of squeezing a large Linux into a very small space, citing two specific machines as examples.

## 1.6 Where to find this document.

The latest copy of this document can be found in several formats at http://website.lineone.net/~brichardson/linux/4mb_laptops/.

## 1.7 Copyright

details.

Toshiba and T1910 are trademarks of Toshiba Corporation. Compaq and Contura Aero are trademarks of Compaq Computer Corporation.

# 2.The Laptops

This section describes the laptops that I have used this procedure on, the problems faced when installing Linux on them and the solutions to those problems (in outline).

## 2.1 Basic Specifications

### Compaq Contura Aero

- 25MHz 486SX CPU
- 4mb RAM
- 170mb Hard Disk
- 1 PCMCIA Type II slot
- External PCMCIA 3.5" Floppy drive

> The PCMCIA floppy drive has a proprietary interface which is partly handled by the Aero's unique BIOS. The Linux PCMCIA drivers can't work with it. According to the PCMCIA−HOWTO, if the drive is connected when the laptop boots it will work as a standard drive and Card Services will ignore the socket but it is not hot−swappable. However, I found that the drive becomes inaccessible as soon as Card Services start unless there is a mounted disk in the drive. This has implications for the installation process − these are covered at the relevant points.

### Toshiba T1910

- 33MHz 486SX CPU
- 4mb RAM
- 200 mb Hard Disk
- Internal 3.5" Floppy drive
- 1 PCMCIA Type II/III slot

## 2.2 The Problem

The small hard disks and the lack of an internal floppy on the Aero make the installation more tricky than normal but the real problem is the RAM. None of the current distributions has an installation disk that will boot in 4mb, not even if the whole hard disk is a swap partition.

The standard installation uses a boot disk to uncompress a root−partition image (either from a second floppy or from CD−ROM) into a ram−disk. The root−image is around 4mb in size. That's all the RAM available in this scenario. Try it and it freezes while unpacking the root−image.

## 2.3 The Solution

The answer is to eliminate the ram−disk. If you can mount root on a physical partition you will have enough memory to do the install. Since the uncompressed ram−disk is too big to fit on a floppy, the only place left is on the hard disk of the laptop. The steps are:

1. Find something that will boot in 4mb ram and which can also create ext2 partitions.
2. Use it to create a swap partition and a small ext2 partition on the laptop's hard disk.
3. Uncompress the installation root−image and copy it onto the ext2 partition.
4. Boot the laptop from the installation boot−disk, pointing it at the ext2 partition on the hard disk.
5. The installation should go more or less as normal from here.

The only question was whether a distribution that wouldn't install (under normal circumstances) on the laptops would run on them. The short answer is "Yes".

If you're an old Linux hand then that's all you need to know. If not, read on − some of the steps listed above aren't as simple as they look.

---

# 3.Choices Made

This section describes the choices available, which options are practical, which ones I decided on and why.

## 3.1 What to use to create the initial root partition?

The best tool for this is a mini−Linux. There's a wide selection of small Linuces available on the net, but most of them won't boot in 4mb RAM. I found two that will:

***SmallLinux http://smalllinux.netpedia.net/***

SmallLinux will boot in as little as 2mb RAM but its root disk can't be taken out of the drive, which is a shame since otherwise it has everything we need (i.e. fdisk, mkswap and mkfs.ext2). SmallLinux can create the needed partitions but can't be used to copy the root partition.

***muLinux http://sunsite.auc.dk/mulinux/***

muLinux will boot in 4mb but only in a limited single−user mode. In this mode fdisk and mkswap are available but mkfs.ext2 and the libraries needed to run it are on the /usr partition which is not available in maintenance mode. To use muLinux to do the whole pre−installation procedure the files needed to create ext2 file−systems must be extracted from the usr disk image and copied onto a floppy.

This gives the option of either using SmallLinux to create the partitions and muLinux to copy the root partition or using muLinux to do the whole job. Since I had two laptops I tried both.

# 3.2 The Distribution

It didn't take much time to choose Slackware. Apart from the fact that I like it but haven't used it much and want to learn more, I considered the following points:

- Slackware has possibly the most low−tech DIY install of all the major distributions. It is also one of the most flexible, coming with a wide range of boot−disk kernels to suit many different machines. This makes it well suited to the kind of hacking about required in this scenario.
- Slackware supports all the methods listed in Which Installation method to use?.
- Slackware is a distribution designed by one person. I'm sure Patrick Volkerding won't object if I say this means its configuration tools are simpler and more streamlined. In my opinion this makes the job of trimming the installation to fit cramped conditions easier.

Version 7.0 was the latest version when I tried this so that's what I used.

## But I don't like Slackware!

You don't have to use it. I can't answer for all the distributions but I know that Debian, Red Hat and SuSE offer a range of installation methods and have an "expert" installation procedure

Does Debian do any other kind?

which can be used here. Most of the steps in this document would apply to any of the distributions without change.

If you haven't used the expert method with your preferred distribution before, do a trial run on a simple desktop machine to get the feel of it and to explore the options it offers.

# 3.3 Which installation method to use?

***Floppy Install***

This means churning out 15 floppies − which only gives you an absolute minimal install and requires a second stage to get the apps you want on. It's also *very* slow on such low−spec machines. This is a last resort if you can't make the others work.

***Parallel−port Install***

Where the parallel port has an IDE device, parallel cable or pocket ethernet adaptor A pocket lan adaptor installation onto these machines will be *very* slow. attached. This would be a good choice for the Aero, leaving the PCMCIA slot free to run the floppy drive.

***PCMCIA Install***

As above, this could be a CD−ROM or network install. This would be the best method for the T1910 − on the Aero it's a bit more awkward.

***ISA/PCI Ethernet Install***

Not an option for the laptops, obviously, but included in case your target machine is a desktop PC.

The tools I had to hand dictated a PCMCIA network install. I will point out where steps differ for the other methods. Whichever method you choose, you need to have a higher−spec machine available − even if only to create the disks for a floppy install.

# 3.4 Partition Layout

## Basic Requirement

This procedure requires at least two Linux Native partitions in addition to a Swap partition. Since one of the ext2 partitions will be in use as temporary root during the installation it will not be available as a target partition and so should be small − though no smaller than 5mb. It makes sense to create for this a partition that you will re−use as /home after installation is complete. Another option would be to re−create it as a DOS partition to give you a dual boot laptop.

## How complex a layout?

There isn't room to get too clever here. There is an argument for having a single ext2 partition and using a swap file to avoid wasting space but I would strongly urge creating a separate partition for /usr. If you have only one partition and something goes wrong with it you may well be faced with a complete re−installation. Separating /usr and having a small partition for / makes disaster recovery a more likely prospect. On both machines I created 4 partitions in total:

1. A swap partition −− 16mb on the T1910, 20 on the Aero (I'm more likely to upgrade the memory on the Aero).
2. /home (temporary root during installation) −− 10mb
3. / −− 40mb on the T1910, 30mb on the Aero.
4. /usr −− All the remainder.

In addition, the Aero uses hda3 for a 2mb DOS partition containing configuration utilities. See the Aero FAQs for details.

## 3.5 Which components to install?

The full glibc libraries alone would nearly fill the hard disks so there's no question of building a development machine. It looks as if a minimal X installation can be squeezed in but I'm sure it would crawl and I don't want it anyway. I decide to install the following (for a full listing see Appendix A):

- The core Linux utilities
- Assorted text apps from the ap1 file set:
- Info/FAQ/HOWTO documentation
- Basic networking utilities
- The BSD games

This selection matches the kind of machine described in What use is a small laptop?.

---

# 4.The Pre−installation Procedure

This section covers creating a swap partition and a temporary root partition on the laptop's hard disk. Nothing here is Slackware−specific.

## 4.1 muLinux Preparation

If you are going to use only muLinux to for this procedure then you need to prepare a disk with mkfs.ext2 and supporting libraries on it. From the muLinux setup files uncompress USR.bz2 and mount it as a loop file−system. If you are in the same directory as the USR file and you want to mount it as /tmpusr then the sequence for this is:

```
losetup /dev/loop0 USR
mount −t ext2 /dev/loop0 /tmpusr
```

>From there copy mkfs.ext2, libext2fs.so.2, libcomerr.so.2 and libuuid.so.1 onto a floppy.

## 4.2 Prepare the installation root files.

Select the root disk you want − I used the color one with no problems but the text one would be slightly faster in these low memory conditions. Uncompress the image and mount it as a loop device. The procedure is the same as in the above section but the root disk image is a minix file−system.

Next you need 3 1722 floppies or 4 1440 floppies with ext2 file−systems − it's better with 1722 disks as you don't need to split the /lib directory. Give one floppy twice the default number of inodes so it can take the

/dev directory. That's 432 nodes for a 1722 disk or 368 for a 1440. If you specify /dev/fd0H1722 or /dev/fd0H1440 then you don't have to give any other parameters so for a 1722 disk do

```
mke2fs −N 432 /dev/fd0H1722
```

If you have mounted the root image as /tmproot and the destination floppy as /floppy then cd to /tmproot. To copy the dev directory the command is

```
cp −dpPR dev/* /floppy/
```

For the other directories with files in (bin, etc, lib, mnt, sbin, usr, var) it's

```
cp −dpPr directoryname/* /floppy/
```

Don't bother with the empty ones (floppy, proc, root, tag, tmp) because you can simply create them on the laptop. boot and cdrom are soft links pointing to /mnt/boot and /var/log/mount respectively − you can also create them on the laptop.

# 4.3 Create the partitions.

## Mini−Linuces and ext2 file−systems − an important note.

To save space, small−Linux designers sometimes use older libc5 librariesand where they do use up−to−date libc6 they leave out may of the options compiled into full distributions, including some optional features of the ext2 file−system. This has two consequences:

- Trying to mount ext2 disks formatted using a modern Linux system can generate error messages if you mount them read−write. Be sure to use the −r option when mounting floppies on the laptops.
- It is not wise to use the mkfs.ext2 that comes with the mini−Linux to create file−systems on the partitions into which SlackWare will be installed. It should only be used to create the file−system on the temporary root partition. Once installation is complete this partition can be reformatted and re−used.

## Procedure

If installing on an Aero, make sure the floppy drive is inserted before switching on and do not remove it.

1. Boot from the mini−Linux

With muLinux, wait until the boot−process complains about the small memory space and offers the option of dropping into a shell − take that option and work in the limited single−user mode it gives you.

2. Use fdisk to create the partitions.
3. Reboot on leaving fdisk (with muLinux you may simply have to turn off and on again at this point).
4. Use mkswap on the swap partition and then activate it (this will make muLinux much happier).
5. If using muLinux then mount the extra floppy created in muLinux Preparation, copy mkfs.ext2 into /bin and the libraries into /lib.
6. Use mkfs.ext2 to create the file−system on the temporary root partition.
7. If you have been using SmallLinux, shut down and reboot using muLinux. Don't forget to activate the swap partition again.
8. muLinux will have mounted the boot floppy on /startup − unmount it to free the floppy drive.
9. Now mount the temporary root partition and copy onto it the contents of the disks you created in Prepare the installation root files. Do not be alarmed by the error messages: if, for example, you copy usr from the floppy to the temporary root partition by typing "cp −dpPr usr/* /tmproot/" then you'll get the error message "cp: sr: no such file or directory". Ignore this, nothing is wrong.
10. cd to the temporary root partition and create the empty folders (floppy, proc, root, tag, tmp) and the soft links boot (pointing to mnt/boot) and cdrom (to var/log/mount).
11. Unmount the temporary root partition − this syncs the disk.
12. You can simply turn off the machine now.

---

# 5.The Installation

This section does not give much detail on the Slackware installation process. In fact, it assumes you are familiar with it. Instead, this section concentrates on those areas where special care or unusual steps are required.

## 5.1 Boot the machine

Make a boot−disk from one of the images. I recommend you use bareapm.i on a laptop and bare.i on a desktop − unless you have a parallel−port IDE device (pportide.i). Boot the laptop from it. When the boot: prompt appears, type "mount root=/dev/hdax" where x is the temporary root partition. Log in as root. Then activate the swap partition.

## 5.2 Floppy/Parport CD−ROM Install

In both these cases, no extra work should be necessary to access the installation media. Simply run setup.

## 5.3 Network/PCMCIA Install

Slackware has supplementary disks with tools for these and instructions for their use greet you when you log in. Use the network disk on a desktop PC with ethernet card or a laptop with pocket ethernet adaptor. Use the PCMCIA disk for PCMCIA install. Once your network adapter/PCMCIA socket has been identified, run setup.

### PCMCIA install on the Aero

The Slackware installation process runs the PCMCIA drivers from the supplementary floppy. Because the Aero has a PCMCIA floppy drive, this means you can't remove the floppy drive to insert the PCMCIA CD−ROM/ethernet card. The solution is simple: the Slackware PCMCIA setup routine creates /pcmcia and mounts the supplementary disk there, so

1. Create the /pcmcia directory yourself
2. Mount the supplementary disk to /mnt. Be sure to specify the type as vfat − if you don't, it'll be incorrectly identified as UMSDOS and long filenames will be mis−copied.
3. cd /mnt;cp −dpPr ./* /pcmcia/
4. Unmount the floppy.
5. Run pcmcia. When the script complains that there is no disk in the drive simply hit Enter: Card Sevices will start. Connect your PCMCIA device and hit Enter.
6. Run setup

## 5.4 Set−up

The Slackware set−up program is straightforward. Start with the Keymap section and it'll take you forward step by step.

### AddSwap

You do need to do this step so it can put the correct entry in fstab but make sure it doesn't run mkswap − you're already using the partition.

### Target

In this section Slackware asks which partitions will be mounted as what and then formats them if you want.

The safest bet here is to leave your temporary root partition out altogether and just edit fstab later once you know you don't need it for it's temporary purpose anymore. If you're going to reuse it as /home then it is OK to designate it as /home − obviously, don't format it now! If you intend to re−use it as a part of the directory structure that will have files placed in it during installation (/var, for example) then you absolutely must ignore it in this step: after the installation is complete you can move the files across.

### Select

Here you choose which general categories of software to install. I chose as follows:

- A − Base Linux System
- AP −Non−X applications
- F − FAQs and HOWTOs
- N − Networking tools and apps
- Y − BSD games collection

I wouldn't recommend adding to this − if anything, prune it back to A, AP and N. That gives you a core Linux setup to which you can add according to your needs.

### Install

Choose the Expert installation method. This allows you to select/reject for installation individual packages from the categories you chose in the Selection step. Appendix A goes through the precise choices I made .

This part takes about 3 hours for a PCMCIA network install. You are prompted to select individual packages before the installation of each category, so you can't just walk away and leave it to run through.

### Configure

Once the packages are all installed, you are prompted to do final configuration for your machine. This covers areas like networking, Lilo, selecting a kernel etc. Some points to look out for:

- If you did a PCMCIA install, don't accept the offer to configure your network with netconfig. This will ruin your pcmcia networking. Wait until you've rebooted and then edit /etc/pcmcia/network.opts
- This is the point where you should install a kernel. For a laptop the bareapm kernel is best, for a desktop simply the bare one.

### Exit

The set−up process is finished but you are not. **Do not reboot yet!** There is another vital step to complete.

## 5.5 Pre−reboot Configuration

On a normal machine you would simply reboot once the installation is complete. If you do that here you may have to wait 6 or 8 hours for a login prompt to appear and another half hour to get to the command prompt. Before rebooting you need to change or remove the elements that cause this slowdown. This involves editing config files so you need to be familiar with vi, ed or sed.

At this stage your future root partition is still mounted as /mnt so remember to at that to the paths given here.

***/etc/passwd***

Edit this to change root's login shell to ash. ash really is the only practical login shell for 4mb RAM.

***/etc/rc.d/rc.modules***

Comment out the line 'depmod −a'. You only need to update module dependencies if you have changed your module configuration (recompiled or added new ones, for example). On a standard system it only takes a second or two and so it doesn't matter that it's needlessly performed each time. On a 4mb laptop it can take as much as 8 hours. When you do change your module set−up you can simply uncomment this line and reboot. Alternatively, change this part of the script so that it will only run if you pass a parameter at the boot−prompt. For example:

```
if [ "NEWMODULES" == "1" ] ; then      depmod −a fi
```

***/etc/rc.d/rc.inet2***

This script starts network services like nfs. You probably don't need these and certainly not at start−up. Rename this script to something like RC.inet2 − that will stop it from being run at boot and you can run it manually when you need it.

***/etc/rc.d/rc.pcmcia***

On the Aero you should also rename this script, otherwise you'll lose the use of your floppy drive on start−up. It's worth considering for any other small laptop as well − you can always run it manually before inserting a card.

Once these changes have been made, you are ready to reboot.

## 5.6 Post−reboot Configuration.

If you made the changes recommended in section Pre−reboot configuration then the boot process will only take a few minutes, as opposed to several hours. Login as root and check that everything is functioning properly.

### Re−use the temporary root.

Once you are sure the installation is solid you can reclaim the partition you used as the temporary root. Don't just delete the contents, reformat the filesystem. Remember, the mke2fs that came with the mini−Linux is out of date.

If you intend to re−use this partition as /home, remember not to create any user accounts until you have completed this step.

### Other configuration tweaks.

In such a small RAM space, every little helps. Go through SlackWare's BSD−style init scripts in /etc/rc.d/ and comment out anything you don't need. Have a look at Todd Burgess' Small Memory mini−HOWTO http://eddie.cis.uoguelph.ca/~tburgess/ for more ideas.

---

# 6.Conclusion

That's it all done. You now have a laptop with the core utilities in place and 50 to 70mb spare for whichever extras you need. Don't mess it up because it's a lot easier to modify an existing installation on such cramped old machines than it is to start from scratch again.

---

# 7.Appendix A:

This appendix lists which packages (if any) from each category might be included in the installation and gives my reasons for including or omitting them. I made no attempt to install X so those categories are ignored.

Although this appendix refers specifically to the Slackware distribution it can be used as a guide with any of the major distributions.

# 7.1 A − Base Linux System

Most of the packages in this category are essential, even those that aren't listed as required by the Slackware set−up program. Because of this, I've listed those packages that I felt could reasonably be left out rather than all the non−compulsory packages that I installed.

## Packages considered for omission:

***kernels (ide, scsi etc.)***

There's no need to install any of these, you get a chance to select a kernel at the very end of the installation process.

***aoutlibs***

This is only needed if you intend to run executables compiled in the old a.out format. Omitting it saves a lot of space. Omitted.

***bash1***

Bash2 (simply called bash in the Slackware package list) is required for the Slackware configuration scripts but there are a lot of scripts that need bash1. I included it.

***getty***

agetty is Slackware's default getty, this package contains getty and uugetty as alternatives. Only include it if you need their extra functionality. Omitted.

***gpm***

Personally, I find this very useful at the console (and the Aero's trackball is very handy) but it's not essential. Included.

***icbs2***

Not needed. Omitted.

***isapnp***

No use here. Omitted.

***loadlin***

Not needed with the setup described here − unless your old laptop has some peculiarity that requires a DOS driver to initialise some of its devices. Omitted.

***lpr***

You could argue that you can do your printing from whichever desktop is nearest but I always find it useful to be have printing capabilities on a laptop. Included.

***minicom***

Not a compulsory include but I want the laptop to do dial−up connection. Very handy. Included.

***pciutils***

Not needed on these old laptops. Omitted.

***quota***

Not vital but it can be used to set limits that stop you from overflowing the limited space available in these laptops. Included.

***tcsh***

I recommend using ash as your login shell. Only include this if you need it for scripts. Omitted.

***umsprogs***

You can leave this out and still be able to access UMSDOS floppies. Omitted.

***scsimods***

No use on these laptops. Omitted.

***sysklogd***

This can interfere with apmd but it does provide essential information. Included.

## Packages installed:

aaa_base, bash, bash1, bin, bzip2, cpio, cxxlibs, devs, e2fsprog, elflibs, elvis, etc, fileutils, find, floppy, fsmods, glibcso, gpm, grep, gzip, hdsetup, infozip, kbd, ldso, less, lilo, man, modules, modutils, pcmcia, sh_utils, shadow, sudo, sysklogd, sysvinit, tar, txtutils, util, zoneinfo

Combined size: 33.4

## 7.2 AP − Non−X Applications

None of these packages are, strictly speaking, essential − although ash is really required for sensible operation in 4mb. Leaving them all out could save the vital space for you to squeeze in your favourite app. I selected a minimal set of tools that I don't like to do without.

### Packages considered for inclusion:

***apsfilter***

Not much point having printing if you can only print text files. Included.

***ash***

This is *the* shell for low−memory machines, only taking up 60k. Use it as the default login shell unless you like waiting 10 seconds for the command prompt to reappear each time. Included.

***editors (jed, joe jove vim)***

elvis is the default Slackware editor and a required part of the installation. If, like me, you are a vi fan then that's all you need: installing vim would be wasteful duplication given the space restrictions. If you can't stand vi and need a more DOS−style editor then joe is small. Emacs fans with some self−discipline might consider jed or jove rather than pigging out on the full−size beast. Omitted.

***enscript***

If you already have apsfilter you don't really need this. Omitted.

***ghostscript***

Including the fonts this comes to about 7.5mb. One to leave until after the core installation, then consider if you need it. Omitted.

***groff***

Needed for the man pages. Included.

***ispell***

Not an essential butvery useful to the overenthusiastic touch−typist. included.

***manpages***

Included!

***mc***

Slackware offers a lightweight compilation of mc but I'm happier at the command prompt. Omitted.

***quota***

Not necessary on what is not a multi−user machine but you may,like me, find it handy to stop you from forgetfully wasting the little space you have. Included.

***rpm***

Don't bother. If you do have an rpm that you would like to squeeze in, use rpm2tgz on a desktop machine to turn it into a tgz package − then you can use the standard Slackware installation tools. Omitted.

***sc***

A useful little spreadsheet packed very small. Included.

***sudo***

Not essential but I find it useful here: it's a cramped environment and an awkward reinstall if you mess things up − sudo helps create user profiles with the power to do the things you need without carelessly wiping your disk. Included.

***texinfo***

Info documentation. Included.

***zsh***

Leave this out unless you're addicted to it or have scripts that must use it. Omitted.

### Packages installed:

apsfilter,ash, diff, groff, ispell, manpages, quota, sc, sudo, texinfo

Combined size: 8.1 mb

## 7.3 D − Development Tools

You could fit C or C++ into this space but the glibc library package is too big, so some pruning would be needed. Do the main installation first and then try it.

There is room for Perl and Python.

### Packages installed:

None

## 7.4 E − Emacs

I don't use Emacs and so saved myself some space. On the other hand, if you are an Emacs fan then you probably use it for e−mail, news and coding so you'll claim some of that space back by omitting other packages.

If you do want Emacs it might be an idea to leave this out while doing the core installation. Once the laptop is up you can try fitting in what you want/need at your leisure.

### Packages installed:

None.

## 7.5 F − FAQs and HOWTOs

If you know it all you don't need these. I installed the lot.

### Packages installed:

howto, manyfaqs, mini

Combined size: 12.4 mb

## 7.6 K − Kernel Source

You can just squeeze it in. If all you want to do is read the source, go ahead.

### Packages Installed:

None

## 7.7 N − Networking Tools and Apps

These packages were selected to provide core networking tools, dial−up capability, e−mail, web and news.

### Packages installed:

dip, elm, fetchmail, mailx, lynx, netmods, netpipes, ppp, procmail, trn, tcpip1, tcpip2, uucp, wget

Combined size: 15.1 mb

## 7.8 Tetex

Another set that will barely squeeze in. I can't say how it would run in the space available.

### Packages installed:

None

## 7.9 Y − BSD Games Collection

I'm addicted to several of these. If I really need that last 5mb they can go.

### Packages installed:

bsdgames

Combined size: 5.4 mb

## 7.10 End result

In total the installed packages plus kernel took up about 75mb of disk space of which 19.5mb was in the root partition and 55.5 in /usr. On the Aero that left 39mb in /usr, 74mb on the T1910.

---

# 8.Appendix B: Resources relevant to this HOWTO

***Linux Laptop HOWTO***

  http://www.snafu.de/~wehe/Laptop−HOWTO.html

***Small Memory mini−HOWTO***

  http://eddie.cis.uoguelph.ca/~tburgess/

***Linux on Laptops***

  http://www.cs.utexas.edu/users/kharker/linux−laptop/ HOWTOs and installation FAQs for a wide range of machines.

***Linux T1910 FAQ***

  http://members.tripod.com/~Cyberpvnk/linux.htm

***Linux Contura Aero FAQ***

http://domen.uninett.no/~hta/linux/aero−faq.html

***Contura Aero FAQ***

http://www.reed.edu/~pwilk/aero/aero.faq Comprehensive FAQ on all aspects of the Contura Aero compiled by the moderators of the Aero mailing list. Good Linux section .

---