Bloor Research - North America

Bloor Research North America
34 Dartmouth Street
Yarmouth, ME. 04096

Phone (207) 228 4944
Fax (207) 846 6662
BloorResearchNA@CS.com

Linux Is Ready

Scalability, Reliability, Security, Flexibility, and Total Cost of Ownership Considerations

November, 2002

Table of Contents

Executive Summary

The Role of Linux

The purpose of this report is to consider whether the Linux operating environment is suitable for deployment in mission-critical enterprise computing environments.

The characteristics for measuring the "enterprise readiness" of an operating environment are well known. To deploy an operating environment in a mission critical situation most IS executives agree that Linux needs to be:

- *Scalable* (providing the enterprise with the ability to expand processing headroom as needed to meet enterprise processing requirements);
- *Available* (ensuring that most applications can run in a highly-available fashion with 99.95% availability if required);
- *Reliable* (ensuring that systems, applications, and databases do not crash);
- *Secure;*
- *Manageable*; and,
- *Flexible*.

Additionally, many enterprise IS executives consider the ability to consolidate servers as an enterprise-class characteristic.

The remainder of this report examines Linux from these perspectives, and contains our critical review of various vendor's Linux strategies and products.

Linux Scalability

Enterprises evaluate Linux scalability from two perspectives: vertical scalability and horizontal scalability.

1. Vertical scalability allows IS managers to run applications and databases that were designed to exploit multiple processors residing within one vertically-scaled system environment. For instance, some large, monolithic, run-the-business enterprise resource planning and human resource applications from

SAP, PeopleSoft, and Oracle were designed and optimized to exploit 8-way, 16-way, 32-way and beyond symmetrical multiprocessing (SMP) servers.

2. Horizontal scalability equates to distributed computing. Using this configuration/technique IS managers are able to harness and aggregate the processing power of multiple distributed systems to achieve scalability objectives.

Note that some IS designers and managers consider clustered systems that share the same database to be vertically scaled — but we do not. Nor do we consider blade servers (essentially distributed processors collocated in the same system enclosure) to be vertically scaled systems.

Bloor NA Linux Scalability Findings

With respect to Linux scalability, Bloor NA found that:

1. At present, ***Linux scales well to 6-way symmetrical multiprocessing (SMP) on Intel architecture — but does not scale well vertically beyond 6-way SMP processor implementations***. However, expect to find Linux 2.5 scaling functionality to be backported into Linux 2.4 kernel distributions (allowing for 8-way scaling in the very near term). And the next major revision of Linux (2.6) is about one year away — and at that time Linux should scale efficiently on up to 16-way SMP platforms.

2. ***Linux does scale extremely well in distributed computing configurations***. IBM, Sun, and Dell all report that they are having great success deploying Linux in tightly coupled cluster environments where Linux is used in mission critical computing environments for high-performance technical computing (HPTC), financial modeling, and life sciences computational tasks.

   Additionally, Bloor NA interviewed twelve Independent Software Vendors (ISVs) that build and deploy computing Grids (Grids are large, distributed resource sharing computing environments) — including AVAKI, DataSynapse, Entropia, Parabon, Platform Computing, United Devices, and others — and these grid ISVs are also reporting great success scaling Linux in distributed computing scenarios.

Linux Availability

How does one make a system highly available? Availability is measured in uptime (for instance, a highly-available system could be available 99.99% of the time; and even more highly-available system could be available 99.999% of the time; a fault-tolerant system would be available 100% of the time). To provide high levels of system availability systems are usually configured to “failover” to other systems or components in order to ensure that computing can take place while the failure is being addressed.

Bloor NA Linux Availability Findings

Early editions of Linux were designed to operate in single systems environments and did not have the functionality needed to take automated corrective action if a failure occurred. Accordingly, human interaction (such as a physical reboot) was required. But, as a result of contributions from the vendor community to the Linux kernel, as well as by contributions from the Linux open source community (that now numbers 400,000 developers), the Linux kernel now has the needed extensions to allow for automated fail-over.

State-of-the-art of Linux failover/availability functionality is best illustrated by today's Linux cluster deployments — particularly in the grid community. Linux has become a "preferred" operating environment for running dedicated custom applications in the commercial life sciences marketplace. In this market, life science companies perform complex, supercomputer-like calculations. Should a processing failure occur, distributed resource management (DRM) software that runs on Linux automatically restarts the failed application module on another available processor — and then takes corrective action to alert a systems manager of a failure or to programmatically correct a failure.

Failover extensions can be found in the base Linux kernel (downloadable for free over the Internet); from Linux suppliers such as Red Hat, SuSE, or other United Linux suppliers; from traditional hardware/software vendors such as Sun and IBM; from point product fail-over software makers, and from grid suppliers (in the form of DRM software).

Linux Reliability

How does one determine whether a system or subsystem is reliable? From a reliability perspective, Information System (IS) managers look at statistics such as "meantime between failure", or examine log files to determine the number of failed processes that have occurred or the number of system reboots that were initiated to restart a system or process. Systems that perform reliably have high meantime between failure occurrences (mainframes, for instance, have 60 years MTBF) and few failures recorded in log files.

Bloor NA Linux Reliability Findings

Bloor NA observes that reliability is dictated by systems hardware as well as the level of sophistication of the operating environment and related applications that run on that operating environment:

1. Bloor NA's research indicates that almost ***90% of server Linux is installed on Intel hardware platforms*** — and these *platforms* (the systems themselves) are generally known to be reliable. But note: ***hardware reliability is highly dependent on the equipment manufacturer*** — and there are other, more reliable platforms than Intel-based servers that run Linux (such as mainframe architectures) from which to choose.

2. ***The Linux operating environment has also been proven to be reliable — especially when used to run dedicated applications*** (for instance, Linux platforms operate extremely well when deployed as dedicated firewalls).

3. ***When Linux systems have failed, the failures have largely been caused by incompatible applications contending for the same systems resources; poorly written device drivers; or limitations in the operating environment (for instance, early revisions of Linux were not written to exploit multi-processor environments).***

Linux Security

Linux, like all operating environments, can be subject to security attack. And like all operating environments, the approach to protect the enterprise from mischief, malfeasance, and damage involves putting in place the proper policies and procedures (such as "don't run executables that are attached to email, especially from people you don't know) as well as the right technologies to reduce risk. Note that these email issues are largely associated with Windows and Outlook. However there are Linux viruses as well.

From a technology perspective, Linux closely resembles Unix and provides Unix-like security. Users require passwords and authorization rights to access Linux services and resources. Linux contains logging, monitoring, and audit capabilities that make it possible to trace system and user activity. And Linux provides secure shells, secure sockets, transport-level security and encryption capabilities to help prevent security breaches. And numerous vendors provide additional point products that operate on Linux — providing additional security functions. So, when IS managers think "Linux security", think should think "Unix security" — because both are pretty much the same.

Bloor NA Linux Security Findings

But, Bloor NA did find that ***Linux security and Unix security are hugely different in one respect: openness.*** Because Linux is based on open source code, developers can read and modify Linux source code to meet their needs (vendors usually close their Unix source code, thus making modification and reading of source difficult to achieve).

This "reading" aspect of Linux is important because users and developers have long suspected the existence of "trap doors" in some Unix implementations (and in Windows). There are two types of trap doors: accidental (that are security exposures), and purposeful (as in planted). Purposeful trap doors are alleged entryways into operating environments that only vendors (and potentially some governments) know about. Accidental trap doors are entryways (such as buffer overruns that open entrances into operating environments) that were not planned — but can create security exposure. Conspiracy theory aside, because Linux is open, a large community of developers can closely scrutinize Linux code for both types of trap doors — ensuring that such entryways are exposed and closed. (and such is not the case with closed operating environments).

Further, this "modification" aspect of Linux is important. Linux developers can build their own layers of security directly on the Linux kernel — and such proprietary extensions are extremely difficult to break. And, as long as this source code is not made available for license in the general marketplace, enterprise developers need not make their security enhancements known. The ability to modify source code (and keep those modifications secret) results in making Linux virtually unbreakable for some enterprises (and governments) that choose to invest in specialized security development. However, it should be noted that most business CIO's will want to avoid making source code modifications that would limit their support options.

Linux Manageability

From a manageability perspective, Linux manageability is similar to Unix manageability. Linux buyers can download various open source management tools and utilities (for data management, content management, and so on). Or, Linux buyers can purchase commercially available Linux point product solutions from companies like BMC, Heroix and Easilize. Or Linux buyers can purchase complete management suites such as IBM's Tivoli or Sun's Management Center or various products from Computer Associates (CA).

Bloor NA Linux Manageability Findings

Although many Unix-based manageability tools, utilities, and applications can be used to manage Linux environments, ***Bloor NA found that several existing Unix management products need to be quality assured (tested and certified) to ensure that they can manage Linux-based servers***.

Having said this, we found several very sophisticated Linux management tools (including workload balancing, performance/tuning, and other Linux management products) available from grid vendors (in the form of distributed resource management tools, utilities, and applications).

Further, we found rich Linux management frameworks/schema (integrated tools, utilities, and applications) available from some vendors (such as CA, IBM, and Sun). In general, the companies that are in the "systems management business" have been aggressive about updating their Unix products to run on Linux.

Linux Flexibility

For thirty years IS managers have been looking for an operating environment that would allow applications to be moved easily from one hardware platform to a distinctly different hardware platform to accommodate changing workload requirements. One of the original design points for Microsoft's Windows was to support multiple hardware platforms including Alpha-, MIPS-, and Intel-based servers — but now Microsoft's Windows server operating environment runs exclusively on Intel (or Intel-based) processors. And although Unix originally offered the prospect of moving applications across disparate systems platforms, the splintering of Unix (the creation of multiple, sometimes incompatible versions of Unix) prevented transparent application re-hosting on diverse hardware platforms.

The Linux operating environment has the same "flexibility" objective — to enable workload to be re-hosted on the appropriate hardware platform to accomplish the task-at-hand. And to date, Linux has been highly successful in achieving that objective — running on systems and devices ranging from small, mobile hardware chipsets such as ARM; to various embedded chipsets; through popular but somewhat obscure chipsets such as Saturn, Hitachi's H8, Amtel AVR, the Motorola 68K family; all the way through powerful, enterprise server chips such as HP's Alpha, Sun's UltraSparc, Intel's Pentium, and Itanium series, and IBM's PowerPC series.

Note that IS managers are not the only ones to benefit from the ability of Linux to run across multiple platforms — Linux platform flexibility also has benefits for Independent Software Vendors (ISVs):

ISVs write and sell software solutions. And because there are so many platforms and operating environment permutations and combinations from which to choose, ISVs generally standardize on certain brands of Unix, and on Windows — eschewing other Unix operating environments in order to reduce quality assurance testing costs. Linux offers ISVs the opportunity to write code once — and then run that code on many different platforms.

More precisely, Linux, and the consistency of the GNU toolchain across various hardware architectures provides ISVs with the ability to write C/C++ applications that can be engineered once an a Linux/Intel platform and then rebuilt and deployed on other hardware architectures without re-engineering.

ISVs can also write Java (J2EE — enterprise edition) code that can run on multiple different platforms (Unix, z/OS, Linux, et al). The trade-off between writing directly to Linux versus writing to a higher level of abstraction is performance (code written to Linux and compiled/optimized for the underlying hardware runs faster).

The bottom line is that ISVs can write to Linux and run their code on many different platforms (as long as they compile for those platforms) — providing ISVs with broader market exposure and less quality assurance. And ISVs can also write Java code that runs on Linux and other operating environments if they so choose.

Bloor NA Linux Flexibility Findings

Linux is flexible; it runs on numerous hardware implementations. ***To ascertain which microprocessors are supported; to locate drivers, compilers, assemblers, and other resources for deploying Linux across a wide variety of processor architectures, Bloor NA recommends that IS managers visit www.SourceForge.com or www.Linux.org*** — homes of the Linux development community. These sites contain valuable information on processor support, tools and utilities, assemblers, debuggers, and other ancillary software that can help IS managers deploy and manage Linux environments.

Also note that Linux vendors usually do a good job of quality assuring various devices, device drivers, systems, and applications that run on their Linux

implementations — and they list tested configurations on their respective sites. So one way to ensure that a system and its components will work with Linux is to check vendor sites for such information.

One Note of Caution

Bloor NA advises that IS managers keep a watchful eye on Linux vendors (such as Red Hat, SuSE, and United Linux vendors) to ensure that they continue to implement Linux in a consistent fashion. For example, pay close attention to whether these vendors implement non-standards-based "proprietary extensions" to their respective Linux libraries. Such extensions, if not implemented by all Linux vendors, cause implementation incompatibilities — potentially leading to the splintering of Linux.

To date, we at Bloor NA have seen few examples of such extensions — but diligence in this regard is called for lest we see the kind of fragmentation that occurred when Unix vendors decided to introduce proprietary implementations.

Linux Server Consolidation

Many IS executives know that it is expensive to manage and secure a distributed computing environment. These IS managers also know that if they are able to consolidate their computer systems into larger, vertically scaled systems, they can recognize several administrative, operational, and license-related benefits.

Some of these benefits include:

1. Test, production and backup servers can be virtualized on a single mainframe;
2. Centralized security and disaster recovery;
3. Lower application software licensing costs;
4. Easier software and server upgrades;
5. Less floor space and simpler cabling;
6. Improved utilization; and,
7. Savings in support and management personnel.

In order to consolidate servers, it is necessary to have two things:

1. An operating environment that can exploit SMP; and,
2. A vertically scalable systems platform (hardware) that allows IS managers to run the same workload that had been previously distributed amongst smaller (generally 2-way and 4-way servers).

Bloor NA Linux Server Consolidation Findings

As we described in the scalability section, Linux does not yet scale well vertically on Intel architecture beyond 6-way server environments (a situation that is expected to be rectified in the short term by back-porting Linux Rev. 2.5 scaling functionality into revision 2.4 of Linux; and in the longer term at year end 2003 with the next revision of Linux).

Despite Linux' inability to exploit large, vertically scaled Intel architecture, ***Bloor NA did find examples of scaled Linux installed on IBM's zSeries.*** IBM's virtualization technology provides the ability to host mixed workloads. Two types of virtualization – one at the microcode level (hypervisor / LPAR) and one at the OS layer (z/VM) – allow for server consolidation. The first type offers native speed but limits the number of OS instances that can be run. The second allows for "virtually unlimited" scaling in the number of OS images that can run concurrently.

The Bottom Line: Linux is Enterprise Ready

After examining Linux scalability, availability, reliability, security, manageability, flexibility, as well as server consolidation characteristics, ***Bloor Research believes that Linux is enterprise ready.***

Vendor Positioning

Linux is based on a licensing scheme called "GPL" — General Public License. This license allows users/vendors to copy and distribute copies of Linux using the GNU General Public License (and these users can make money reselling licenses with support if they so desire). But this license also states that users/developers that modify Linux and make the modified version generally and commercially available also need to publish any Linux code they modified (and make that source code generally available to the Linux community). In other words, vendors can add Linux "extensions", but they are not able to create proprietary value added extensions — instead, these extensions (if marketed commercially) need to be documented and provided back to the Linux community. (Note that when writing applications on top of Linux, the GPL does not apply; these can be closed source as well as open source not limiting the ISVs ability to make money on that application.)

Bloor NA found dozens of vendors competing in the Linux marketplace. Given this GPL condition, many Linux suppliers have struggled with the issue of how to make money on Linux (because its difficult to make money by adding value to the operating environment and its also difficult to make money when the operating system is available as a free download). Some vendors have tried to earn revenue by providing product support or integration services; others have tried to make money by selling Linux applications; still others have tried to make money selling hardware. Some vendors remarket Linux while making money selling support services. Other vendors partner with companies like Red Hat or SuSE — and remarket those companies' versions of Linux on Intel and non-Intel hardware platforms. Other vendors sell Linux point products. Still other vendors sell complete management environments, or complete infrastructure environments.

From a vendor perspective, Bloor NA advises that prospective Linux buyers would be well advised to consider their supplier's business case when choosing enterprise-class Linux operating environment/database/application providers. In order to ensure a quality and long-lived Linux experience, make sure that your Linux supplier's business model makes sense (and that your supplier's model enables it to make money selling Linux-related hardware, software, or services).

Preface

In 1999, Bloor Research took a very close look at the "enterprise readiness" of the Linux and Windows NT operating environments — and concluded:

> As a file and print server, Linux comes out on top, particularly for large organisations with various locations where remote management is an important option. The same goes for Web and mail servers where the uptime is crucial, although NT doesn't score badly in combination with Exchange either. In a database server environment, there is little or no difference between the two - it really depends on the characteristics of the database and the vendor's advice. The scale tips to Windows NT when it comes to application servers, because there is so much more software available for this platform, even if Linux is starting to catch up. And as for groupware servers : the application will determine the ultimate choice, but except for Lotus Notes, NT is the favourite here. Do we have mixed workloads (Web server, mail server, file and print server, etc.), then Bloor Research says : by all means, go for Linux. ***But not up to the enterprise level, because neither of the operating systems is ready for that task yet.***

Well, a lot has changed in both operating environments since we wrote the original "Is Linux Enterprise Ready " comparative white paper. So, it's time to revisit Linux to ascertain whether the Linux operating environment is enterprise-grade yet — to determine "is Linux *ready*?"

What Is This Report About?

In 1999 we examined Linux from nine perspectives: value for money (Total Cost of Ownership — TCO), user satisfaction, application support, interoperability, scalability, availability, support, operational features (flexibility), and functionality…

But since 1999, Linux has matured significantly. In this research report, Bloor Research – North America (Bloor NA) broadens our consideration of Linux. We consider many of the same issues that we examined in 1999 (such as Linux scalability; reliability/availability; security, and Total Cost of Ownership) — but we also take a look at:

- Linux flexibility (the ability to run Linux on differing hardware platforms);
- Linux as a platform for server consolidation;
- Middleware solutions that can be used to help build enterprise-class Linux infrastructure; and
- Linux market/competitive positioning.

Defining "Ready"

Many buyers of information systems have already been through this "enterprise ready" drill before. Over the past fifteen years we've seen Unix rise from the halls of academia to become an enterprise class operating environment able to conduct hundreds of thousands of transactions per second, capable of generating complex graphical models, and able to run enterprise mission-critical business systems. And we've seen Microsoft's Windows operating environment grow from a simple graphically oriented desktop environment, to a file and print server (circa 1996), to a powerful commercial server environment. Further, over the years, we've seen operating systems like IBM's z/OS (formerly MVS) and OS400 grow and mature to become archetypical models for "enterprise-grade" operating environments.

So, based on historical precedent, we already know what it takes to make an operating system "enterprise grade". It takes:

- *Scalability (increased processing headroom; server consolidation capability);*
- *Reliability;*
- *Availability;*
- *Security;*
- *Manageability*; and,
- *Flexibility.*

Linux – A Very Different Operating Environment

Although this "is-it-enterprise-ready?" drill may seem familiar, Linux is distinctly different than other enterprise operating environments in that:

- First, it's free (when acquired from the open source community). And when compared to the @$700/server fee for a basic Windows server operating environment — and multiplied by hundreds or thousands of servers, free is quite a bargain!

  However, it is important to note that free Linux comes with little (or more likely no) support. And that may be okay for hobbyists, home users, and some businesses — but many enterprises require some level of support to help answer questions and for troubleshooting. Accordingly, several Linux vendors make revenue today by reselling Linux along with support.

- Second, its open source. And because it is open source, a development community of over 400,000 people can participate in building and extending this operating system (no one vendor has such an army of developers). This community also safeguards Linux from splintering (the problem that occurred when vendors created their own proprietary extensions to Unix — and did not open their source code to the market).

- Third, it is based on a licensing scheme called "GPL" — General Public License. Linux source code is freely distributed and available to the general public. Users are permitted to copy and distribute copies of Linux using the GNU General Public License (and these users can make money reselling licenses with support if they so desire). But users/developers who modify and make their code generally available also need to ensure that the modified source is also made generally available to the Linux community. This approach is distinctly different than other commercial licensing approaches in the marketplace today (for more details visit: http://www.linux.org/info/gnu.html).

- Fourth, it's strategic. Companies such as IBM, Hewlett-Packard, Dell, and Sun have all announced formal support for the Linux operating environment. IBM has even gone so far as to promise to cascade its rich AIX (Unix) functionality to Linux over time.

- Fifth, it's flexible. Linux can run on devices ranging from ARM processors in mobile devices through powerful enterprise servers.

  Linux offers IS managers something they've been wanting for a long time: a flexible operating environment that allows applications to be deployed on a given hardware platform — and allows those applications to be easily redeployed on other hardware platforms as workload requirements change.

Other Considerations

We also know that enterprises consider other factors when making strategic operating environment decisions. They consider:

- *Investment protection* — Is the operating system flexible (can it run on multiple, disparate hardware platforms)?

- *Cost* — What does it cost to purchase? To configure/deploy? To manage? What is the total-cost-of-ownership (TCO)?

- *Application support* — Which applications can it run? How well does it run them?

- *Applications development/middleware/infrastructure* — What application development tools and utilities are available? What middleware is supported? How does one develop an integrated information infrastructure that operates on this operating environment?

- *Support* — Are vendors/application suppliers committed to supporting this operating environment? If so, how committed are they?

- *Manageability* — Tools, utilities, point products, schema, and frameworks that make it easy to deploy, monitor, control, and otherwise manage systems and distributed systems. And,

- *Standards* — Which standards does this operating environment embrace?

These "other considerations" are discussed in various chapters within this report.

How Is This Report Organized?

In this report we structure each chapter as follows:

- Concept (a review of the dynamics of the issue-at-hand — i.e. reliability, scalability, or security);
- Considerations (how do IS managers approach solving issues related to reliability, availability, scalability, etc.);
- Linux Situational Analysis (assessment of Linux situation vis a vis the issue-at-hand);
- Real-world examples (proof points — how early adopters are faring); and,
- Conclusions (the results of our analysis).

Learning Objectives

Upon completing this report, we would like you to understand:

- The decision-making criteria related to enterprise readiness (short-comings, issues, strengths);
- Why Linux flexibility is important;
- Cost of Acquisition and TCO implications of Linux;
- Server consolidation on Linux;
- Infrastructure considerations related to Linux; and,
- Vendor market/competitive positioning related to Linux.

Comments/Feedback

As with all Bloor Research – North America reports, your feedback is both welcome and appreciated. Please contact joe.clabby@bloor-research.com with comments and critique.

Chapter
1

Linux Reliability/Availability Considerations

Concepts: Reliability and Availability

The concepts of reliability and availability are distinctly different:

- *Reliability* has to do with whether the required resource (system/storage/application) is operational when needed to perform a computing task; and,

- *Availability* has to do with how quickly corrective action is taken should a resource not be available.

Reliability

How does one determine whether a system or subsystem is reliable? From a reliability perspective, Information System (IS) managers look at statistics such as "meantime between failure", or examine log files to determine the number of failed processes that have occurred or the number of system reboots that were initiated to restart a system or process. Systems that perform reliably have high meantime between failure occurrences (mainframes have MTBF of 60 years) and few failures recorded in log files. Also, 100% reliability equals totally available.

Availability

How does one make a system highly available? Availability is measured in uptime (for instance, a highly-available system could be available 99.99% of the time; and even more highly-available system could be available 99.999% of the time; a fault-tolerant system would be available 100% of the time). To provide high levels of system availability systems are usually configured to "failover" to other systems or components in order to ensure that computing can take place while the failure is being addressed.

The Reliability/Availability Bottom Line

Conceptually, reliability issues deal with the health of a systems platform and the stability of the operating and application environments. Availability issues deal with corrective actions to be taken should a failure occur in order to meet expected levels of performance.

Considerations

What factors cause reliability and availability issues? And how do systems manager ensure that systems and applications meet required service level expectations? Let's take a closer look…

Reliability and availability problems are most often caused by:

1. Failed hardware components;

2. Software device drivers;

3. Software incompatibilities between disparate programs; and,

4. Operating environments that are unable to properly manage resources (for instance: unable to manage multiple processors).

So, to adjudge Linux from a reliability/availability perspective, one must examine the hardware platforms on which Linux runs; the quality of software device drivers that Linux uses to address systems components such as graphics subsystems and storage; and the kernel itself (to ascertain whether it is solid or whether it requires consistent rebooting). Finally, if Linux is used in a multi-application environment, quality assurance prototyping and testing should be conducted to ensure that there are no application incompatibilities or resource utilization conflicts. (Note that some Linux vendors perform such testing for prospective buyers — thus eliminating the need for component integration and testing).

Linux Situational Analysis

Based upon user feedback, Linux (for the most part) is considered to be reliable and can be configured to be highly available — but this statement needs to be qualified…

Reliability

Bloor NA observes that reliability is dictated by systems hardware as well as the level of sophistication of the operating environment and related applications that run on that operating environment:

- *Platforms* — Bloor NA's research indicates that almost 90% of Linux is being installed on Intel hardware platforms — and these *platforms* (the systems themselves) are generally known to be reliable (but note: hardware reliability is highly dependent on the equipment manufacturer — and there are other, more reliable platforms from which to choose). Additionally, pricing is a consideration for choosing Intel versus other reliable platforms (commodity Intel hardware is generally lower priced than other hardware platforms). If pricing is not a driving factor, then there are other platform options from which to choose that provide greater meantime-between failure reliability characteristics.

- *Operating Environment* — The Linux operating environment has also been proven to be reliable — especially when used to run dedicated applications (for instance, Linux platforms operate extremely well when deployed as dedicated firewalls).

  In its raw state, Linux's major availability advantage against Windows NT is that it rarely suffers software failure. Windows NT does suffer this due to a number of factors, including memory bleeding and software incompatibilities due to rogue DLLs and other technical issues. Linux reliability will diminish with multiple applications, but we're not to the same degree as Windows NT.

- *Applications* — When Linux systems have failed, the failures have largely been caused by incompatible applications contending for the same systems resources, poorly written device drivers, or limitations in the operating environment (for instance, early revisions of Linux were not written to exploit multi-processor environments).

Availability

Early editions of Linux were designed to operate in single systems environments and did not have the sophisticated software needed to take automated corrective action should a failure occur. Accordingly, if failure occurred, human interaction (such as a physical reboot) was required. But, as a result of contributions from the vendor community to the Linux kernel, as well as by contributions from the Linux open source community (that now numbers 400,000 developers), the Linux kernel now has the needed extensions to allow for automated failover.

In fact, Linux high-availability has become extremely sophisticated in a very short time. State-of-the-art of Linux failover/availability functionality is best illustrated by today's Linux cluster deployments. Linux clusters have become a "preferred" operating environment for running dedicated custom applications in the commercial life sciences marketplace. In this market, life science companies perform complex, supercomputer-like calculations. Should a processing failure occur, distributed resource management (DRM) software that runs on Linux automatically restarts the failed application module on another available processor — and then takes corrective action to alert a systems manager of a failure or to programmatically correct a failure.

Note that with failover there can be a loss of processing time or a degradation of service (which may get counted as "unavailability"). Availability is complex to define. If measured by availability of applications to users, there are no examples of 100% available systems even from Tandem. 99.999% has been achieved on mainframe, and on Tandem. 99.95 is really very good. Unix doesn't normally get that good in practice (except for single workload boxes), but IBM i-series AS/400 does. The Visa processing system (amongst the most highly available) suffers about 5 minutes downtime in a year (on a mainframe). The ATM systems run by Tandem were similar.

Source: Bloor Research UK

Failover extensions can be found in the base Linux kernel (downloadable for free over the Internet); from Linux suppliers such as Red Hat or SuSE; from traditional hardware/software vendors such as Sun and IBM; and from grid suppliers (in the form of DRM software).

Real World Examples

Reliability

- *Turning Stone Casino Resort* located in Verona, New York launched a totally cashless gaming environment that electronically tracks and records results. Patrons use cards to manage their account and collect winnings. Based on a desire to improve selection and provide customized service patrons while improving the efficiency of the IT infrastructure, Turning Stone shifted to an open, Linux-based solution.

  Turning Stone replaced its backend gaming systems with an IBM solution, including H70 and H50 models and i-series 720 machines (the i-series is the new name for the AS/400 — it is important to note that Turning Stone is using the i-series platform instead of Intel-based servers — most likely due to the reliability characteristics of the i-series). The i-series servers provide a platform for Turning Stone's property management and financial applications. The casino's custom protocol translation application, which converts player terminal information coded in proprietary protocol to a universal gaming protocol is being run on 19 x-series 135 systems (Intel-based platforms). The IT department has found that the Linux solution provides open, remotely manageable, 24x7 operation. Beyond that, they have found in testing that should a failure occur, the system can be rebuilt in approximately 20 minutes.

  As a result of this implementation, Turning Stone can now offer more games, and by gathering customer information, a more tailored gaming experience.

- *Westport River Winery* moved from Microsoft to Linux to reduce the impact of accelerating licensing fees and to alleviate problems with performance and reliability. As a small business, controlling costs while maintaining customer service is the key to remaining competitive. Westport Rivers installed ACCPAC Advantage Series with IBM DB2 database software as well as Lotus Domino running on Linux and they expect to save the 2 to 3 hours per week typically spent troubleshooting issues with the proprietary system. Now with more time dedicated to managing the business, this winery is recognized as one of the top five sparkling wine producers in the country. And as the business continues to grow, the Linux-based system can scale to handle it.

- *Le Figaro,* one of France's leading newspapers since 1826 (with a daily circulation of almost 1.4 million readers) constantly faces 24 x 7 deadlines while trying to coordinate both editorial content and news garnered from reporters. Le Figaro installed Red Hat 4.2 in 1996 to create a stable, reliable company-wide system. Their system has grown to include

numerous servers running Red Hat Linux 6.2 and 7.0. Linux was an ideal solution because it could work in an environment with different types of servers. But the primary reason Linux was chosen was because Linux servers "don't crash" and they "don't require any support or maintenance", meeting the 24 x 7 demand of Le Figaro's business. These lower operating costs combined with the lack of Linux licensing fees has also lowered operating costs significantly. For more information go to http://www.redhat.com/whitepapers/

Availability

- *BBDO INTERACTIVE* a German Web design and hosting service provider (and a subsidiary of BBDO, one of the world's largest advertising agencies) selected a Linux and DB2 infrastructure to provide 99.99 percent Web hosting uptime for their customers. IBM was selected based on their broad product line and experience and commitment to Linux and high-availability solutions. Based on a desire to attract more high-end clients, BBDO wanted to improve their uptime guarantees — they selected IBM DB2 Universal Database Enterprise Edition for Linux on IBM x-series 330 servers running Red Hat Linux in a high-availability configuration. After implementing this solution and providing customers with a 99.99% uptime guarantee and improved customer service, BBDO also realized substantial cost savings: operating environment licensing fees 80% lower than comparable installations; TCO estimated at 75% less than comparable installations; and no payment of penalties for downtime.

Conclusions

Linux reliability is dependent on the hardware platform on which it runs; on the quality of driver and application software, and on the kernel itself. The Linux server market uses Intel-based systems 90% of the time — and these systems are known to be reliable (depending on system manufacturer as well as the components used within).

It is Bloor NA's perspective that Linux systems can operate extremely reliably provided that:

- **The systems platform is purchased from a reputable, known-for-quality vendor;**
- **Proper quality assurance if performed to test device drivers, application interaction, and system integrity before deployment. Note: if you buy from a vendor you should expect that vendor to qualify device drivers; if you download Linux on your own, you are responsible for qualifying your drivers.**

If these conditions are met, it is our opinion that Linux can operate *extremely reliably* in enterprise class computing environments.

With respect to systems availability, availability has to do with corrective action that should be taken should a system, network, or application failure occur (and this action can be taken by a human, or a software program that understands what policies, procedures, and technologies are to be used to facilitate recovery (i.e.- if a mission-critical system fails, it should have priority access to spare fail-over system nodes or components). This necessitates that humans consider the importance of a given application/system environment, and put in place the policies and procedures needed to ensure that a particular application system environment meets the required level of availability. Then, should a failure occur, a failed system or application can be restarted (either manually or programmatically) following the predetermined policy and procedures that dictate the corrective action to be taken (for instance — failing over to another system or component).

It is Bloor NA's perspective that Linux can operate in a highly-available fashion provided that:

- **IS managers take the time to put in place the proper policies and procedures required to meet availability expectations; and,**
- **IS managers deploy systems with the required fail-over components or redundant fail-over systems in order to meet availability expectations.**

Chapter
2

Linux Scalability Considerations

Concepts

Scalability has to do with the ability to expand processing power (headroom) as well as the ability to consolidate servers.

Scaling computer systems is usually accomplished in one of two ways:

1) Vertical scaling (expanding the number of processors within a physical system environment); and/or

2) Horizontal scaling (adding distributed, networked, systems or clusters to add incremental computing power. Grids also fall into this category).

An argument can be made that tightly coupled processors in a distributed cluster can create a virtual vertically scaled system. We do not consider these types of clusters to be vertically scaled because much work has to be done to distribute the application workload in order to achieve a virtual vertically scaled effect.

Further, blade servers house many servers within the same physical system environment — making blade servers appear to be vertically scaled. But in reality, blade servers consist of multiple CPU blades that are essentially distributed servers that are all contained in the same physical blade rack/housing. So, in reality, blades are horizontally scaled servers that share a common systems bus as well as power supply and other system resources.

Also related to scalability is the concept of server consolidation (the ability to replace/house multiple servers with one platform). Server consolidation is closely tied to the concept of vertical scaling — it allows IS managers to have centralized control and easier management of servers (because formerly distributed servers are consolidated onto one platform).

Considerations

What causes an IS manager to chose vertical scaling over horizontal scaling? Frankly, scalability is not a binary yes-or-no situation. Most IS manager use both approaches — and make their choice based on application requirements and behavior.

- In general, an IS manager chooses a vertical approach when an application is monolithic in nature (and is difficult to parse and run as a distributed application) or just plain runs better in a large symmetrical

multiprocessing environment (because it was written to exploit multiple processors contained in the same platform).

- In general, an IS manager chooses a distributed computing approach because the application can exploit multiple distributed systems and meet its scalability requirements; and because its is sometimes more cost efficient or affordable for an IS manager to add computing power in increments using distributed processors.

How is scalability measured? IS managers measure scalability on the basis of performance benchmarks including the Transaction Processing Council's TPC-C, TPC-H benchmarks; the EC-Perf (benchmark for Java); and the NotesBench on Domino. Linux does well in all of these benchmarks except for the TPC-C (OLTP) benchmarks because of Linux limitations in scaling beyond 6-way processing environments on Intel architecture.

One other consideration worth mentioning is that when Linux is able to scale to 8-way, 16-way, and beyond, it will start to challenge the processing power of Unix-based servers. In the mid-range and long-term (over the next two to five years), Bloor NA expects Linux to erode marketshare from Unix-based systems.

Linux Situational Analysis

Early versions of Linux did not scale well on vertically scaled symmetrical multiprocessing (SMP) architecture. In other words they were not able to exploit multiple processors. The current version of Linux (2.4) does well exploiting 6-way Intel server environments — but needs to improve on 8-way and above architecture. Currently, Linux 2.5 is under development and has functionality to support 8-way scalability. And the 2.6 version of Linux (due by year-end 2003) can be expected to address 16-way Intel server environments.

Having said this, it is important to observe that Linux does extremely well in geographically distributed and/or loosely coupled (horizontal) computing environments comprised of 2-way and 4-way Intel servers. In these environments, Linux can be found in tightly coupled high-performance computing clusters performing complex calculation, transaction, and modeling activities. And Linux can also be found in large, loosely coupled Grid environments (a grid is a tightly or loosely coupled network environment that allows for resource sharing) where groups of Linux processors tackle complex computing (or even supercomputing) tasks and/or are used for collaborative activities (such as collaborative modeling). Dozens or hundreds of Linux processors can be linked together in Grid configurations — potentially providing unprecedented levels of computational power.

Server Consolidation

The Linux server consolidation trend is also worth a closer look. Although Linux on Intel server consolidation doesn't make much sense today due to limitations scaling beyond six processors, Linux on IBM's z/OS may make sense for addressing certain server consolidation requirements. IBM's zSeries (mainframe) allows up to 15 Linux images (or Linux and other operating system images) to be

loaded and run simultaneously on a zSeries server. This consolidation is possible because of IBM's two types of virtualization technology — one at the microcode level (hypervisor / LPAR) and one at the OS layer (z/VM). The first type offers native speed but limits the number of OS instances that can be run. The second allows for "virtually unlimited" scaling in the number of OS images that can run concurrently. The zSeries and the Linux server consolidation topic are covered in greater depth in the next chapter (Chapter 3 — Linux Server Consolidation).

Real World Examples

- One example of high performance horizontal scaling is the Dell-Linux cluster at Pennsylvania State University's Center for Academic Computing. This cluster consists of 128 PowerEdge 1550 Servers, with a total of 256 processors running Linux Red Hat. Dell clusters offer high performance computing to researchers who demand supercomputer-like performance at an attractive price. In selecting a research platform, Linux Red Hat/Dell was selected based on scalability, price, reliability, footprint and the technological innovation typical of open source software.

- Boeing's R&D organization will replace mainframes with Linux compute clusters to run simulations required to design aircraft more quickly and cost-effectively. After evaluating several options, Boeing selected MSC.Software to provide Beowulf clusters (Beowulf is an approach to supercomputing that links PCs into a compute grid), engineering software designed for Computational Fluid Dynamics, MSC.Linux (a version of Linux optimized for compute-intensive applications), and HP clusters to deliver scalability and performance at a greatly reduced cost.

- Florida International University (FIU), a public research university located in Miami, conducts research on database management systems and various applications in its High-performance Database Research Center (HPDRC). One example of their work is the geographical data visualization and analysis solution called Terrafly. Developed in conjunction with the U.S. Geological Survey and major satellite data suppliers, Terrafly enables visitors to www.terrafly.com to "fly" over two-dimensional imagery and interactively manipulate data. These "flights" have a number of practical uses. For example, a prospective home buyer can scan neighborhoods, viewing homes and their respective selling prices. Would-be travelers can fly over resort areas, looking at surrounding areas and hotel locations, hyperlinking directly to the hotel's website. Terrafly serves the Miami and Washington DC areas today, but the goal of the project is to serve the entire US within a year and to go worldwide within five years.

  Based on the need for a high-performance, scalable infrastructure to support the high-resolution images and huge growth expected by Terrafly, the HPDRC selected IBM DB2 running on both IBM AIX and Linux. In addition to the scalability provided by Linux, the HPDRC saw additional Linux benefits including ease of software deployment and stability.

- *Wolfermans*, a gourmet baked goods company, uses IBM's Integrated Platform for Linux to scale their Internet business, which has quadrupled in size over the past four years. During peak times, Wolfermans wants to ensure that its 1 million online shoppers see no break in performance. Wolfermans, is using the IBM Linux cluster (which includes e-server x330 systems), eOneCommerce software from eOneGroup, WebSphere, and DB2 database software, as the basis of its internet business. With the new Linux cluster from IBM, Wolfermans has seen improved Web site performance that can easily scale to handle unpredictable peak demands.

Conclusions

This chapter provided example after example of Linux' ability to scale to meet customer requirements. But, a closer look at today's Linux scaling reveals that Linux customers are scaling horizontally using 2-way and 4-way Intel servers — not vertically. This is because Linux does not scale well vertically on Intel-based servers beyond 6-way server environments.

On the other hand, on non-Intel architecture (IBM's zSeries for instance), Linux is able to scale using virtual machine technology. IBM's z/VM supports 15 instances of Linux on the same server — thus providing an alternative to Intel-based platform architecture that can be used, amongst other things, for Linux server consolidation purposes.

Still, as Linux moves up scale (to 8-way and beyond platforms), Bloor NA expects that it will encroach on Unix territory — eroding low-end Unix initially, and then followed with a strong move into the Unix midrange as Linux security, manageability, reliability, availability, and other features become more sophisticated.

The next revision of the Linux kernel (2.6) is expected to allow vertical scaling to 16-way processors. Meanwhile, 8-way Linux scalability is being backported into Linux revision 2.4.

It should also be noted that Linux scales *extremely well* in tightly coupled, horizontal, high-performance cluster environments (performance clusters aimed at tackling complex numerical and graphical computing tasks). Grid vendors are agog at the horizontal scaling characteristics they have observed in Linux grid environments.

Chapter
3

Linux Server Consolidation

Throughout the report there are numerous customer examples where the use of Linux resulted in improved reliability, scalability, flexibility and manageability. But a closer look at a few of these customer examples shows that IS buyers are also using Linux to consolidate distributed servers — and recognizing lower Total Cost of Ownership accordingly. This chapter takes a closer look at how Linux is being used for server consolidation — and what results buyers are seeing by deploying consolidating Linux on IBM z-series enterprise servers.

Concepts

Many IS executives know that it is expensive to manage and secure a distributed computing environment. These IS managers also know that if they are able to consolidate their computer systems into larger, vertically-scaled systems, they can recognize several administrative, operational, and license-related benefits. Some of these benefits include:

- Test, production and backup servers can be virtualized on a single mainframe;
- Centralized security and disaster recovery;
- Lower application software licensing costs;
- Easier software and server upgrades;
- Less floor space and simpler cabling;
- Improved utilization; and,
- Savings in support and management personnel.

How Is Server Consolidation on Linux Accomplished?

In order to consolidate servers, it is necessary to have two things:

1. a vertically scalable systems *platform* that allows IS managers to run the same workload that had been previously distributed amongst smaller (generally 2-way and 4-way servers). And, as we described in Chapter 2 (Linux Scalability Considerations), the current revision of Linux on Intel-based servers does not yet have the headroom to scale well beyond a 6-way server. We also observed that 8-way through 16-way support is imminent (when the 2.6 release of Linux is made available in late 2003).

Note that IBM currently offers a 16-way Intel server (x-series) and both HP and Dell provide 8-way processors. Unisys offers a 32-way Intel platform (but it does not yet run Linux). It is reasonable to expect that when Linux 2.6 is released the hardware 8- and 16-way solutions needed to start consolidating Linux servers will be ready for immediate deployment.

3. An operating environment that is tuned to capitalizing on symmetric multiprocessing (SMP) — the ability to exploit multiple processors efficiently. Although there are other approaches to scaling (such as NUMA), traditional SMP is the way that scaling is best accomplished on Intel-based platforms — hence Linux SMP will be the way that Linux achieves vertical scalability. And because ninety percent of server Linux is installed on Intel-based processors and these servers only scale well to 6-way at present — the Linux server consolidation trend has not yet "taken-off".

Also described earlier in this report was a reference to IBM's zSeries enterprise servers and IBM's z/VM technology. With these technologies IBM can consolidate Linux-based file/print servers, Web servers, and application servers on a single platform. To accomplish this IBM provides a "virtual machine" technology called z/VM that simulates the existence of multiple processors, memory, I/O, and other resources to "guests". This technology gives Linux *logical* SMP headroom by simulating and controlling access by Linux to IBM's zSeries processor.

Because IBM can host multiple Linux instances on multiple virtual machines that sit on one platform (the zSeries), IBM is able to provide a vertically scalable Linux solution that can provide the centralized management, centralized security, improved utilization, and smaller footprint benefits usually associated with consolidated, centralized server environments.

Finally, it is worthy of note that Linux also runs on IBM's iSeries processors (formerly the AS/400). The i-series is extremely popular in small and mid-sized businesses where it runs mission critical applications as well as specialized and custom applications in a highly-reliable/available fashion. In most iSeries cases, Linux is run in an iSeries partition where Linux applications can gain access to DB2 databases and otherwise work cooperatively with iSeries applications from within a dedicated partition.

Considerations

Cost is a key element in determining whether server consolidation makes sense. And many factors influence cost — including:

- Number of professionals needed to manage a consolidated server environment;
- Training;
- Conversion/migration;

- Equipment/maintenance costs;
- Application and operating system software license fees;
- Software support ;
- Floor space/footprint reduction implications;
- Downtime costs;
- Systems utilization (consolidation may result in not needing to purchase or forestalling purchase of additional distributed servers); and so on.

Real World Example

- *Air New Zealand*, an airline industry leader in the Asia Pacific region, recently announced that they will replace roughly 150 Compaq servers with a single mainframe — IBM's zSeries — running Linux, IBM WebSphere Application Server, DB2 Database, and Tivoli software. As part of the shift to Linux on the mainframe, Air New Zealand will replace 4,000 Microsoft Exchange email and file and print clients with Bynari, an open source email application. Air New Zealand chose IBM's Linux platform based on application availability, flexibility, open source cost savings and long-term TCO gains from centralized ease of manageability.

- *Mobil Travel Guide*, well-known as a trip-planning necessity, will use SuSE Linux, IBM WebSphere and an IBM zSeries mainframe to support its expansion. With the help of IBM Global Services, this outsourced system is being set up as virtual Linux servers running operations from IBM's data centers as part of IBM's Linux Virtual Services program. This infrastructure of virtual servers allows ExxonMobil Travel Guide to support a rapidly growing business without making a large up-front investment. Based on a utility computing model, computing power will be available on an as-needed basis, allowing ExxonMobil to respond quickly to seasonal changes in demand. Over time, the company will move all of its web operations from Windows NT to the new services model.

Conclusions

Linux server consolidation may offer many benefits — particularly in the areas of reduced management costs, reduced OS licensing costs, and reduced floor space utilization. But Linux has not generally been used for consolidation to date because the OS has needed to mature somewhat in order to exploit greater than 4-way Intel processors (8-way and 16-way support is imminent).

To address Linux server consolidation limitations, IBM has found a way to run multiple instances of Linux on its zSeries enterprise servers. Using its z/VM technology, IBM is able to run hundreds or more instances of Linux on a single server — and let each share mainframe processing power. The Linux zSeries provides an intriguing proposition as a Linux consolidation server that can help reduce distributed computing costs through centralized management.

Chapter
4

Linux Security Considerations

Concepts

From a technology perspective, Linux closely resembles Unix and provides Unix-like security. Users require passwords and authorization rights to access Linux services and resources. Linux contains logging, monitoring, and audit capabilities that make it possible to trace system and user activity. And Linux provides secure shells, secure sockets, transport-level security and encryption capabilities to help prevent security breaches. And numerous vendors provide additional point products that operate on Linux — providing additional security functions. So, when IS managers think "Linux security", they should think "Unix security" — because both operating environments are pretty much the same with respect to security.

Prospective Linux buyers should be aware that Unix-to-Linux security ports are underway throughout the industry. Products that provide additional security to Linux are available from Tivoli, CA and others.

Considerations

Linux, like all operating environments, can be subject to security attack. And like all operating environments, the approach to protect the enterprise from mischief, malfeasance, and damage involves putting in place the proper policies and procedures (such as "don't open e-mail and download executables from people you don't know") as well as the right technologies to reduce risk.

Linux Situational Analysis

As stated previously, Linux and Unix are very much alike. But, Linux security and Unix security are hugely different in one respect: *openness*. Because Linux is based on open source code, developers can read and modify Linux source code to meet their needs (vendors usually close their Unix source code, thus making modification and reading of source difficult to achieve).

This "reading" aspect of Linux is important because users and developers have long suspected the existence of "trap doors" in some Unix implementations (and in Windows). There are two types of trap doors: accidental (that are security exposures), and purposeful (as in planted). Purposeful trap doors are alleged entryways into operating environments that only vendors (and potentially some governments) know about. Accidental trap doors are entryways (such as buffer overruns that open entrances into operating environments) that were not planned — but create security exposure. Conspiracy theory aside, because Linux is

open, a large community of developers can closely scrutinize Linux code for both types of trap doors — ensuring that such entryways are exposed and closed. (and such is not the case with closed operating environments).

Further, this "modification" aspect of Linux is important. Linux developers can build their own layers of security directly on the Linux kernel — and such proprietary extensions are extremely difficult to break. And, as long as this source code is not made available for license in the general marketplace, enterprise developers need not make their security enhancements known. The ability to modify source code (and keep those modifications secret) results in making Linux virtually unbreakable for some enterprises (and governments) that choose to invest in specialized security development.

Real World Examples

- *7-Eleven*, a worldwide leader in convenience retailing, is using IBM Linux to protect its internal e-mail infrastructure. 7-Eleven is running the Trustix Mail Server with AntiVirus scanning on an IBM eServer x-series environment running Linux to scan e-mail for viruses before distributing it internally to all 7-eleven staff and employees throughout the 7-Eleven computer network. 7-Eleven's Linux decision is based on the reliability of the Linux operating system as well as the security functionality it delivers.

- On June 3, 2002 the *German Minister of the Interior* and IBM signed a comprehensive cooperation contract that will enable the German administration to buy IBM hardware and software running Linux under competitive pricing conditions. Germany's decision to move to Linux and other open source software was part of a German security initiative based on increasing security concerns. The ability of Linux to provide a secure, flexible, cost-effective infrastructure made Linux a cornerstone of the new technology initiative. The German administration viewed Linux as a heterogeneous and more reliable alternative to Windows that would offer them more flexibility when selecting software. The solution will include IBM eServers hardware pre-installed with Linux distributed by the German SMB enterprise SuSE Linux AG.

In addition, a number of security agencies, including UK government security agencies chose Linux for security reasons.

Conclusions

Linux security is similar to that of Unix. The basic underlying technology consists of password protection linked to authorization rights, secure sockets, encryption, and so on. If additional security is required, many if not most security point products that run Unix can run on Linux.

IS buyers need to understand that Linux, like Windows and Unix, is also subject to the same types of virus, denial-of-service, and other hack approaches. So, IS managers would be well advised to put in place proper policies and procedures for preventing intrusion, deception, and damage. Further, IS managers would be

well advised to install security updates as they are released to patch potential security vulnerabilities.

What is most intriguing about Linux is its openness. The fact that developers can get access to source code enables them to read the source code and be assured that there are no “secret entrances” into their secured enterprise environments. Additionally, the fact that developers can add security extensions that they do not have to publish adds an extra layer of security that commercial, vendor supplied operating systems don’t provide. In these respects, Linux security is better than the security provided by Unix. However, it should be noted that most business CIO’s will want to avoid making source code modifications that would limit their support options.

Chapter
5

Linux Flexibility Considerations

Concept

For thirty years IS managers have been looking for an operating environment that would allow applications to be moved easily from one hardware platform to a distinctly different hardware platform to accommodate changing workload requirements. One of the original design points for Microsoft's Windows was to support multiple hardware platforms including Alpha-, MIPS-, and Intel-based servers — but now Microsoft's Windows server operating environment runs exclusively on Intel (or Intel-based) processors. And although Unix originally offered the prospect of moving applications across disparate systems platforms, the splintering of Unix (the creation of multiple, sometimes incompatible versions of Unix) prevented transparent application re-hosting on diverse hardware platforms.

The Linux operating environment has the same "flexibility" objective — to enable workload to be re-hosted on the appropriate hardware platform to accomplish the task-at-hand. And to date, Linux has been highly successful in achieving that objective — running on small, mobile hardware chipsets such as ARM; various embedded chipsets; popular but somewhat obscure chipsets such as Saturn, Hitachi's H8, Amtel AVR, the Motorola 68K family; all the way through powerful, enterprise server chips such as HP's Alpha, Sun's UltraSparc, Intel's Pentium, and Itanium series, and IBM's PowerPC series.

ISVs Benefit Too

Note that IS managers are not the only ones to benefit from the ability of Linux to run across multiple platforms — Linux platform flexibility also has benefits for Independent Software Vendors (ISVs). ISVs write and sell software solutions. And because there are so many platforms and operating environment permutations and combinations from which to choose, ISVs have had to standardize on certain brands of Unix, and on Windows, in order to reach the broadest range of customers while minimizing quality assurance costs. Linux offers ISVs the opportunity to write code once — and run that code on many different platforms.

More precisely, Linux, and the consistency of the GNU toolchain across various hardware architectures provides ISVs with the ability to write C/C++ applications that can be engineered once an a Linux/Intel platform and then rebuilt and deployed on other hardware architectures without re-engineering.

ISVs also have the opportunity to write Java applications that can run on Linux. These applications are written once (at a higher level of abstraction than at the operating system level) and can then run on numerous hardware platforms

without having to be compiled. The benefit here is that applications can run on multiple platforms without having to compile code for each specific platform (a portability versus performance trade-off).

The net result: ISVs that build on Linux can save time and quality assurance costs by creating one version of their code (as opposed to numerous specialized versions as is the case with Unix) that can run on multiple platforms.

> In terms of ISVs there is another factor that encourages the use of Linux as the reference platform/first port. This is the tendency for developers to work from home on their own machines. Their own machines tend to be high spec Intel PCs running Linux. In some ISVs developers have implemented Linux as the first platform without consulting management. We heard from a source at Platinum that, a few months before the CA acquisition, a memo was round the 31 developer centers that Platinum had, asking each for estimated time to convert their software products to run on Linux. 29 responded by saying that the port was already done. In fact, at many of those sites, Linux was already the reference platform.
>
> Source: Bloor Research UK

Considerations

To ascertain which microprocessors are supported; to locate drivers, compilers, assemblers, and other resources for deploying Linux across a wide variety of processor architectures, Bloor NA recommends that IS managers visit www.SourceForge.com or www.Linux.org — Web homes of the Linux development community. These sites contain valuable information on processor support, tools and utilities, assemblers, debuggers, and other ancillary software that can help IS managers deploy and manage Linux environments.

Linux Situational Analysis

Linux is flexible; it runs on numerous hardware implementations.

IS managers need to keep a watchful eye on Linux vendors (such as Red Hat, SuSE, and other United Linux suppliers) to ensure that they continue to implement Linux in a consistent fashion. For example, pay close attention to whether these vendors implement non-standards-based "proprietary extensions" to their respective Linux libraries. Such extensions, if not implemented by all Linux vendors, cause implementation incompatibilities — potentially leading to the splintering of Linux.

To date, we at Bloor NA have seen few examples of such extensions — but diligence in this regard is called for lest we see the kind of fragmentation that occurred when Unix vendors decided to introduce proprietary implementations.

Real World Example

1. *Israel Aircraft Industries,* Israel's largest company with 17 factories distributed throughout Israel, selected a Linux platform based on the

flexibility of Linux. Using Linux on 60 clustered HP LP1000r or NetServers, IAI has achieved the price/performance ratio they were seeking. But at the forefront of the Linux decision was the desire to avoid being “locked-in” by a particular vendor who owns and controls source code. Linux, because it is open source, can be modified and customized for a customer’s particular needs. This allowed IAI to write extensions that have been tailored to their individual requirements.

2. *Centrelink*, the Australian Government's premier service delivery agency, has signed a four-year partnership agreement with IBM to provide mainframe capacity and associated software and services. Under Centrelink's strategic sourcing framework, IBM provides information technology infrastructure and services, as well as assistance to establish and support a Linux Laboratory to reduce costs and provide flexibility for the agency's IT systems.

Conclusions

From our perspective, Linux has (to date) achieved the design goal of providing platform choice and flexibility (just go to SourceForge.com and you’ll find dozens of chipsets supported — and these are complemented by dozens of development tools and utilities).

We see no signs of the kind of splintering that caused the Unix platform flexibility effort to fall apart.

We, therefore, expect that the Linux kernel will remain flexible — and add even greater processor support — for the foreseeable future.

Chapter
6

Linux Total Cost of Ownership Considerations

Concepts

Different enterprises have different metrics for measuring Total Cost of Ownership (TCO). Further, TCO measurement preferences also vary on a country-by-country basis, based on considerations like tax write-downs, cost of skilled labor, and so forth.

Research analyst firms generally take one of two approaches to help establish those metrics:

1. a *quantitative* approach; or

2. a *qualitative* approach.

The quantitative approach generally looks at acquisition and operational costs to determine how and where cost savings can be found.

The quantitative approach usually involves polling user populations according to pre-established measurement criteria such as cost of goods, cost of management, cost of deployment/integration, etcetera. Included in this quantitative approach would be:

- Reduced systems management costs (administrative, operational, support, and training must be weighed);

- Hardware/hardware support/maintenance costs (including the cost for redundant components or systems if high-availability is required);

- Software costs (including infrastructure software/middleware, security extensions; as well as application/database licensing fees); and

- More…

The qualitative approach looks at the advantages of making a move to a new technology, such as: the Linux operating system is free for download (can't get much less expensive than that) — and it's reliable; it's scalable; it's manageable; and it can be properly secured (so you don't have to sacrifice value). This approach does not focus on measuring costs. (This report is a qualitative report).

The difficulty that we, Bloor NA, have with taking a quantitative approach to Linux is that most organizations tend to measure quantitative costs differently. Some organizations focus on cost of acquisition; others focus on reduced human administrative/management costs; others include education and training costs. Still others combine all three and more (while other enterprises focus on one or two aspects). Due to the constant debate on characteristics for measuring TCO, we at Bloor NA focus exclusively on qualitative reporting.

Having said this, we offer the following comparative chart such that IS buyers can compare typical Linux acquisition costs to Microsoft Windows alternatives. We also provide references to a few excellent quantitative reports (done by another analyst firm) on TCO for the Linux market space.

Comparing Linux/Microsoft Packaging/Costs

From a packaging perspective, Windows 2000 comes in three "flavors" — Windows 2000 Server, Windows 2000 Advanced Server, and Windows 2000 Datacenter Edition. The Server and Advanced Server editions may or may not have functional equivalents in the Linux world depending on which company or source you obtain your Linux from).

The key element to consider when purchasing Microsoft's Windows operating environment is the "client access license CAL" fee. There are three different versions of Windows — so buyers need to determine which version meets their computing requirements and then buy enough licenses to cover the user population they are looking to serve. In small enterprises with few users, the license costs can be close. But in medium and large enterprises, the cost difference can be quite substantial as illustrated in Table 2 (below).

The Linux operating system, on the other hand, is not "for sale". Documentation, feature value-add, and support is what's for sale when buying Linux. So Linux can be "free" if you download a free version with little to no documentation; it can also cost a few thousand dollars (if enterprise-level technical support is required). The following

Table 2 — Windows 2000 License Fees vs. Linux License Fees

Operating System Characteristics	Microsoft	RedHat	GNU Linux	Cost Difference
Minimalist			Free to $80 with document-ation.	
Server Edition for 5 users	$860 street (5-user CAL); $1,070 (10-	$799 to $1499 to $2499 depending on support requirements;		MS $860 RH Linux -799 Difference **$61** (low support) MS $10,700

for 100 Users for 500 users	user CAL) x 10 licenses = **$10,700** $1,600 (25-user CAL) x 20 licenses = **$32,000**	No user limits		RH Linux 1,499 Difference **$9201** (medium support) MS $32,000 RH Linux 2,499 Difference **$29501** (high support)
Advanced Server Edition for 1000 users	$3,400 (25-user CAL)			
DataCenter Edition	Not Comparable	Not Comparable	Not Comparable	

Source: Bloor NA – September, 2002

As Table 2 indicates, license fees alone for Windows versus Linux operating environments can differ tremendously — especially due to Microsoft's per seat licensing schema. But what is not apparent from Table 2 are the cost ramifications of adding mail and messaging, or application development tools/utilities, or infrastructure. The next section explores the cost ramifications for adding infrastructure components.

Considerations

To gain a good understanding of Linux from a quantitative perspective, Bloor NA recommends that prospective Linux users examine two RedHat-sponsored IDC (a research firm known for quantitative work) reports: *The Role of Linux in Reducing the Cost of Enterprise Computing*; and *Linux: A Journey Into the Enterprise*. These white papers:

1. illustrate the cost savings that can be realized using Linux;
2. provide statistics based on end-user research; and
3. provide a methodology on how to build a TCO study.

Here's a brief summary of these whitepapers. The full reports can be found at http://www.redhat.com/whitepapers/.

Qualitative Observations from Bloor NA

We do have the following observations to add to the RedHat White Papers:

- The Linux/Intel platform (especially Linux clusters) offers excellent price/performance because of the low acquisition costs of Intel server hardware. Customers are attracted to the Linux promise of "Intel pricing with Unix reliability". Beyond that, Linux includes or supports many systems services and applications that would typically be purchased separately with a Windows or Unix solution. And because of the flexibility of Linux, many customers will reuse existing hardware as they make the shift to Linux.

- In the software maintenance phase, Linux benefits from loose licensing arrangements. Many Linux customers purchase a single copy of support to run on a Linux reference platform, thereby saving the cost of purchasing support licenses for each individual server. Linux resellers are currently searching for ways to overcome this support provisioning dilemma.

- Linux can run on IBM enterprise servers (mainframes) — making it possible for customers use Linux as the basis for server consolidation initiatives. Server consolidation affords these customers the opportunity to realize cost savings from centralized support and management.

- The reliability of Linux also results in cost savings. In the case of Westport River Winery (described in earlier in this report), shifting from a Windows to a Linux platform saved 2-3 hours per week in reboot time.

- Talented Linux systems managers often cost less than their Unix counterparts (because of a growing pool of university trained Linux coeds who are just now entering the job market). We grant that these coeds are not likely to understand complex, run-the-business enterprise application integration issues, nor high-availability and grid cluster configurations — but we are definitely of the opinion that for simple deployments and straightforward, single/dedicated application environments, the new pool of Linux talent is worth evaluating.

Anecdotal reports suggest that Linux support is better than ANY other OS support, because problems are addressed faster (by the open source community). There are also legendary stories of someone asking for the existence of a driver and someone else just writing it for them. That kind of support doesn't exist on any other OS.

Source: Bloor Research UK

Real World Examples

- Based on a desire to cut costs, *Amazon.com* made the switch from a RISC/UNIX platform to a Linux platform. Amazon chose Red Hat Linux as

a cost-effective alternative to UNIX that would offer the flexibility of an open source platform, ease and speed of implementation, and scalability and reliability across hundreds of servers. Aside from the savings in operating system licensing costs, Amazon discovered that shifting from UNIX servers to Linux running on Intel servers also yielded a substantial reduction in hardware costs, as well as giving Amazon the flexibility to buy from multiple vendors. Amazon found that they were replacing $60K servers with servers costing as little as $10K. Because of the TCO benefits of Linux, Amazon reported that their technology and telecommunications costs were down $17 million or 24% from 2000 to 2001. For more information, go to http://www.redhat.com/whitepapers/

- The *Jet Propulsion Laboratory (JPL)*, managed by the California Institute of Technology, is NASA's lead center for robotic exploration of the solar system. JPL is using a preassembled Linux cluster solution, consisting of 66 IBM x-series systems to perform analysis of mission data transmitted by the MLS probe on the Aura Spacecraft. The new IBM solution provides JPL with fully supported packaged system of hardware, software and services that will lower total cost of ownership.

- *Mississippi State University* has installed an 1,038 processor IBM Linux cluster made up of x-series systems at their Engineering Research Center. The research performed on the clustered system helps the Navy analyze ship designs to create quieter and more efficient submarines through computational fluid dynamics. The cluster also provides the massive compute power needed for remote sensing, computational physics, and automotive research. The decision to use IBM Linux clusters is based on price performance— Linux clusters provide supercomputer levels of performance to power the University's compute hungry simulations, without incurring the expense of a supercomputer.

Conclusions

Research analyst firms generally take one of two approaches to evaluate Total Cost of ownership:

1. a *quantitative* approach; or

2. a *qualitative* approach.

The quantitative approach generally looks at acquisition and operational costs to determine how and where cost savings can be found — and typically considers:

1) Costs to acquire (licensing fees of OS, systems platform, etc.);

2) Costs to manage (including support, training & implementation services and staff management costs); and

3) Costs of software (including infrastructure, applications, and databases).

The qualitative approach generally looks at the advantages and dynamics and benefits of moving to a new technology. In this report, Bloor NA examines qualities such as scalability, reliability, availability, manageability, and more. We don not focus on measuring costs. (Hence, this report is a qualitative report).

We do note that there are several good Linux TCO studies on the market today. The two that we like best have been done by IDC (and sponsored by Red Hat). They are: *The Role of Linux in Reducing the Cost of Enterprise Computing*; and *Linux: A Journey Into the Enterprise*. And the full reports can be found at http://www.redhat.com/whitepapers/.

Chapter
7

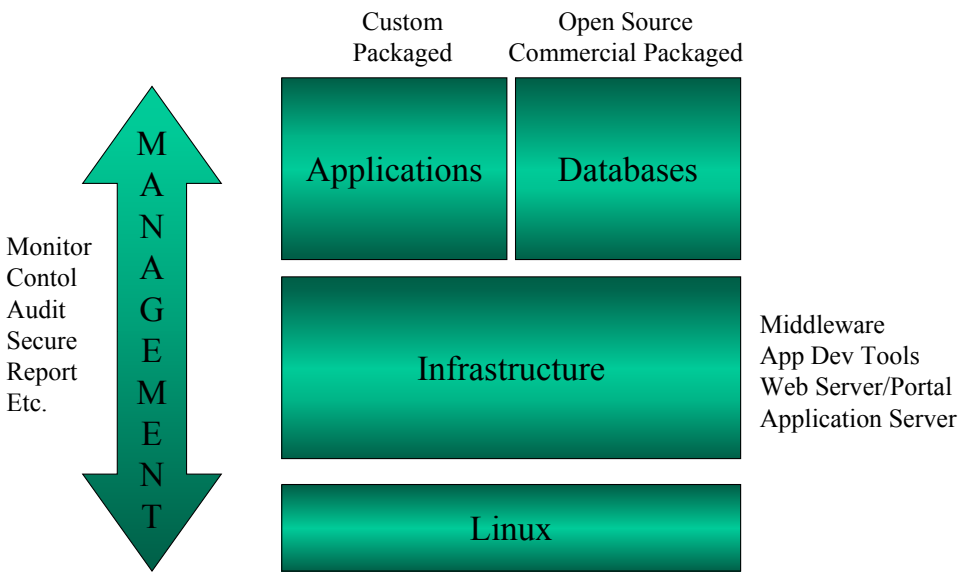
Linux Infrastructure Considerations

To this juncture, this report has focused on Linux attributes such as reliability, scalability, and security. And we've also provided some guidance regarding Linux TCO. But, it's now time to switch gears and look at middleware and infrastructure software that runs above the operating environment and makes it possible to run applications, databases, mail/messaging applications, and Web portals and services.

Concepts

As an operating system, the primary role of Linux is to control system resources (such as access to networks, file and print subsystems, storage subsystems, etcetera). But to run Linux as an enterprise server, IS managers require far more than just an operating environment — they require tools and utilities for building and deploying applications; they require middleware to facilitate program-to-program communications; they require systems management facilities to monitor and control applications and databases that run on top of the operating environment; and more (see Figure 1) …

Figure 1 — Linux Infrastructure



Source: Bloor NA — September, 2002

Considerations

Many of the application and Web development tools and utilities, much of the middleware, and most of the management tools used to build Linux infrastructure — as well as mail/message and database environments — are readily available from two sources:

1) the open source community; and

2) the vendor community.

IS managers, based upon the type of applications they are seeking to deploy, need to decide which approach best suits their needs. The dynamics of choice for building Linux infrastructure are as follows:

- IS managers generally use the open source approach if:

    - the Linux server is dedicated to serving only one purpose (for instance, acting as a firewall or running a confined application); and/or

    - the enterprise is seeking to build a unique, custom solution that requires or allows developers to exploit the underlying operating system for competitive gain.

    (Note: choosing this approach to building Linux infrastructure implies developmental commitment to integrating infrastructure with the underlying operating environment and testing various infrastructure, application, and database modules for operational integrity. To build such environments, open source tools, utilities, databases, middleware, etcetera can be acquired and downloaded over the Internet. (The reason that this approach implies developmental/IS administration commitment is that little, if any, support for code acquired for free via open source Linux downloads can be found).

- IS managers generally chose to use vendor supplied infrastructure products if:

    - the enterprise desires to run multiple applications on a Linux server;

    - the enterprise does not wish to spend developmental time and effort integrating and supporting infrastructure elements on Linux; and/or

    - the enterprise already runs vendor supplied infrastructure elements on other enterprise systems (and is seeking consistency in its information infrastructure).

    (Note: using this approach to building Linux infrastructure usually implies that an enterprises does not wish to commit developmental resources to

infrastructure design, deployment, testing, and integration — hence, the enterprise is relying on its Linux vendor to provide infrastructure integration).

Linux Situational Analysis

The considerations for choosing open source Linux infrastructure and commercially packaged infrastructure are pretty straightforward.

The open source approach to building Linux infrastructure should be taken by enterprises that have simple, dedicated applications; developmental resources for application development, testing, and support; and a willingness to perform application/database integration. Possible drivers for this course of action may be cost (a complete solution could be architected using free, open source software); competitive advantage (the enterprise wishes to build custom applications); and/or security (an enterprise could build a custom, secure environment).

The vendor-integrated, commercial packaged infrastructure approach should be used if:

1. the enterprise does not wish to spend its developmental time building and testing infrastructure solutions;

2. the enterprise needs to run multiple or cooperative applications on its Linux server(s); and/or

3. consistency and compatibility with existing systems within an existing information infrastructure. Tools, utilities, and integrated software (including application development environments, middleware, database, mail-and-messaging/groupware, Web portal/services, and other infrastructure products and services) are available from a number of vendors including Red Hat, SuSE, Sun, IBM, Hewlett-Packard, Dell, and various grid vendors). The following chapter ("Chapter 7 — *Vendors: Market/Competitive Positioning*") takes a close look at the product offerings of these vendors.

Real World Examples

- New York City's *Clarity Payment Solutions, Inc. (Clarity*) offers payment processing products to meet the needs of the growing market for prepaid payment options such as debit cards and stored value cards. Clarity targets companies in a variety of vertical markets including incentives, consumer promotions, payment solutions, payroll, healthcare reimbursement and insurance. Clarity evaluated Microsoft SQL Server on Windows, Oracle on Sun Solaris and IBM DB2 on Intel servers running Red Hat Linux. Linux infrastructure with DB2 was chosen based on enterprise reliability features not found in Windows, and better price/performance than the Oracle/Sun solution ($250K for Linux DB2 and $1.5M for Oracle/Sun). Clarity has found that DB2 running on Linux infrastructure provides the performance, scalability, and uptime (over 18 months without a reboot) required to support the rapid growth of the

electronic payment processing market — and the resulting need for Clarity to support millions of transactions per day.

Linux has also provided a platform for Clarity's in-house payment processing application that many customers choose to access as a Web service.

- *Toyota Motor Sales (TMS) USA,* has over 10,000 employees in over 1300 locations nationwide. Toyota selected Linux and Dell PowerApp servers as the best way to offer information-rich content to its dealers over a wide-area network with limited bandwidth. In order to provide this capability, Toyota needed to make modifications to source code. The configuration of the Red Hat Linux package was done with Apache. Toyota made modifications to the Apache code; added SNMP utilities, DNS, NTP and secure shell into the server; and optimized network settings. With Linux, the process was quick and relatively simple when compared to what would have been a much more complex process with packaged software. Another key benefit for Toyota is that remote access and management are part of the Linux operating system. Remote diagnostics, not available on other platforms, monitor and update field installations. And Red Hat also included many tools and utilities available from the open source community, saving in software licensing costs. For more information go to http://www.redhat.com/whitepapers/

- San Diego, California-based *Structural Bioinformatics, Inc. (SBI*), a leader in computational proteonomics and the use of protein structure information, performs computations on protein gene sequences to target specific proteins that will be used by pharmaceutical companies to develop drugs for the treatment of illnesses such as cancer and cardiovascular disease. Looking for a lower cost, higher performance, more reliable infrastructure, SBI embarked on a migration from Oracle and Sun to an IBM/Linux/Intel solution. SBI chose DB2 for Linux and Linux clusters of IBM x-series servers along with IBM WebSphere Application Server for delivering transactions over the Web.

  SBI discovered that their choice of a Linux, DB2 and x-series servers improved their cost efficiency in running the highly compute intensive protein modeling application. With the new Linux-based infrastructure, costs were reduced from roughly $28 per calculation to less than $1and more modeling computations were performed in less time, speeding the drug discovery process.

Summary Observations

The big question to be asked about Linux infrastructure is: "how should my enterprise build the infrastructure to support the applications and databases that run on top of the Linux operating environment"? And the answer on how to build infrastructure is dependent on the types of applications that your enterprise intends to run:

- If your enterprise's intent is to develop custom applications, or if your intent is to run a single dedicated application on your Linux server — and if you're willing to provide developmental and support expertise needed to deploy Linux and integrate its various infrastructure components — then use open source infrastructure.

- If your enterprise's desire is to run a multi-application server that can work cooperatively with other distributed servers, and your enterprise is not willing to devote resources to integrating infrastructure, applications and database — then go to packaged Linux infrastructure solutions.

Chapter
8

Vendors: Market/Competitive Positioning

Market Positioning

As described previously, the Linux kernel is free (unless you desire packaged software and documentation; or enterprise extensions such as clustering, security, and manageability; or various levels of support such as response within hours or days). Based upon purchase criteria, Linux could cost between $49 (for the basic personal version) and $2,500 or more (for the extended advanced server operating environment plus strong support). In fact, the primary income for several Linux vendors is making money by selling Linux along with support.

The sources of supply for this Linux kernel are:

1) The open source community (free downloads over the Internet). Extensions (such as security, management, and administration) that augment the basic Linux operating environment can be found within the open source community (examine www.Linux.org or www.SourceForge.com for a myriad of downloadable Linux extensions, tools, and utilities). Further, various 3rd party vendors build point product solutions that can be used to augment open source Linux; and,

2) Linux vendors (such as Red Hat, SuSE, et al — an extensive list of Linux source + extensions suppliers can be found at http://www.linux.org/vendors/retailers.html). These Linux vendors essentially package the kernel — and make their money by providing Linux personal, professional, and server packages that include extensions in areas such as clustering, security, systems management, installation wizards, graphical user interfaces to Linux file structures, even extensions for functions such as high availability . Many of these vendors also generate revenue by providing Linux design, deployment, and management professional services.

The Bigger Picture

As stated in the previous chapter, *enterprise deployments* of Linux require security, manageability, middleware, scalability, and reliability extensions. From Bloor NA's perspective, IS buyers have four sources of supply from which to acquire such extensions:

1. The open source community;

2. Point product software makers (that provide application development or systems management tools, for instance);

3. Infrastructure/application integration software providers (such as Oracle, BEA, Computer Associates, IBM, and others); or

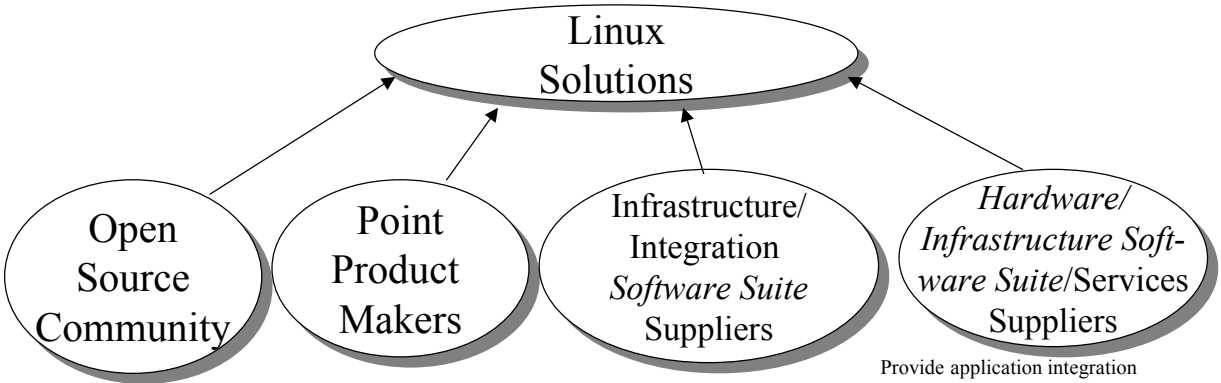
4. Systems/software makers like Sun, IBM, Dell, and Hewlett-Packard.

This chapter examines open source and vendor software products that can be used to help build a sound, enterprise-class Unix infrastructure.

Understanding the Competitive Dynamics: Linux Market Positioning

How are the abovementioned Linux open source, operating system companies, software makers, and hardware/software companies positioned? The following graphic (Figure 2) illustrates how the open source community, Linux operating systems makers, point product software makers, infrastructure/integration software makers, and system hardware/software/services providers position from a market and competitive perspective to service the Linux marketplace.

Figure 2 — Open Source/Vendor Market/Competitive Positioning



Provide open source operating environment; plus infrastructure/integration tools & utilities; development environment; database;groupware; etc.

Competitive Positioning: Opens source community thinks "Microsoft" Reality: Infrastructure providers are real competitors.

Other Comment: When competing with packaged suites, placing other open source code on open source Linux can create additional integration challenges.

Provide point product solutions for system management, security, configuration, administration, etc.

Competitive Positioning Other point product solution makers; infrastructure solution providers that sell software from suites ala carte.

Provide application integration tools, utilities, connectors as well as infrastructure solution suites that provide manageability, security, messaging, and the like. These run on multiple hardware platforms.

Competitive Positioning Other infrastructure solution providers that sell hardware/software and services; system management and database companies.

Competitors Appl. Integration Companies such as BEA, webMethods, IONA, Tibco … Grid makers including Avaki,, Platform Computing et al. Database companies likeOracle. Systems management firms like CA. H/W, S/W, ser-Vices suppliers whose software runs on other platforms such as IBM, HP.

Provide application integration tools, utilities, connectors as well as infrastructure solution suites. Suites contain web server, application server, development tools, mail & messaging, etc. These companies want to sell hardware, software, and pro services.

Competitive Positioning Other infrastructure solution providers that sell hardware/software and services.

Competitors IBM (with Tivoli, Lotus, DB2, WebSphere—across multiple platforms including Linux on Intel and Power PC). Sun (with Sun ONE, Sun Management Center, Sun Grid Engine, etc. on Linux on Sparc and Intel hardware). Others…

Source: Bloor NA- September, 2002

Understanding Linux Market Dynamics

This section examines the competitive positions of Linux infrastructure product suppliers.

The Linux Open Source Community

The Linux open source community consists of over 400,000 programmers and contributors dedicated to making Linux a robust operating environment — and devoted to providing Linux infrastructure solutions as either freeware or shareware. This community is extremely proud of the fact that Linux has been recognized by the analyst community as the world's fastest growing (in terms of marketshare) operating environment. This community feels as if it "owns" Linux.

In many respects, the rapid growth of Linux can be seen as a reaction to Microsoft's dominant role in the operating system, system software (infrastructure, tools, utilities, management), office productivity, and business applications marketplaces — and the company's current pricing strategies. Microsoft's unchallenged position in some markets has enabled the company to raise prices in areas where there are no real competitive threats (like the company's office suite); to raise prices on entry-level low-end servers (again, because other vendors are not positioned to successfully challenge Microsoft's Windows on Intel monopoly). And frankly, Microsoft's "my-way-or-the-highway" attitude has offended or concerned a great number of IS managers.

As a result of these concerns, the open source community has created its own database management solutions including MySQL; its own application servers such as JBOSS and TomCat;; its own mail/messaging solutions such as SquirrelMail; its own database management solution such as phpMyAdmin; its own groupware solution with TUTOS; and other alternative-to-Microsoft solutions.

Point Product Solution Providers

Point product solution makers provide turnkey, specialized solutions to address Linux issues such as administration, configuration, monitoring, security, content management, multimedia production, and so on. The benefit of using solutions provided by point product software makers is that such solutions are generally better tested, more easily installable, and better documented than free versions of the Linux operating environment, tools, utilities and applications.

Examples of Linux 3rd party point products include Heroix and Easilize. These companies make Linux management software. They compete with each other as providers of management solutions — and they also compete with larger management software makers such as Computer Associates, IBM, Hewlett-Packard and others (although the larger companies tend to provide complete encompassing Linux management solutions as opposed a la carte point solutions

Infrastructure/Integration Software Suppliers

There are numerous software companies that provide infrastructure solutions that run on top of the Linux operating system. Some of these companies approach

infrastructure from application integration origins (BEA, webMethods, et al); others approach Linux infrastructure design and deployment from a resource management perspective (AVAKI, DataSynapse, Platform, et al); some approach Linux infrastructure from both perspectives (HP, IBM, Sun, etcetera).

Full-suite Linux infrastructure *software* solutions suppliers such as Computer Associates, BEA, webMethods, Oracle, and the dozens of Linux grid software suppliers clearly see each other as competitors. But these software suppliers also compete directly with providers full, integrated hardware/software solutions (hardware, operating system, middleware, mail/messaging/groupware, databases, etc.) such as IBM, Sun, and Hewlett Packard.

Hardware/Software/Services Suppliers

Some vendors provide Linux infrastructure solutions that are integrated on their respective hardware platforms (Sun, IBM, HP, and Dell do this). The following is a high-level overview of how these companies are positioned in the Linux marketplace:

Dell

Dell's strategy is very clear: Dell is seeking to supplant Unix solutions from vendors such as IBM, Sun and HP with more cost-effective Dell Intel-based Linux solutions. Dell is targeting the enterprise market with Linux Red Hat/Oracle 9i as an alternative to UNIX. Dell's HPC (High Performance Computing) cluster solutions are based on Linux and target the high performance market segment where compute "grids" are popular – life sciences, financial services, manufacturing and scientific and research. And Dell is clearly pursuing the volume market for web servers, appliances and file and print servers. In addition, Dell is also offering Linux-based management products to augment the TCO benefits of Linux as a way to differentiate their Linux offering. Dell also offers a full set of professional services including support, implementation, UNIX migration and application porting. Dell wins points for its TCO story, horizontal scalability and manageability. But Dell does not provide the flexibility that is available from companies like IBM or HP who offer Linux on multiple platforms (Dell is a Linux on Intel-only shop).

Sun

Sun is in a very interesting position with respect to Linux. The company has a huge Unix base that it must protect — yet many of Sun's customers and dealer/distributors recognize Linux as a viable Unix alternative…

With respect to Linux, Sun offers Sun management and Sun middleware products (such as Sun Grid Engine and Sun ONE) products on Linux. In addition, Sun offers Linux professional services (particularly focused on building Java solutions on Unix and Linux). Sun also recently announced the LX-50, an Intel-based server that supports Linux. Also announced was an initiative called SunLAMP (Linux, Apache, MySQL and PHP), designed to give LX-50 customers an integrated hardware/software bundle. The Sun ONE line (application, portal

and messaging servers) will also be ported to the LX50. The LX-50 is targeted to Web caching, firewalls, and streaming media.

Sun's market approach is to capture "edge" computing applications (as an alternative to Windows), and applications in the high-performance technical computing (HPTC) segment. With respect to Windows, Sun's goal is to replace Windows/Intel-based servers and applications with Sun Linux solutions in traditional Windows markets (including office). In the HPTC segment, Sun is particularly aggressive selling its Grid Engine solutions (grid computing clusters) — and sells Unix grids, Linux grids, and combined grids.

Sun will continue to position Solaris as the high-end solution by emphasizing UNIX's strength in high availability, file systems, and partitioning.

HP

The new HP (the merged HP/Compaq) has become a "partnering company". The company partners with providers of the Linux operating systems and solutions, including partnerships with Oracle, BEA, SAP, Red Hat, SuSE, SendMail, Mitel, TurboLinux and SteelEye to name a few…

From a platform perspective, HP has a three-prong strategy operating environment/platform strategy that includes UNIX, Windows and Linux solutions. HP supports Linux (Red Hat as well as SuSE Linux is available on all servers with and United Linux to be available later this year) across its entire range of Intel-based servers including HP Intel-based Proliant servers, its NetServers, two blade architectures, high-availability clusters, and its workstations. HP's Intel line includes 8-way Intel SMP, as well as two Itanium-based servers).

HP target markets include telco, Internet infrastructure, server appliances, application development, enterprise application servers, database servers, ERP, technical computing (including visualization) and HPTC (high-performance technical computing).

Is HP a Linux Integrated Infrastructure Provider — Or Not?

Bloor NA believes that HP has a lot of work to do in the articulation of its Linux middleware solutions provisioning plan. On September 15th, 2002, HP announced its "HP Discontinuance Plan" relevant to HP middleware product offerings. The company announced the discontinuance of HP Application Server (HP-AS), HP Application Server Resilient Edition, HP Web Services Platform, HP Core Services Framework, HP Total-e-Server, HP Process Manager, and the Changengine family of products.

The crux of the company's announcement was:

- HP essentially killed-off HP proprietary middleware development (finally!);
- The company announced plans to work more closely with *software partners* (like BEA — partners who build middleware products). Especially noteworthy is that HP's focus partners in the September

announcement build Java-based infrastructure solutions (more specifically J2EE solutions).

  - Note that HP followed this Java middleware partnership announcement with a major Microsoft .NET initiative in early October, 2002. This initiative promised all sorts of technical and sales support for .NET — and positions HP to sell and support two competing architectures.

- The company announced that it will continue to focus on extending its OpenView systems/network management technologies and HP Utility Data Center "to produce a comprehensive 'management fabric' for optimizing infrastructure assets".

IBM

To understand IBM's position in the Linux marketplace it is necessary to examine IBM's strategic objectives for Linux as well as its product and services offerings. This section examines IBM's Linux strategic positioning.

Strategic Considerations — The Business Revenue Model

Linux is based on a licensing scheme called "GPL" — General Public License. This license allows users/vendors to copy and distribute copies of Linux using the GNU General Public License (and these users can make money reselling licenses with support if they so desire). But this license also states that users/developers who modify Linux and make the modified version generally and commercially available also need to publish any Linux code they modified (and make that source code generally available to the Linux community). In other words, vendors can add Linux "extensions", but they are not able to create proprietary value added extensions — instead, these extensions (if marketed commercially) need to be documented and provided back to the Linux community.

Given this business constraint, many Linux suppliers have struggled with the issue of how to make money on Linux (because its difficult to make money by adding value to the operating environment and its also difficult to make money when the operating system is available as a free download). Some vendors have tried to earn revenue by providing product support or integration services; others have tried to make money by selling Linux applications; still others have tried to make money selling hardware. IBM is one of the few companies that can make money selling all of the above.

- From a hardware perspective, IBM provides Linux on its four hardware platforms. So, even if buyers obtain Linux for free (or at a very low cost), IBM still makes money selling its hardware platforms.

- ·From a software perspective, IBM provides a complete J2EE (Java-based) Web services application development environment as well as integrated infrastructure components that run on top of the Linux operating environment. The company provides vertical middleware for program-to-program communications such as WebSphere MQ. And IBM

provides horizontal infrastructure integration components such as Lotus Domino (for Linux messaging/workflow), Tivoli (for Linux management), WebSphere (for J2EE Web services application development), and DB2 (as an enterprise-class integrated database). All of these integrated infrastructure solutions provide IBM with Linux revenue opportunities.

- From a services perspective, IBM provides education, training, deployment, migration and support services; as well as professional integration services through its IBM Global Services (IGS) organization. Professional services provisioning also provides IBM with Linux-related revenue opportunities.

How does a prospective buyer of IBM Linux solutions benefit from IBM's ability to make money by selling Linux hardware, software, and services? Where other vendors are having difficulty making their Linux value propositions economically feasible, IBM has clearly found a business model that makes investment in Linux profitable.

IBM's Linux strategy also considers "how to capture applications on IBM platforms". The goal of any systems software/hardware maker is to have the "preferred" platform that attracts the most ISVs (because ISVs bring applications to the platforms — and applications are ultimately what sells a given platform and related services).

To this "applications capture" end, IBM offers a complete J2EE (Java Enterprise Edition) development environment that runs across all four IBM platforms. With this J2EE application development environment, IS buyers and ISVs can build custom and/or packaged application solutions that can be deployed on any IBM platform or non-IBM J2EE platforms.

From a professional services perspective, IBM's Global Services (IGS) organization provides far more than just Linux support services. The company also provides a wide range of integration services, worldwide deployment services as well as outsourced management services. This ability to provide a wider range of lucrative value-added services beyond mere deployment and troubleshooting services, coupled with IGS's sheer size (over 150,000 people), positions IBM extremely well to service and support customers where other vendors struggle to do so.

Chapter Summary

The fast growth and success of Linux is seen by many people to be a direct alternate response to Microsoft business practices and pricing policies. But the Linux open source movement is not all about derailing Microsoft — its about having access to operating systems code, its about sharing intellectual property, its about lowering computing costs, and more…

There are four sources of supply for Linux infrastructure products:

1. the open source community;

2. point product solution providers;

3. infrastructure/integration software makers; and

4. hardware/software/services providers.

Each source serves the evolving Linux marketplace differently.

The open source community strives to improve the Linux kernel — while at the same time building tools, utilities, and applications that address various element of Linux infrastructure (for instance: systems management, middleware; data management, mail/messaging/groupware, etc.).

Point product solutions providers provide turnkey point solutions designed to address one particular issue (such as system monitor and control; administration). Point product solutions providers compete with each other — and often with providers of infrastructure and management software suites.

Infrastructure/integration software makers provide Linux application development tools and utilities (usually Java-based) as well as underlying infrastructure (such as software that can be used to build Linux applications that can run on horizontally-integrated mail/messaging/groupware infrastructure as well as on vertically integrated program-to-program communications middleware). Some of the products within these suites can be parsed into a la carte offerings (in essence, point products) that compete directly with product offerings from point product solution providers.

The final category of Linux infrastructure providers is the hardware/software/-service providers. These vendors build complete Linux infrastructure implementations designed and optimized to run on their respective platforms. Suppliers in this category include Dell, Hewlett-Packard, IBM, and Sun.

From a competitive perspective:

- Dell is known for Linux on Intel-based servers only (Dell does not provide heterogeneous Linux products). The company is doing very well placing Linux in HPTC environments.

- HP is still working-out its Linux strategy. The company has become a partnering company and has established relationships will all leading suppliers of the Linux operating system, as well as with suppliers of applications and databases that run on Linux. And the company has excellent Intel hardware offerings. HP needs to get its application development act together — is it strongly behind Java-based development, or .NET-based development, or both?

- IBM has the best Linux business plan in the industry. The company can make money selling Linux-based hardware, infrastructure software, and professional design, deployment, and integration services.

- Sun is in an awkward position with regard to Linux. The company has a very large Unix base to protect — and the stronger Linux gets, the more that base is threatened. Still, Sun has been active in selling Linux — especially in Grid environments.

Summary Observations

Linux is Ready

Roadmaps already exist for measuring the enterprise readiness of an operating environment. As with Unix and Windows, Information System (IS) managers evaluate operating environments on the strength of security offered; on reliability and availability characteristics; on horizontal and vertical scalability, and manageability.

Scalability

From a scalability perspective, Bloor NA's research indicates that Linux scales extremely well from a horizontal, distributed systems perspective (typically comprised of 2-way and 4-way Intel servers networked together). We also found that Linux scales well vertically to 6-way servers — but beyond 6-way environments Linux needs better memory management and better SMP support. Having said this, we also note that IBM has implemented Linux on its z-series enterprise servers (mainframes) — and offers a software management environment that allows up-to 14 instances of Linux to exploit a mainframe processor (essentially creating a very large, vertically-scaled Linux SMP environment). Linux is expected to scale well in 8-way environments in the near term (as Linux revision 2.5 8-way scalability is back-ported into Linux revision 2.4); and in 6-way environments with the next major revision of the Linux operating environment (version 2.6) due in early 2003.

Reliability

Reliability is highly reliant on system architecture as well as how applications and database use the underlying operating environment. Because 90% of all Linux is installed on Intel systems — and because Intel systems (depending on the manufacturer) are reliable — hardware reliability based on meantime between hardware component failure is generally high. Other non-Intel platforms (such as IBM's i-series and z-series) are generally known to be more reliable than commodity Intel servers — but cost more to acquire.

Availability

Availability comes into play when systems components or applications/databases fail. When this occurs, it is necessary to have a failover plan in place to failover to working hardware components or systems, or to fail over to other instances of an application or database. Failover is achieved via hardware configuration as well as through software (that redirects failed components or applications). Linux failover capabilities can be obtained from a number of sources — including the

open source community as well as from various commercial software and hardware makers.

Security

Probably the foremost consideration regarding Linux enterprise-readiness is security. And Linux security is very similar to Unix security: the underlying technology consists of password protection linked to authorization rights, secure sockets, encryption, and so on. If additional security is required, many if not most security point products that run on Unix can run on Linux.

Manageability

From a manageability perspective, Linux buyers can download various open source management tools and utilities (for data management, content management, and so on). Or, Linux buyers can purchase commercially available Linux point product solutions from companies like Heroix and Easilize. Or Linux buyers can purchase complete management suites such as IBM's Tivoli or Sun's Management Center or CA's Linux management products.

Total Cost of Ownership Considerations

Linux provides enterprise system buyers with an attractive value proposition: IS buyers can realize significant cost savings in the areas of capital expenditures, management, administration, deployment, training, and operations without sacrificing reliability, security, or scalability.

For the most part, Linux TCO can be viewed from three perspectives:

1. Costs to acquire (licensing fees of OS, systems platform, etc.);

2. Costs to manage (including support, training & implementation services and staff management costs); and

3. Costs of software (including infrastructure, applications, and databases).

Note that operating system license savings (illustrated in Table 2) are only a small part of overall cost savings attributable to Linux. Additional savings can be realized through deferring additional hardware purchases or by purchasing low-cost Intel servers; by buying only a single Linux support license for development systems and not for production systems; through server consolidation; through reduced application, database, and infrastructure license costs; and by using less expensive university-trained Linux talent.

Good advice for building a TCO case for Linux can be found on RedHat.com in the form of two RedHat-sponsored IDC reports: *The Role of Linux in Reducing the Cost of Enterprise Computing*; and *Linux: A Journey Into the Enterprise*. These reports contain Linux TCO methodology and recommendations — as well as survey results that show the kinds of TCO return-on-investments that early adopters have experienced with Linux.

Linux Infrastructure Considerations

For Linux to compete as an enterprise-class operating environment, it is necessary to augment the basic operating environment with extensions in the areas of reliability, scalability, manageability, and security. It is also necessary to build a Linux infrastructure that enables Linux systems to communicate with each other (and potentially share resources).

The big question to be asked about Linux infrastructure is: "how should my enterprise build the infrastructure to support the applications and databases that run on top of the Linux operating environment"? And the answer on how to build infrastructure is dependent on the types of applications that your enterprise intends to run:

- If your enterprise's intent is to develop custom applications, or if your intent is to run a single dedicated application on your Linux server — and if you're willing to provide developmental and support expertise to do so — then use open source infrastructure.

- If your enterprise's desire is to run a multi-application server that can work cooperatively with other distributed servers, and your enterprise is not willing to support and integrate infrastructure, applications and database, then go to packaged Linux infrastructure solutions.

> The difference between Linux and Windows is primarily a matter of the wealth of application software. Windows simply has a large number of packages than Linux and in many instances the packages do not port at all. Linux is catching up and, we believe, will overtake Windows in two or three years as the ISVs realize that Linux is a better proposition. We therefore expect to see many sites running a dual Windows/Linux strategy. IBM is pursuing a strategy of encouraging ISVs onto Linux in order to make Linux the primary *industry reference platform*. The effect will be to marginalize Solaris ahead of Windows, but eventually to marginalize Windows.
>
> **Source:Bloor Research UK**

Linux Market/Competitive Positioning

There are four sources of supply for Linux infrastructure products: the open source community; point product solution providers; infrastructure/integration software makers; and hardware/software/services providers. Each source serves the evolving Linux marketplace differently.

1. The open source community strives to improve the Linux kernel — while at the same time building tools, utilities, and applications that address various element of Linux infrastructure (for instance: systems management, middleware; data management, mail/messaging/-groupware, etc.).

2. Point product solutions providers provide turnkey point solutions designed to address one particular issue (such as system monitor and control; administration; etc.).

3. Infrastructure software-only providers come at the industry from all angles, including application integration, grid computing, and more. Very often these software makers provide comprehensive suites of software that can be used to build Linux applications; to enable Linux program-to-program communications, to enable mail/messaging/groupware, etcetera. Some of the products within these suites can be parsed into a la carte offerings (in essence, point products) that compete directly with product offerings from point product solution providers.

4. The final category of Linux infrastructure providers is hardware/software/service providers. These vendors build complete Linux infrastructure implementations designed and optimized to run on their respective platforms. (Note that IBM's Linux infrastructure can also run on other platforms from companies like Dell, Hewlett-Packard, and Sun).

Conclusions

In every category that we traditionally measure to ascertain the enterprise-readiness of a particular operating environment, Linux passes the test:

- Linux scales very well to 6-way servers today (with 8-way and 16-way support due out soon). And Linux can scale extremely well horizontally (up to thousands of nodes) using grid software.

- Linux is known to be reliable (dependent on the hardware used as well as on the application environment deployed). From a hardware perspective, 90% of all Linux is installed on Intel platforms (so if the underlying hardware has been purchased from a reputable source, hardware reliability should not be an issue). From a software perspective, drivers and ill-behaved applications have always been a source of failure for all operating environments. If an application is "well-behaved" and if drivers have been properly quality assured (QA'd), Linux will likely run in a reliable fashion.

- Linux systems can be configured for high-availability;

- Linux systems can be managed by open source tools and utilities as well as by commercially available software products;

- Linux systems are just like Unix systems from a security perspective. IS managers need to put in place proper security policies as well as products. Further, IS managers need to ensure line-level (communications/networking) security as well as system level (file/content) security. And Linux products exist that ensure both.

Further, the total cost of ownership of Linux systems can be compelling both from an acquisition cost perspective as well as from a reduced operational costs perspective.

From the perspective of Bloor Research North America there is little doubt that Linux can be successfully deployed to run mission critical applications — and is, in fact, enterprise ready.