Preparing to Take Exploration Computing to Linux

Will Morse and Ed Hutchinson
Anadarko Petroleum Corporation

Introduction:

You will be going to Linux. You may be one of the first or one of the last, but you will go. How easy and cost effective that transition will be depends on how well you plan and how well you prepare. Each E&P company, and in some cases each business unit within an E&P company, will approach this somewhat differently. This paper is not going to give you a plan. It is about what you need to ask yourself and what you need to investigate to develop your company's plan. It is important that you develop that plan soon, before the business runs over you with "entrepreneurial" installations that may not tie back effectively to your entire system. Most large E&P organizations already have some Linux, perhaps only pilots, but the time is now.

Background:

While you would never imagine so from the reports in the media, Linux is just another flavor of Unix. In that way, it is little different from Solaris, IRIX, AIX, HP/UX or any of the other "proprietary" flavors of Unix. Linux is also not unique in that there are other versions of Unix that execute on Intel IA-32 (a.k.a. X86) and clone microprocessors. These include OpenBSD, FreeBSD, Solaris X86 and others. The main thing that is unique and desirable about Linux is that it is an extremely widespread and well supported version of Unix that executes on a variety of hardware, but most importantly, Intel IA-32 and IA-64 and AMD Opteron platforms.

It is often misunderstood that there are flavors of Linux. There are, but not in the widespread version that everyone understands as Linux. The only special version of Linux that is at all important is SELinux (Security Enhanced Linux), which is really more a major set of patches to the standard version.

What varies is not the Linux operating system itself, but the DISTRIBUTION, in other words, the ancillary programs that come with Linux. Red Hat, Mandrake, SuSE, Debian, etc. are DISTRIBUTIONS of the same Linux operating system with a variety of ancillary software. Even these distributions are remarkably similar, varying mostly in systems administration features rather than anything the normal user would have to worry about.

Going to Linux is not just a matter of trading in Sun or SGI and getting Red Hat back, however. There are important issues including hardware, systems, networking, training, transition plans, and most of all cultural changes. Success in implementing Linux requires a good plan and a lot of preparation.

In addition to preparing and planning at your own shop, now is a key time in the development of these applications and it is important to make the need for certain standards known to your suppliers. The most important of these is Linux Standard Base, which is described in detail below.

Motivation for Going To Linux:

It is important to realize just how quickly the petroleum industry is going to Linux. This is not something where you can just wait and see what the other guy does. The motivation can be summarized as:

Be There or Be Square

We have been looking at all our exploration software vendors. This might not include all yours. There are only three "hold-outs" at this point. Everyone else either has a Linux version or is in the process of developing a Linux version.

You should also consider larger picture items. Where are your future systems administrators and, for that matter, users coming from? If a college student gets any Unix training at all, it is most likely that this training will be on Linux. What platform are all the research people using to develop new generation software? Most likely Linux. Software vendors do not want to spend the extra money for Sun or SGI systems any more than clients do. Where do the software vendors want to develop? Most likely Linux.

Linux changes your purchasing workflow.

It is no longer a matter of :

        Our application runs on Sun systems
          =>   We need a Sun platform
              =>   We need to run Solaris
              =>   We probably need to run CDE if we don't have adequately
                      trained systems administration to use FVWM.

It now becomes:

> Our application runs on Linux
>> => It's Linux Standard Base, so it runs on any distribution
>>> => It runs on any IA-32 platform, we can bid this out
>>>> => We need to buy the best bid and the best distribution
>>>> => We could run KDE, GNOME, or FVWM,
>>>> our choice, not our vendors.

You now have truly valid <u>second sources</u> for not only the hardware, but also the operating system and the ancillary software. This creates competition, thus lower prices, and insulates you from the financial and/or technical problems of any one vendor.

A major benefit, especially for Landmark users, is that the Linux version of the Landmark suite runs in <u>24-bit color</u>, not 8-bit color as on the Sun systems. Landmark is coming out with a 24-bit friendly version of its software in October 2003, but by the time all the third-part devkit developers catch up, it will probably be mid 2004 for most of these vendors to come out with 24-bit compatible versions on Sun. As a matter of clarity, Landmark will still be 8-bit color, they will just use an 8-bit slice out of the 24-bit color map rather than requiring the whole system to be 8-bit. This will effectively eliminate flashing.

Many people expect that by going to Linux they will <u>save money</u>. They are generally thinking of the capital cost of the hardware and operating system when they think this. It is true that capital costs will probably be reduced, but it is likely that G&A expense will go up, and the total cost of ownership (TCO) may be around the same or even more. There are a lot of costs associated with managing Linux systems that these people may not be taking into account. You will be going to Linux. You should manage your expectations and those of your management that you will be doing more for the same money or slightly less, don't sell it on capital savings.

Cultural Changes:

Linux is not <u>FREE</u> software, it is <u>FREEDOM</u> software. Your saving in hardware and software are to some extent offset by your increased responsibility for a lot of things that just came with the package before. How large this offset is depends a great deal on how well you plan and prepare.

One change is that the era of "Customer Control" is over. Customer Control is when a vendor sells you something, they now think they own you. They try to make sure you do not buy from anyone else or do anything to leave a competitive gap. You now have a world of hardware, software, support, and other alternatives that no one vendor can lock up, although they will still try.

Another change is that you will have to do more to support yourself. You will no longer call up some support desk and say "I'm having a problem with program P" and wait for them to come up with an answer. You will need to maintain contact with a world of newsgroups, mailing lists, industry contacts, and so on. When you seek support, as often as not your message will be "I had a problem with program P, and here is how I fixed it."

Because there are more alternatives and because your support will be more internal, you will need to have good training. Training can no longer be treated as a perquisite to be taken only if you have spare time.

Planning:

There are many questions you must ask yourself and your organization in order to make an effective plan. This section covers some of the major questions.

- Are you going to use Linux workstations just for a replacement for the existing scientific workstations? Are you going to use Linux to replace the office productivity applications? Are you going to try to get to a single desktop to handle both scientific and office?

  It is very tempting to go for a single desktop, but that is also a much larger project with more things that can go wrong. You will probably be replacing a lot more desktops and training a lot more people. On the other hand, you get it all over at once while you have the skills set up to do it.

  Of course, if you go full across, you have to deal with your legacy Microsoft and Microsoft based applications.

- Do you support a full range user environment like KDE or GNOME (and if so, one or both)? Or do use a more limited environment such as FVWM?

  How much do you want to support? Do you want your Exploration Applications support group to have to be familiar with programs like GIMP, Kivio, and Konqueror? On the other hand, if your users install these applications at home, they might expect your support people to know them as well. How much control do you plan to give your users over their environment and applications?

- Will you be installing applications and/or data locally, or will you use some kind of central filers? If central filers, will these be Linux systems or a commercial solution such as Network Appliances or Auspex?

  Local applications and data can be faster, but you are limited in the amount of space you can give to an individual system. You will have more problems upgrading the applications on a local install rather than across the entire system as with a central filer. It will be harder to share data. It will be harder to make good backups.

On the other hand, most people who are trained on Linux will expect to see local applications and data. Whatever filer solution you pick will involve that much more training for your users and support people.

Another wrinkle is the impending upgrade to NFS version 4 (NFSv4). Most E&P companies are on NFSv3 and some are still on NFSv2. NFSv4 has many security and other enhancements including support for file locking. NFSv4 has had heavy development on Linux. V2 and v3 have some problems on Linux. This might be the time to look into NFSv4 (www.nfsv4.org).

- Will you retain some proprietary Unix systems for various servers? Servers might include license servers, Oracle servers, print/plot servers, NIS or Active Directory servers, ftp and other internet connections, backup and disaster recovery servers, servers for special "legacy" applications (those that have not converted to Linux yet), and others.

  While most of these can probably run on Linux with no problems, it might be easier to convert the desktop systems first and then come back for the servers. On the other hand, the servers probably won't have the graphics issues, and may not have the NFS or kickstart issues, so it might be easier to start with servers. The applications will probably be a major factor in determining which is easier to start with.

  You may also want to change how you do the servers. For instance, you might decide to do all printing via Samba rather than lpr (BSD) or lp (SysV). Printing using Samba uses the Microsoft CIFS (Common Internet Filesystem) mechanism which helps to consolidate all printing to one technology. The Unix printing mechanisms are common points of computer security vulnerabilities and it might well simplify your life to get down to one technology for printing.

- How will you handle backups and disaster recovery?

  A tape drive on each desktop is expensive and probably not very efficient if you have more than five or so desktops. If you use a central filer exclusively, backups will be pretty easy, although time consuming. If you let users store data on their local machine, you may need some kind of a backup daemon that collects this data to a central server to go on backup media. Of course, then how do you restore it. How do you handle a situation when a user transfers to a different department or leaves the company?

  Centralized backup has its own woes, however. You might find yourself needing several backup servers to get the work done in a reasonable time. You also cannot tolerate network outages during the backup window. People often get used to turning off PC's when they leave at night.

- How will you handle both systems administration and user training?

  Traditional Linux certification and training programs expect that all the applications and data are loaded locally, all users and services are defined locally, and the network connection is just to connect to email and web services and other internet services. If you use a more filer centric model, you may need find additional training resources or develop your own training to cover NFS, NIS, etc.

- What is your transition plan? Unless you are a very small company, you probably cannot convert from, say, Solaris on Sun hardware to, say, Red Hat 8.1 on HP hardware over a single weekend. If nothing else, at least one department will have some crunch on at any given time.

  You might want to take steps to make your Solaris and/or IRIX systems look and feel more like Linux systems before the transition. CDE (dtwm) is not very easily changed, but FVWM (www.fvwm.org) can be changed to look a lot like Linux. This may seem like a minor point, but many of the small features of an environment become entrenched habits that drive users crazy.

  A particular concern is the title bar buttons. On a traditional Motif / CDE system, the upper left button (with a hyphen in it) calls a menu, the next to the right-most button (with the little dot) is to iconify, and the right-most (big square) button is to maximize.

  On Linux systems, as well as Microsoft systems, the buttons are all on the right sides and have underscore to iconify, a box to maximize, and an X for the close button.

  If your users are used to the traditional window that looks like:

  and now switch to a window style like

```
user system:directory
> □
```
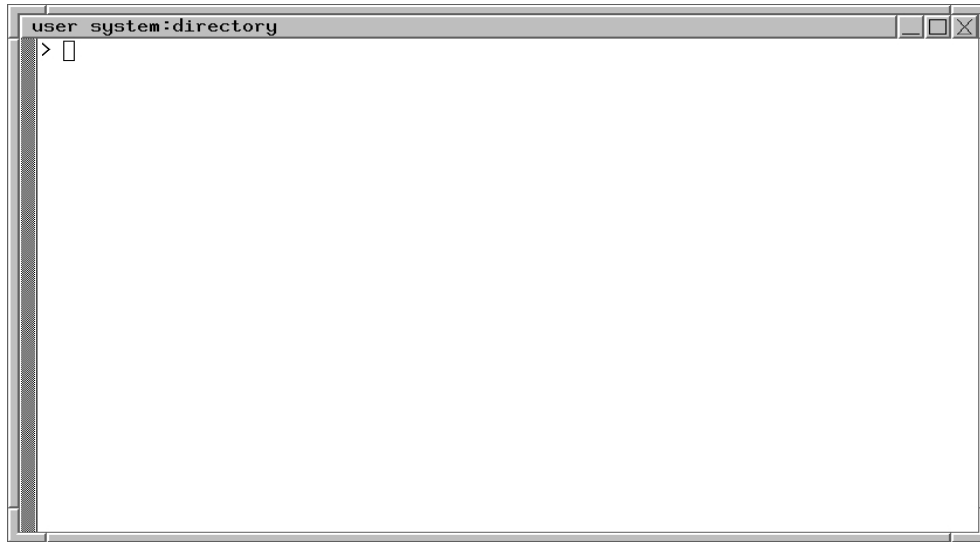
They may have problems of clicking the wrong button by force of habit.

Similarly, your users may still be using the long obsolete "C-shell". Linux uses bash as its standard shell, but "T-shell" – tcsh – is also available. On Linux, the program /bin/csh is actually tcsh. Both bash and tcsh are standard shells on Solaris and IRIX and it might be a good idea to get your users started with these shells on your current system to simplify the transition. These are by-no-means new shells, they have been around since the late 1980's and are quite stable.

You can learn about bash from www.gnu.org/manual/bash/index.html

You can learn about tcsh at www.tcsh.org/Home

You can learn about FVWM at www.fvwm.org

Hardware Generally

The main issues between different sources of IA-32 hardware (HP/Compaq, IBM, Dell, Gateway, white boxes) is not so much the processor, RAM, disks, or what have you. These are pretty much either the same or compatible. There can be a few BIOS issues, but these are unlikely. The big problem is with the framebuffers (graphics cards) and their associated drivers. Network cards are probably the next biggest problem.

There are occasionally issues about what level of IA-32 processor the applications and/or kernel are compiled for.

Linux started out in the days of the "i386" processor and there are still many programs and kernels compiled at that level. These programs and kernels do not take advantage of the more advanced instruction sets and other features of more recent processors. This doesn't need to be a show stopper, but it is something to watch out for. Some vendors are

running their pilot Linux development on hand-me-down PC's that aren't good enough for office applications anymore.

Different E&P software vendors are certifying on different combinations of framebuffers. It is important to pick a framebuffer that all your applications can take advantage of.

Unlike all but the newest Sun and small system SGI framebuffers, it is most common for PC framebuffers to drive both monitors in a mode resembling Xinerama on the Sun systems. In other words, both monitors are included in a single image to the X server. Some framebuffers accomplish this with two different kinds of connectors (one for each monitor), however, which can be a pain.

Vendor FUD:

Some application vendors will try to fool you and say that they have to configure your system for you and that their application is only certified and will only work on a system that they build and configure for you. This is, of course, poppycock.

Worse, they charge you extra for all this "added value", which eats into the whole benefit of reduced cost for the platform.

Worse yet, you lose the benefit of the second source / competitive market.

Don't let your vendor get away with this. If you do, you lose and everyone else loses. Ultimately even the vendor loses because they cut themselves out of those potential customers who want their own configuration.

Selection of a hardware vendor can make a big difference in the cost and effort you need to put in to get to Linux. Most PC hardware vendors have heard of Linux and know it is "the next big thing", but not many of them have any actual experience with it. They may "certify" that their hardware runs a particular level of Linux, but there may be many options they haven't tried because they are unfamiliar with the Unix paradigm. Network Interface cards (NIC's) are a particular area to watch out for.

32-bit versus 64-bit Hardware

The vast bulk of E&P software is being ported to the 32-bit IA-32 architecture. 64-bit systems are a little fresh and the winners have not shown clearly yet. You might choose to stay with 32-bit for the next year or so and then look at 64-bit when we see who wins. Certainly don't hold up the whole conversion to Linux waiting for a 64-bit architecture.

There are basically three 64-bit approaches. These are the Itanium2 from Intel, Hammer from AMD (just released as Opteron), and Yamhill which is Intel's not-so-secret Plan B in case Itanium2 doesn't make it.

Itanium2

Itanium2 (IA-64) looks much more like a RISC architecture than the traditional IA-32. The instructions use 41-bits, providing 241 opcodes. Instructions are in three instruction "clusters" with a five-bit sequencing code. Only one floating-point operation can be in a cluster. There are 128 general purpose registers, 128 floating point registers, 64 "predicate" registers, 8 branch registers and 28 various special purpose registers for a total of 328 registers. There is only one memory address mode. The compiler can provide "hints" and "speculative loads".

Itanium2 has a special, low speed, 32-bit IA-32 compatibility mode. This mode will execute 32-bit applications, but is much slower than a state-of-the-art IA-32 system. This mode is intended as a crutch to get users over to applications compiled for 64-bit. They don't want it to be fast because they want you to get over to 64-bit.

An interesting note is that Itanium2 processors have very good floating point – an area where Intel processors have been criticized in the past, but at some cost in integer and string performance. Most E&P applications are predominantly integer and string, although this is changing as advanced graphics are taking off. Itanium2 is basically built for servers, not desktops.

HP/Compaq has an Itanium2 rack-mount system available with Linux all compiled for 64-bit and with a 64-bit gcc compiler. SGI has a high-end cluster system in the Altix 3000 Family that is also based on 64-bit Linux on Itanum2 with Intel and GNU compilers. Neither of these systems are designed for a desktop, dual monitor, exploration class desktop, but that isn't to say these systems aren't coming.

Hammer / Opteron

Hammer (now released as Opteron) is AMD's offering. Hammer is more like what most people expect in a 64-bit processor – a "fat" 32-bit processor. Hammer is more like an extended IA-32 processor, it is not an Itanium2 clone at all. It is basically a super Athlon with 64-bit extensions.

Hammer uses a hardware assisted, RISC based, X86++ emulation. It has sixteen 64-bit registers. In 32-bit mode it uses the lower half of the bottom eight registers. There is a simple mode switching mechanism so that you can run 32-bit and 64-bit applications at the same time. It is highly optimized for multi-processor designs.

The basic strategy with Hammer is to give you a superior performance with 32-bit applications while giving you a transition path to 64-bit applications. This is probably a good bet in the short term as there are an awful lot of 32-bit applications that will have to be ported, or at least recompiled, to run in 64-bit mode.

<u>Yamhill</u>

Yamhill is Intel's Plan B in case Itanium2 doesn't work out. Details are sketchy, but it appears to be basically similar to AMD's Hammer.

<u>Other 64-bit architectures</u>

Alpha is dead. There are no future versions of Alpha.

Sun SPARC is moribund but still kicking. There is a SPARC-IV chipset coming out which is a slightly cleaned up SPARC-III except there will be two "cores" – processors – on one module. SPARC-V is also coming with a new "pipeline". The likelihood of success for this chipset is pretty questionable. The clock rate will go up, but not as fast as for Intel chips. There is no intention by Sun of supporting Linux for the SPARC. There is a non-Sun effort to expand the SPARC port of Linux, but without Sun pushing it there isn't much future for it. There are no E&P applications being ported to it.

SGI MIPS is moribund also. SGI is moving to Itanium for its heavy duty processing machines (Origin -> Altix class) but is basically staying with the MIPS chip for the heavy duty graphics class (Onyx).

The IBM Power4 and PPC970 systems are the main competition to the Itanium2 and Hammer at this point.

Why you care about 64-bit

32-bit processors are limited to 2 gigabytes of virtual memory for any given process. You can get up to 4 gigabytes by using shared memory, but this is something that has to be compiled into the program and is not for the faint of heart. This is not an important limitation in home computers, game computers, or general business computers. It does start to matter in large web servers, database servers, data mining applications, and of course, in geoscience applications. People who have gotten used to greater-than-32-bit systems with Sun and SGI may be unpleasantly surprised to have to go back to worrying about these things.

The greater the difference in the architecture, the longer it takes to port and the more likely it is for a bug to creep in. Converting to effective 64-bit is not just a simple matter of recompiling the same code. Many older programs have been sloppily written to expect pointers and integers to be the same size. When pointers suddenly change to 64-bits, but the integers being used as pointers do not, there is an opportunity for bugs. If a 32-bit integer is used as a pointer or array index, that array is still limited to 32-bits and thus 4 gigabytes. Making effective use of a 64-bit architecture will require changes in most programs.

For an in-depth and enlightening discussion of 64-bit processors, see

http://www.extremetech.com/article2/0,3973,231,00.asp

Other Hardware Issues that Need to be in Your Plan

- Linux uses the BIOS. This is not much different than Sun using OBP (Open Boot PROM), but there are various versions of the BIOS. It is not so much a hitch, just another variable you need to account for.

- IRQ's – Interrupt Request Line or Interrupt Request Queue. There are sixteen levels of IRQ with various types of hardware for each. This is not an issue for a standard desktop unless you put tape drives and other special devices on the desktop. It needs to be covered in your plan.

- DMA Channels are Direct Memory Access. These are mainly used with disks, tapes, and sound cards.

- Graphics cards heat and power load. This is really the same as with the proprietary Unix systems, just that there are so many more options and alternatives in the Linux / PC world.

- Will the Network Interface Card (NIC) operate with your network upon boot, or do you have to boot it and make some adjustment. If you are going to be using "Kickstart", it will matter. Kickstart is similar to Jumpstart on Solaris, and is used to install the operating system.

- As alluded to previously, there are older versions of the IA-32 architecture going back to the i386.

Operating Environment Issues

Linux is just the Operating System. Red Hat, Mandrake, SuSE, Debian, etc. are Operating Environments which contain Linux.   These operating environments are where the different distributions of Linux differ, not the kernel itself.

As you plan and prepare, you need to think about items like:

- The administrative environment. Are you going to use NIS or NIS+? Are you going to copy the administrative files with rdist or something similar? What about Active Directory or LDAP?

- The automount / autofs function in Linux works differently to the automount functions is Solaris and IRIX. You may need to think about how you will handle these if you use NFS.

- Kickstart is a mechanism for configuring a system similar to Jumpstart in Solaris.

- Red Hat Network (RHN) is a for-cost service provided by Red Hat to keep your patches and so on up-to-date. You can have a local copy. There are a lot of possible configurations. You may chose to not use Red Hat, but you will need some mechanism to keep your patches synchronized.

- Red Hat Package Manager (RPM) is available on distributions other than Red Hat. RPM is similar to RHN in that it helps you keep your packages and versions up to date.

- Computer Security. Linux has IPchains which is a per machine firewall.

Linux Standard Base:

- If you take nothing else from this paper, take this.

- Linux Standard Base (LSB) is a subset of the Linux Operating Environment that is common to all LSB certified distributions (most everybody except Debian). **Even Solaris will be LSB compliant in Solaris 10!!!.**

- Applications should be certified as LSB. If they are, they will run on any LSB certified Linux distribution.

- Push your application vendors to certify on LSB.

- See "www.linuxbase.org"

The process with Linux Standard Base is that the application vendor can down load a test suite. He then tests his application and changes anything necessary to meet the tests. When it is all running, he contacts the LinuxBase people and has them test his application for compliance. If it all works, he gets the application certified and it will run on any LSB operating environment. There is a charge for the LinuxBase testing and certification, but it is pretty nominal.

LSB User-id Numbering:

LSB versions of Linux (which is most of them) have a new convention in which user-id accounts numbered 0-99 are system accounts. 100-499 are reserved for "dynamic allocation by systems administrators and post-install scripts". User accounts are to start at 500 and go to 65,536.

It may at first seem that 65,000 accounts are a lot, but consider that you may put all your office staff on Linux as well, maybe not for running technical applications, but perhaps for connectivity purposes. You may want to establish ranges for each office in your

enterprise and leave room for additional offices or mergers. Some systems are now allowing more than 65,536, but of course you have to use the lowest common denominator for all your systems.

There is not currently a similar convention for group-ids, however if you have to deal with this for user-ids now, you might as well do something about group-ids.

Premium Editions:

The best known premium editions are Red Hat's Advanced Server and Advanced Workstation. Basically these editions are released more slowly – once every eighteen months instead of every six months for standard Linux – and have more attention paid to quality assurance. These editions also try to stick to a more core set of programs and functions.

Advantages include:

- Since the release cycle is longer, there is less systems administrator time tied up in just keeping up with the next release.
- Useful kernel patches and other patches are pre-applied.
- More testing.
- Better support as the vendor support staff will know the core well, they don't have to know the entire universe of possible packages.

Disadvantages include:

- Many interesting packages may not be included. If your vendor develops on the traditional edition and you are on the premium edition, you may not be able to run some applications without installing packages and patches yourself.
- Lock into the premium edition vendor
- Additional cost.
- The patches don't go away. You will still have frequent operating system and ancillary program patches to apply even though the base version does not change.

You will find security patches still come frequently and you need a plan to handle these.

Training:

Since Linux is just Unix, you might think that training would be something you could get by without. But how much do your people really know about Unix? Is this a good time to get your systems administration and systems support personnel that training that has fallen out of the budget the last several years?

While each shop will vary, there is a general consensus that going to Linux saves money but increases support staff. There is a pretty tight market for people who know Unix generally, let alone people who are up to the latest on Linux. Especially people who

understand the Exploration world AND Linux. There is a pretty good chance that you will need to "grow your own".

You will need to have at least some people with programming and troubleshooting training in C. Many people suggest C++ or Java or even C#, but many of the basic programs used in Linux are C programs, including Linux itself, bash, tcsh, ksh, Xfree86, perl, tcl/tk, and FVWM. GNOME and KDE use a lot of C++ and Java.

Depending on how you configure your systems, you may have to provide training to people who didn't used to be involved with Unix support. If you move your office productivity tools to Linux, for instance, you might be looking at training the PC help desk.

Sadly, some IT people look at their career path in terms of vendors and applications they support rather than what their immediate employer wants to use. You may be looking at a lot of turnover in your office helpdesk staff and other support staff when you go to Linux, resulting in even more training requirements.

Certification:

Almost all current Linux certification programs are based on the "university" model. In this model, all the programs and data are installed locally on the system and the network is mainly for email and Internet access. There is little or no NFS, NIS, networked license managers, remote Oracle access, etc. as is the case in most Exploration shops. The expectation is that email and web tools are your primary applications.

For the typical Exploration shop, certification might be useful, but not the complete background a systems administrator needs to have.

General certificate programs include:

- LPI - Linux Professional Institute
- RHCE – Red Hat Certified Engineer

Vendors including IBM and HP/Compaq also have more vendor specific programs.

Testing:

The best way to prepare and plan for going to Linux is to get a few systems and test them both in how to set them up properly and with your applications. If you use NFS, you will need to learn to set it up properly.

You may need to download and build some Open Source programs. Even if you don't need to immediately, this would be good practice for you and might highlight issues (change control, naming conventions, development vs. production libraries and directories) that you might otherwise skip past.

Remember that most of the people programming your exploration applications are also somewhat new to Linux. They may build in dependencies that you are not expecting. They may not adequately anticipate a new Linux production environment. You want to find these things out in a pilot program, not when you roll the systems out to the users.

You should be prepared to test on several generations or releases of Linux. Your vendor may develop or certify on an older release of Linux or your distribution, but you will want to run on the latest version that you can to minimize catch-up once you get across.

Transition:

You will probably not be able to convert your entire staff to Linux over a weekend. You need to think of a transition plan. This plan should involve one or more pilots. You need to think about whether to start with high-end users or low end users (most Linux systems are very fast but they have a 32-bit, 2-gigabyte limit).

There will be a lot of pressure to start with high-end users. This mostly comes from the marketing arm of your applications vendors who are eager to show you how fast their new applications work on Linux. This may not be the best way to approach this. Since Linux systems are by-and-large cheaper than Unix systems (at least in terms of capital), it makes a lot of sense to replace your bottom-end, fully depreciated, older technology Unix systems first. These systems are not usually in the hands of your high power users.

By replacing from the bottom up, you :

- Replace fully depreciated hardware, or hardware coming off lease.

- Have fewer problems with the 32-bit, 2-gigabyte limitations, high performance network cards, high performance cutting edge graphics cards, etc.

- Work through any problems before they hit your high-end users. Low-end users may be in a better position to do something else while you fix their system.

- Develop a body of experience that will help you later as 64-bit systems mature and you start to deploy 64-bit systems for the high-end users when their turn comes around.

- Make a lot of friends with the low-end users because they get a really big jump in performance.

Conclusions:

- Going to Linux can be fairly simple to very complex depending on how you configure your systems and how much you want to bite off in the early days.

- The better and more thorough your planning and preparation, the easier it will be. Allow time for testing and a pilot project.

- Expect, and manage your management's expectations, that your benefit will come from better performance, better support, better access to new software, and better access and career path for recruiting support personnel, rather than from capital savings on the hardware and operating system.

- Don't forget training.

- The time to start is now. Linux is already the standard and you will be using it before you know it.