Linux Installation Strategies mini−HOWTO

Tobby Banerjee

tobbyb@usa.net

Philippe Martin – Conversion to DocBook SGML

Revision History

Revision 1.0.1	2001−05−02	Revised by: pm
Conversion to DocBook SGML.		
Revision 1.0	2000−12−20	Revised by: tb
Initial release.		

This document discusses a few installation strategies for those who intend to dual−boot between Windows and Linux.

Table of Contents

1. License

You are free to do with this document whatever you please but reading it would be a good idea. I am naturally not responsible for any consequence or action arising out of the use of information provided in this document. I am not a very responsible person and you heed me at your own peril.

2. Mail

Should you find this document helpful please feel free to mail me at <tobbyb@hotmail.com>. Microsoft shares will be highly appreciated but short of that flattery will do just fine.

3. The basics

The primary purpose of this document is to make the co−existance between Windows and Linux as simple and uncomplex as possible which means as far as Windows is concerned Linux doesn't exist. It is now possible to install Linux on a Windows partition as a disk image with what is known as the loopback filesystem − RedHat and Mandrake offer this − but it's an extremely bad idea because of the way Windows handles files. Defragmenting the system will become a nightmare and Linux itself will crawl and it will ruin both your Windows and Linux experience. The same applies to the Umsdos filesystem (though Linux won't crawl) and it's best to install Linux on a partition of its own. This document makes a few presumptions but nothing to worry about − just a few.

- You are using Windows 9x (95,98) not Me or 2000.
- You are familiar with computers ie you know how to boot the system using the floppy or cdrom drive
- You know your way around DOS and Windows
- You know what a partition is and want to install Linux on a partition of its own
- You have a large hard disk − upward of `8.4` Gb − which currently holds Windows and on which you intend to install Linux (even those who with smaller disks can read on).

With that out of the way we can begin. The first thing to do is to make space for Linux. Like other Linux wannabes you are likely to be stuck with a huge single partition with Windows on it. This would be your *C* drive. In my case it was hogging `13.5` gigs. You have to make a separate partition for Linux out of this space (which would then be labelled as *D*). You can either use Windows Fdisk which really is not a good idea as it would entail losing all your data and making new partitions from scratch, one for Windows and one for Linux, or you can take the easier and far more convenient way out and use a free and excellent non−destructive partitioning utility called Fips. What makes Fips special is its ability to create a new partition out of the free space in your *C* partition without any loss of data. The latest version of Fips (`2.0`) is available at its homepage http://www.igd.fhg.de/~aschaefe/fips/ and most Linux distributions carry it under the `/dostools` or `/dosutils` directory in the primary cd.

We are going to concentrate on using Fips so if you are going to use Windows Fdisk − which actually is totally unnecessary unless you have just bought a new system and even then it's a better idea to make a single partition with Windows Fdisk, install Windows on it and then use Fips to split it for reasons that I'm going to explain later in this document − or a commercial partition utility like Partition Magic you are on your own.

---

4. A note about hard disks and partitions

A hard disk can have a maximum of `4` primary partitions – don't ask me why, I don't know. Apart from primary partitions a hard disk can also have what is known as an extended partition which inturn can hold a number of logical partitions – I believe the number is close to `15`. The extended partitions are not real partitions like primary or logical partitions in that they don't store data but are actually containers for logical partitions which is where data is actually stored – as you can see it is uneccessarily complex. Thus in Windows *C* is a primary partition and if you used Windows Fdisk – Windows Fdisk will not make more than one primary partition to partition your hard disk – *D* is usually the first logical partition of the extended partition. *E* would be the second logical partition and so on. In Linux things are slightly different. The first primary partition is called `/dev/hda1`, the second primary partition `/dev/hda2` upto the fourth which is `/dev/hda4`. Linux refers to the an extended partition as in the case of a disk with `1` primary and one extended partition as `/dev/hda1` for the primary and `/dev/hda2` for the extended. The logical partitions of the extended partitions are referred to as `/dev/hda5`, `/dev/hda6` and so on. The second hard disk would be referred to as `/dev/hdb`, the third `/dev/hdc` (usually the cdrom drive if set as secondary master) and the fourth and last hard disk as `/dev/hdd` (last because the motherboard has place for a maximum of `4` IDE devices) SCSI devices are referred to as `/dev/sda` – thank god for that.

5. The concepts

What Fips does is make a new primary partition out of the free space in your hard disk. This is an inherently dangerous operation and it's a good idea to back up all your important data lest something go wrong. Having said that Fips has been used successfully by a large number of people in different conditions and unless there is something already wrong with your hard disk, FAT (File Allocation Table) or partition table things will work just fine. You should read the Fips documentation.

Before you set out to partition the hard disk you have to decide how you want to boot into Linux. The default and most common method is to use Lilo – a simple and basic bootloader program that overwrites (after saving) your hard disk's MBR (Master Boot Record) and on booting the system gives you a choice between booting either Windows or Linux. If you select Windows it boots Windows and you are met with the familiar windows screen or if you select Linux it boots Linux and if you don't make a choice in the stipulated time – could be anything from ±5 seconds depending on on how you set it up – it boots the default operating system, Windows or Linux, again depending on how you set it up. You can restore your original MBR anytime, when uninstalling Linux for instance, either by typing **/sbin/lilo `-u`** within Linux or typing **fdisk `/MBR`** in DOS (this will overwrite the MBR with the original Windows MBR) needless to say you would lose the option to boot into Linux after this and thus access to your Linux partition.

The problem with using Lilo especially for those with large disks is you would have to give up a substantial chunk of your disk space to Linux. This is not a flexible solution since while you can access your Windows partition from within Linux – the large MPEG files can thus be stored in Windows and accessed by both operating systems – the alternate is not true.

This is because of an arcane BIOS limitation that limits Lilo to the first `1024` cylinders of the hard disk. Windows and Linux have no such limitations – it has something to do with address translation and LBA which I have yet to fully understand. Suffice to know the BIOS cannot access data that is beyond this `1024` cylinder limit and since Lilo is dependent on the BIOS to boot the OS's nor can it. Thus if you want to use Lilo to boot Linux you got to make sure that the Linux partition is well below the `1024` cylinder limit. My hard disk – and most modern hard disks have more than `1024` cylinders – has `1650` cylinders and using Lilo means making the Linux partition at about the `1000` or `1010` cylinder mark which translates into a whopping `5.2` Gb of `12.9` Gb to Linux and this wasn't working for me since my Linux stuff hardly need more than `2` Gb and my Windows data on the other hand with MP3's and MPEG movies needs a lot more than `7` Gb.

The solution, and in Linux there are always solutions, is Loadlin, an underrated alternative to Lilo which is inexplicably not as popular or widespread, inexplicable because it's far more flexible. If you intend to use Loadlin and for those with large hard–disks it's an excellent alternative you don't have to worry about the `1024` cylinder limit. Even better you can launch Linux from your Windows desktop and modify your `autoexec.bat` file to give you an option to boot into either of the OS's during boot time not unlike Lilo. You are then free to size your partitions as you please. In my case a `2.2` Gb extended partition with `3` logical partitions. That's right `3` partitions. It's normal to make three partitions, one for the Kernel boot files, one for the root partition and one for the swap partition but more on that later.

If you must use Lilo but are unwilling to give up a lot of space there is a solution. Place the Linux partition in the middle of the disk. You have to split the new partition you just created using Fips into `2` (you would have to use Fips once again for this) the first partition which falls within the `1024` cylinder limit can be used for Linux and the second you can use for Windows. Thus your hard disk will then have `3` partitions the first and third for Windows and the one in the middle for Linux. I don't like this arrangement but it's up to you. (At the moment we are refering to the Linux partition as one but during the installation you will split it into `3` using the Linux tools) You can also make a small `15−20` Mb `/boot` partition and place it before the `1024` cylinder mark and the rest of the installation anywhere on the hard disk but again I quite don't like this.

6. Partitioning

Now we are ready to actually partition the hard disk. First make a DOS boot disk (type **format a: /s** at the DOS prompt or in DOS). Put the following three files in the disk, `fips.exe`, `restorrb.exe` and `errors.txt`. Defragment your hard disk – this puts all the data at the beginning leaving enough space for Fips to create a new partition from. If you use Norton SpeedDisk select the unfragment free space option. Norton usually puts data at the end of the disk and this will prevent Fips from creating a new partition. Run Scandisk and reboot using the bootdisk you just created.

Type **fips** at the prompt and Fips will show you your partition table with a warning about partition table inconsistency. If your disk has more than `1024` cylinders its usually safe to ignore this warning. Next Fips will ask you is you want to proceed – the program is extremely meticulous – answer yes obviously and when prompted to save a backup of your partition table and boot sector answer yes – this step is absolutely vital as it allows you to restore your hard disk to its unpartioned single partition state should anything go wrong. It also comes in handy if you want to uninstall Linux or resize your Linux partition. It doesn't matter what happens to the new partition in the meantime, if you have the back up you can restore your hard disk to its original unpartioned state. This reversible operation will work in all circumstances but one; if you format your original (Windows) partition which was split to make space for Linux in the meantime, the original FAT is overwritten with a new and smaller one and the restore operation will not work.

After making the backup you will be presented with a screen with `3` numbers, the first represents your original partition – notice it is at its smallest size, Fips has allocated all free space in the hard disk to the new partition, this is the default behavior. The second column shows the cylinder info which will be the guide for those who plan to use Lilo. The third shows the size of the new partition you are going to create – it is at its largest size. Use the arrow keys to resize the partition, you would just have to reduce the size of the new partition since it is at its maximum to a size you want. For Lilo users the new partition has to be below the `1024` cylinder mark, `1000` is a reasonable beginning for the new partition. After you are satisfied with the size of your new partition press enter, Fips will ask you to confirm and write the new partition table.

Reboot your system. Remember to change the BIOS settings to boot from the hard disk or you will end up rebooting with the floppy. Double click My Computer and you will see the result of all the hard work, there will be a new drive labelled *D* alongside the original *C*. Don't touch *D*, run scandisk on your original partition (*C*) – hopefully there will be no errors – and at this point you are ready to install Linux.

Some of the Linux documentation mistakenly ask you to delete the newly created partition using Windows Fdisk. Do not commit that blunder because if you do the Linux partitions will show up in My Computer when you reboot after installing Linux and since Windows obviously cannot read these drives and the ext2 filesystem, using Explorer and acessing your hard disk in Windows will become a time consuming and destabilizing affair. So don't do it.

7. Installation

Reboot using your Linux Installation CD − I use Redhat 6.2 which is fairly straight forward and I will focus on it. Most Linux distributions have really worked on their installation programs and you shouldn't have a problem. Just have the details about your hardware in hand, the horizontal and vertical refresh rate of your monitor in case the installation program doesn't automatically detect it (you will find the exact figures in the manual), the video card, its memory type (SDRAM or SGRAM) and size, and your mouse type − PS2 or Serial. As I mentioned earlier you will have to make about 3 partitions for Linux, you can make more but 3 is the recommended number. There are 3 main installation types, *Workstation*, *Server* and *Custom*. The Server class install erases everything in the hard disk so beware. The workstation and custom installs have sub−classes but that's for you to explore.

Loadlin users have to select custom installation and dump Lilo when given the oppurtunity because the other types of installations install Lilo by default and without asking and we dont want that do we? Lilo users have no restrictions; you can opt for whichever install type you like.

---

8. The Lilo way

Lilo fans can use Disk Druid which is RedHat's default partitioning utility – yes we haven't finished with partitioning just yet. First delete the second partition which will show up in Disk Druid's screen as `/dev/hda2`. Then create a new partiton, set its mount point to `/boot` and size to `16` Mb. Create the second partition and set its mount point to `/`(that's called the root partition) and size to *growable*. This tells Disk Druid to reclaim all free space after you have finshed partititoning . Create the third partition and set its mount point to swap and size to about `127` Mb presuming you have `64` Mb RAM, those with less can make a `64` Mb swap partition.

9. The Loadlin way

Those who intend to use Loadlin however cannot use Disk Druid, they will have to use Linux Fdisk which may have an intimidating reputation but is actually pretty simple. When presented with the installation type screen select custom and with fdisk (you will find a small button with fdisk on it at the top of the screen). For some reason Disk Druid doesn't let you proceed with the installation if your `/boot` partition is above the `1024` cylinder limit so if you try to use it to create the partitions the install program will refuse to proceed. Fdisk is a pretty straight forward program and easy to use too. First you have to delete the partition you just created using Fips. Many people don't understand why they have to delete the partition they just created at considerable risk. This new partition is actually a FAT32 partiton which Linux cannot use and when you delete it it doesn't revert to being a part of *C* but is actually UNPARTITIONED FREE SPACE which you can use to create new partitions.

You have to understand a few Fdisk operations, typing **m** lists all the options, typing **p** at the prompt shows you the current partition table, **d** deletes a partiton, **n** creates a partition, **q** quits without saving changes so if you make any mistakes there is nothing to worry about just type **q** and start all over again, **w** writes changes to the partition table use only after you are sure and **L** shows you the hex numbers for different partition types but you only need to know `2`, Linux native is hex number `83` and Linux swap is hex number `82`. So why do you need the Hex numbers? While Disk Druid allows you to mount a swap partition without going into the hex numbers in Fdisk the only way to make a swap partiton is to give it its correct hex number which is `82`.

Let's start, in Fdisk type **p** at the prompt and you will see the current partition table. Type **d** and select the partition you want to delete which in this case would be `/dev/hda2` (just type `2`) type **p** again and you will see the change reflected in the partition table. There will now be a single FAT32 partition `/dev/hda1` (which is actually *C*). Type **n** to create a new partition and for `type of partition` the choices being *primary* or *extended* select **e**. There is no hard and fast rule about this for instance you could have a primary partition for `/boot` and an extended partition holding two logical partitions `/` and swap. I recommend an extended partition that holds three logical partitions. Now you have to size your extended partition properly since it's a container for the three logical partitions. From now on you just have to respond to Fdisk. It will ask you to set the size of the extended partition that you want to create. This partition will start where your Windows partition ends and ends at the end of the disk. You can either give the size in M (megabytes) in cylinder numbers or in K (kilobytes). Just accept the default for the start of the partition and select the end cylinder number for the end – Fdisk will give you the end cylinder number.

Repeat the operation for the three logical partitions, only this time you would have to select logical instead of extended and the appropriate sizes which are `16` Mb for the boot partition, `127` Mb for the swap partition (the size of the swap partition is variable depending on installed RAM, double your RAM is recommended) and the rest of the free space for the root (`/`) partition. Find your way – it's simple, really! After creating the three partitions type **p** and you will see the partitions you just created. There will be `/dev/hda2` (extended) `/dev/hda5` (logical) `/dev/hda6` (logical) and `/dev/hda7` (logical). There is just one thing left to do, set the hex number for the swap partition. Type **t** and select the `127` Mb partition you created which would be `/dev/hda6` enter `6` for the partition number and `82` for the hex number (you will be prompted) and you have just finished with partitioning. Type **w** after making sure and Fdisk will write the new partition table and exit. The next screen will be the Disk Druid screen our Lilo friends have already seen and you just have to mount the partitions with Disk Druid. Select `/dev/hda5` (the `16` Mb partition) and mount it as `/boot`. Select `/dev/hda7` and mount it as `/` (the root partition)

---

10. Rebooting

The rest of the installation is straight forward and requires no hand holding. Select the packages you want, make the boot disk when given the oppurtunity to (absolutely essential especially for Loadlin users) and reboot the system. Lilo users should press **`dos`** at the Lilo prompt to boot into Windows. Like promised none of the Linux partitions will be visible to Windows and we can all rest in peace for that. Lilo users should reboot the system and boot into linux this time by typing **`linux`** at the Lilo prompt and depending on the choices you made you will either be at the console or the gnome or kde desktop. From one amateur to another it's always a good idea to log into the console first and not graphically and then starting X because if X crashes on you still have access to the system to make amends. Lilo is set to boot into Linux by default after a gap of `5` seconds, you can change the default to Windows by editing the `lilo` file in `/etc`. Look for the line that says *default=linux* and change it to dos. Alternative launch lilo from an xterm and change the asterisk from linux to dos (use the arrow buttons, dummy). That's set now for Lilo users.

---

11. Setting up Loadlin

Loadlin users still have a little work to do. First boot into Linux with the floppy. If you logged in graphically start a terminal emulator like xterm or some other. There are always a couple of these under the utilities menu. Type **mkdir /mnt/dosc**. This is the first step to make your Windows partition accessble to Linux and this is particularly important for Loadlin users as it is the easiest way to tranfer the Linux kernel to your Windows partition. Next start linuxconf from the system menu or the xterm and look for the filesystem menu. Expand accesss local drives and select add. Type **`/dev/hda1`** in the partition field, **`vfat`** in the type field and **`/mnt/dosc/`** in the mount point field and mount the partition. Alternatively you can directly edit the `fstab` file in `/etc` in a text editor (Gedit is a good choice) and add the following line exactly as shown.

```
/dev/hda1     /mnt/dosc     vfat     defaults     0   0
```

This should give you access to your Windows partition, it will be in `/mnt/dosc/`. Copy the linux kernel to your *C* directory and rename it to `vmlinuz`. You will find it in `/boot` and probably called *vmlinuz* followed by your kernel version number in my case `vmlinuz-2.2.14-5.0`. Do not copy the `vmlinuz` with an arrow in it, its a symbolic link. Reboot into windows, get the loadlin package from the Linux cd – it will be in the `/dostools/utils` subdirectory. Unzip it and place `Loadlin.exe` in the *C* directory. Loadlin and the kernel can be anywhere on your windows partition but for the time being we will leave it in root directory of Windows. Now its up to you how you want to boot into Linux. Read the loadlin documentation for the details. The roundabout way is to restart the system in DOS mode and type:

```
C:> loadlin vmlinuz root=/dev/hda7 ro
```

This will boot you into Linux. The easier way is to make a `.bat` file with the same information and placing it on your desktop. Open notepad type **`c:\loadlin c:\vmlinuz root=/dev/hda7 ro`** and save the file as `linux.bat`, right click the file, select advanced, and check the msdos mode and warn settings. Make a shortcut to this file and put it on your desktop. The next time you want to boot into Linux just double click this file and you will be in – talk about convenience.

If you want to be prompted to boot into linux before Windows boots like the Lilo guys you have to make another `.bat` file and call it from your `autoexec.bat` file. It's very simple and the details are available in the Loadlin documentation. The file looks a bit like this.

```
        @echo off
        cls
        echo.
        echo.
        echo.
        echo.
        choice /t:n,5 "Do you wish to boot Linux?"
        if errorlevel 2 goto End
        c:\loadlin c:\vmlinuz  root=/dev/hda6  ro
        End
```

(This little script has been filched from the Loadlin documentation which you would know if you did the sensible thing and read the documentation)

Save this text as `linux.bat` or if there already is a `linux.bat` in your Windows root directory you can place it in another directory or give it another name. Add the following line to your `autoexec.bat` file **`c:\linux`** `'`(or whatever you have named the `.bat` file). The main thing if you are using a `.bat` file is to tell Loadlin where to find the kernel image file (`vmlinuz`). The above `.bat` file gives you the option to boot into Linux before Windows loads by typing **y** (for Linux) or **n** (for Windows) within five seconds at which point it boots Windows. Enjoy.

12. Removing Linux and restoring your Windows partition to its Original state

Depending on whether you are using Lilo or Loadlin the method differs slightly. Lilo users have to first restore their original MBR either by typing **/sbin/lilo** **`-u`** within Linux or typing **fdisk** **`/MBR`** in DOS. The next steps are common for both Loadlin and Lilo users. These are not really neccessary but just to be on the safe side I do them all the same.

Boot using the Linux installation CD and type **`Linux expert`** at the first prompt, answer whatever you wish to the next questions until you get to the install selection screen. Here select *Install* and *With Fdisk* and use Fdisk to delete all Linux partitions. Write the changes to disk and **Ctrl−Alt−Delete** your way out of the installation program or reset your system to reboot. Remember to change the BIOS settings to boot from the hard disk or else you will reboot from the CDROM again.

Reboot once again, this time with the fips disk that has the backup of your partition table and boot sector − you saved it, didn't you? − and type **restorrb** at the prompt. Answer **`yes`** to the question, the program will restore your partition table and boot sector and you can pat yourself on the back for accomplishing the impossible − reclaiming lost partition space! This is all the more reason for making sure that the crucial backup file (`Rootboot.000`) is saved in at least `5` different locations. You can never be too careful. Reboot back into Windows and run scandisk on the newly increased *C* drive. It will report one error in the fsinfo sector which is just a used and free space accounting error. Your Linux data is now naturally lost for ever. Hopefully this procedure won't be neccessary unless you are increasing space allocated to Linux or getting rid of Windows.

---

13. XFree 4.01

It's a good idea to upgrade your Xserver to `4.01`. At the moment only RedHat 7.0 ships with this version of X but it is availabe from the X consortium's (how corny) ftp site – ftp.xfree86.org/. It provides direct acceleration and increased performance for a large number of accelerated cards and those with Nvidia cards are in luck for not only does X 4.01 natively provide excellent support for most of their cards but Nvidia have made available high performance OpenGL accelerated drivers – cheers to them. See XFree's home site – http://www.xfree86.org/ – for more info.

---

14. Tips and Suggestions

- Whatever soundcard you have Alsa drivers will deliver better performance and if like me you have the Yamaha 724 the Alsa drivers are nearly as good as those provided by Yamaha for Windows. You will of course have to compile them for your system but don't let that deter you. Its a simple matter of three commands. The alsa drivers are available at http://www.alsa−project.org/
- Those used to scrolling with their mouse wheel/s will not like Linux if they can't scroll. The good news is you can but it is not set up by default. Add the following line to your `XF86Config` file in `/etc/X11/` in the mouse section **`"ZAxisMapping 4 5"`** (obviously without the quotes though in Xfree 4.01 you would have to enclose the numerical values in quotes) If this doesn't work visit the Linux scroll wheel homepage at http://www.inria.fr/koala/colas/mouse−wheel−scroll/ set up by a curiously named fellow called Colas Nahaboo.
- By default the hard disk is not configured to perform at its optimum and in most cases the difference can be dramatic. DMA, 32 bit IO and multiple sector transfers are disabled by default and you should enable them at the earliest this is of course presuming you have a relatively new hard disk that support these features. As a test you should benchmark your hdd before and after you change the settings just to see the difference tweaking can do to your system. The utility to change the settings is hdparm and you can run it from an xterm or the console but for a permanent solution append the command to your `rc.d` local file in `/etc/rcd`. Before you use hdparm read the documentation carefully. Its a powerful utility and misusing it can trash your disk. First run **hdparm** with the `−i` option to get the info on your hdd, then benchmark it without the changes by running **hdparm `−Tt` /dev/hda**. Then run **hdparm `−c 1 −d 1 −m16 −k 1` /dev/hda**

  hdparm `−d 1` (sets `dma` to `on`) `−m`*x*(this sets multiple transfers where *x* represents the maximum (in my case `16`) that your hdd allows − running **hdparm** with the `−i` option gives you the maximum that your hdd allows) `−c 1` (sets 32 bit io) and `−k 1` keeps the setting. In my case the data transfer shotup from a mere `3.16` to `16.8` Mb/sec!

- A file system driver for the ext2 filesystem for Windows, ext2fs is available from its homepage http://uranus.it.swin.edu.au/~jn/linux/ giving you the added convienience of reading your Linux partitions from within Windows.

15. The State of Linux – What to Expect

Programmers and Enthusiasts – For programmers Linux is already a near perfect solution. No other system gives so offers you a wider array of tools and platforms for development. The flexibility is simply amazing and it doesn't carry a price tag. An added benefit is the sense of community and that's priceless.

Enthusiasts never had it so good. There is aways something new to discover and since the system is totally transparent your chances of understanding computing and programming in general – you will be interested if you aren't already – are infinitely higher than in Windows. Tinkering will never be the same again.

For the home user – There are 5 separate areas here, the internet, gaming, basic word processing, administration and multimedia. We aren't going to talk about ease of use here because the days of an ugly and unwieldy interface are long gone, at the moment as far as usability and sheer visual pizazz is concerned Gnome has taken the pants off Windows so there is nothing to worry about there. File management, installing and uninstalling software and basic administration are more streamlined and centralized and as you use Linux more you probably will get over any prejudice toward the command line which is nothing like the impotent DOS prompt. This is an outrageously powerful shell that will accomplish anything you can think off faster than it takes you to select a button. You might be intimidated at first but its the heart of Linux the control centre, the headquarters, your direct line to the kernel. Hardware configuration is not as simple as in Windows mostly because plug and play has complicated things not eased them but it's not difficult either, the best thing to do is set your BIOS to *non pnp* so that it can configure things especially the sound card, but *read read read*. There is a lot of documentation on your system and with that behind you everything becomes simple. There is no excuse for ignorance.

15.1. The Specifics – Wordprocessing

Corel WordPerfect is good but for the simplest solution try Abiword – http://www.abisource.com/. It's good and bears an uncanny resemblance to MSWord – now now. Both are free and widely available – I don't know about Wordperfect 9 , 8 is free, good and available. Other options are Applixware and Star Office. I haven't used Applix but Star Office is really ugly and slow and just the sort of software that could put you off Linux and its just as bad in Windows. Version 5.1 was tolerable but after Sun took it over its deteriorated and version 5.2 is the pits. Plain text and html editors are sprouting by the day. Most distribution carry a lot of software so look. I'm not much of a spreadsheet person but there is GNUmeric which is not unlike Excel and a glaring inadequacy in printing is now rectified. It looks good. Corel have ported nearly all their apps and they are not as expensive as their Windows counterparts and with the same functionality.

15.2. Multimedia

This is one area where Linux will not be found wanting. There are as many if not more mp3 players than Windows. The most common is XMMS, is somewhat like Winamp and just as good – has some great visualization plugins. Other mp3 players are Gqmpeg, Freeamp, Alsaplayer. That's enough for me. Mpeg1 vcd players are a bit scarce, MTV is good and available – hteztp://www.mtvplayer.com/ free for personal use – a totally free solution is available from lokigames – http://www.lokigames.com/ There are more. There are a few mp2 players movie is good. I' don't have a dvd player so I don't know but Xmovie should tackle mpeg2 streams. Most of my mpeg1 players are GL accelerated and perhaps my vcd playsback as well as Windows Media Player. Ripping is a reality with a lot of options no more

ASPI headaches − most of the Linux apps read digital data directly from the discs and all the popular codecs are available; Xing, Fraunhofer but I use Blade and Lame − two excellent freeware encoders avalible for all platforms. There are also a couple of tools for cdwriting. Check some of the Linux archives.

15.3. Internet

Again things are really happening here though one wishes there were a few more browsers other than Netscape but then Netcape is adequate. There are all the other stuff you will find for Windows, chat utils, instant messaging, ftp, downLoaders etc.

15.4. Games

The problem area is games, there is a project on to port popular games to Linux by Lokigames and they have a number of titles ready − http://www.lokigames.com/ − Quake 2&3 are out, System Shock, Heavygear etc. The thing here is to make sure you have at least Xfree 4.01 and support for your video card.

15.5. Desktop Publishing

That leaves us with DTP and this is one area where Linux could pound Windows but the thrust has been missing. Linux could really be optimized for DTP. Ghostscript − a PostScript RIP − already offers a professional (if you know how) solution for printing and your printer is probably supported. If you already have a postscript printer there is no problem to start with. For image editing there is Gimp. It is nearly as good as Photoshop − and that's some compliment − but for lack of CMYK and colour management support ruling it out of serious print work but it can be a good semi−professional solution and for web work perfect and it's fun to use. Gimp also has a couple of good drivers for Epson printers. Ghostscript has a problem with bitmaps − not all bitmaps but photographs, you could eventually get some good output but it will take some amount of tweaking. As far as text and vector graphics is concerned it's good enough to replace my Epson driver in Windows (it has a Windows version) GhostScript is available at its homepage http://www.cs.wisc.edu/~ghost/

Word processing is already covered. The problem areas are vector editing and page layout. Framemaker for Linux is a solution and CorelDraw is available but they are both commercial offerings. There are vector editors but all are at a nascent stage. Sketch, Killustrator and Impress are coming along but far away from professional solutions like Draw and Adobe Illustrator. For page layout there is Tex but those used to Pagemaker, Quark or InDesign are not going to be amused, its tough to learn and difficult to control and there are way too many issues with things like fonts and its not wysiwyg − we can't live without that.

On a last note at the moment the only difference in productivity for me is in vector graphics, page layout and games. They are perhaps a year away at which point Linux would have over taken Windows − at least as far as I'm concerned − and I would have little reason to dual boot but I think having Windows is in a odd way comforting as is having Linux, perhaps I will always use them both.