



## **The Imperative for Linux Standards: A Recommendation for the Future**

A White Paper Prepared by the Free Standards Group

**Table of Contents**

- Section One: Background and Issues ..... 3
  - A. Introduction ..... 3
  - B. The Vision of a Standardized Linux ..... 4
  - C. Where is the LSB Today? A Summary of Recent Accomplishments..... 5
- Section Two: An Introduction to the Free Standards Group and the LSB ..... 6
- Section Three: What is to be Done ..... 10
  - A. Market Realities ..... 10
  - B. The Road Map ..... 11
- Section Five: Summary ..... 14

# Section One: Background and Issues

## A. Introduction

An operating system's success is inextricably linked with the number and quality of applications that run on top of it. Linux and its variances between distributions, however, present ISVs with a unique set of challenges: different distributions of Linux make use of different versions of libraries, there are important files stored in different locations, and so on. If an ISV wants to reach a global Linux audience they must support more than one distribution of Linux. These challenges and variances make it difficult – and costly – for ISVs to target the Linux platform. The Linux Standard Base was created to solve these challenges and lower the cost of supporting the Linux platform.

Even before Linux became an enterprise operating system, the Linux community was concerned about fragmentation and its effects on software developers. In response, the community determined that a binary standard for Linux was important, not only for success in challenging Microsoft Windows, but for guaranteeing a broad and deep availability of applications for the platform. Out of this concern, the community banded together to form the Free Standards Group, a standards body tasked with developing open, international standards that would deliver on the vision of portability within a competitive Linux distribution ecosystem.

**The Way Ahead:** The purpose of this paper is to describe the roadmap of the Linux Standard Base (LSB). It begins by examining the history and current state of the LSB by exploring the support it receives from the Linux ecosystem: distribution vendors, independent software vendors (ISVs), system vendors, end users and other open source projects and organizations.

This paper will also present a detailed roadmap on how to secure Linux for the future. While we may never achieve the nirvana of “write once, run everywhere,” the goal of growing the Linux market by making it easier for applications to target the Linux platform is within sight. Conversely, the danger of a “Tragedy of the Commons” is also possible. Without a widely supported binary standard for Linux, a single vendor, de facto standard will emerge, effectively removing choice and locking end users in. Truly, this would be a sad shortfall to the original promise of Linux, and an unfortunate benchmark for all open source software.

The vision of a standardized Linux is clear. The remaining task is to agree on the best way to get there. This paper will describe that way.

## B. The Vision of a Standardized Linux

**All for One:** The vision of a standardized Linux must balance the needs of the competitive distribution ecosystem with the requirements of end users and independent software vendors for interoperability. The Free Standards Group and the Linux Standard Base were created to exactly balance the needs of those various groups.

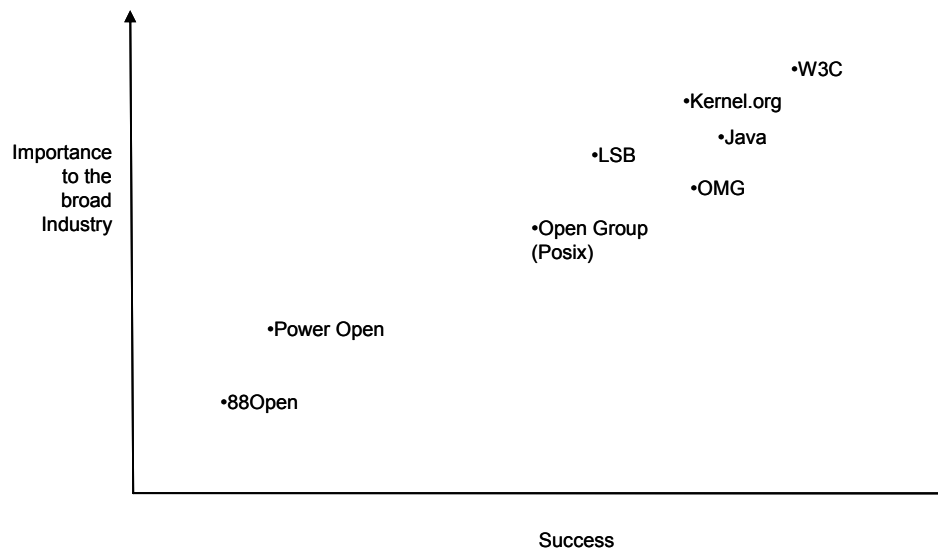
What does a standardized Linux ecosystem look like?

- a healthy distribution network competing on support, service, security, price and other factors based on OEM requirements for LSB compliance
- broad availability of applications for the Linux platform, covering everything from the most complicated data center systems to shrink-wrapped consumer applications available at retail outlets
- an open standard that allows ISVs to write their application to the Linux platform at a low cost
- broad choice of application software for Linux
- reduced support costs for ISVs and systems vendors since all have a clear set of application and distribution guidelines that inform their software and hardware development
- reduced development costs for distribution vendors as a base set of commonality exists, leveraging multiple vendors and allowing them to focus on innovating at the “unique” higher levels of functionality

What type of organization(s) could deliver on the promise of such a world? The experience of the past tells us that organizations that are open to all stakeholders, and that provide appropriate input and influence to all those affected, are those that achieve their objectives. A consortium’s success at setting open standards correlates highly and directly to the absence of proprietary advantage to any individual company. Those that have instead been structured to give undue influence to one, or only a few, vendors have been rightly recognized (and therefore shunned) by those that do not wish to be duped into helping create what proves to be closer to a proprietary solution.

Throughout history, successful consortia have therefore relied upon achieving a high degree of agreement among a critical mass of large and small companies alike, both hardware and software alike. Often, a common enemy has served to rally the community as a whole to set an open standard to avoid the dominance of a closed, de facto standard. Not surprisingly, the most successful and stable consortia are those whose purposes are most beneficial to the industry as a whole, and therefore receive the market support of non-members and members alike.

**One for All:** The Free Standards Group is no exception to this rule (see figure 1). ISVs, end users and distribution vendors all benefit from a well-supported binary standard.



**Figure 1 - Correlation of standard's nonproprietary importance to industry and consortium's success.**

Of course, expectations must be realistic, and it has been a common error in the past to over promise what open standards can deliver. Even with an open source operating system like Linux, there will always be technical complexities that result in some ISVs needing to do more portability work than others. But a robust and comprehensively supported standard will eliminate much of the heavy lifting required today, delivering huge economies to the marketplace, and therefore very substantial incentives to build and use products and services that are based upon the Linux environment.

In the vision of a standardized Linux, the ecosystem (and all who depend upon it) thrives when Linux thrives, providing a true, open – and very welcome to customers – alternative to the proprietary computing platforms of old.

### **C. Where is the LSB Today? A Summary of Recent Accomplishments**

*“There's no question in our mind that LSB 2.0 represents the best chance to prevent Linux fragmentation. Customers and vendors alike have plenty at stake in the existence of a robust, widely supported standard. We urge all parties, enterprise application vendors especially, to back LSB 2.0 wholeheartedly.”*  
- eWeek

**Past and Present:** What is the promise of application portability today? While the FSG has done fine technical work, its impact on the marketplace in its first years was less impressive. In the last year, however that has changed, and we are already well on our way to a standardized Linux world. Through the efforts of the FSG, the Linux Standard Base has achieved great strides in the past year, including the following key accomplishments:

- LSB 3.0 resolves key issues between major distribution vendors, including C++, resulting in the announced intention by all major Linux distribution vendors to certify on LSB 3.0
- For the first time, major ISVs such as Veritas, Oracle, MySQL, BakBone and others have either joined the FSG or given their public support of the standard
- New memberships have increased by 70 percent over the last year, including the addition of over a dozen ISV members where there was previously no ISV participation at all
- Funding has increased by 40 percent
- The Chinese Government has signed an agreement to use the LSB as the base of its emerging national standard for Linux, and to become a certification authority
- The FSG has experienced dramatically increased visibility and awareness of the standard through new marketing efforts, including features in *The Wall Street Journal*, *Business Week*, *USA Today*, *eWeek* and *the Associated Press*
- A book, *Building Applications with the Linux Standard Base*, was published in 2004
- The LSB has been submitted to ISO for approval as an international standard; the vote will conclude in October 2005

## Section Two: An Introduction to the Free Standards Group and the LSB

**Origins, Mission and Action:** The Free Standards Group was formed in 1998 to promote open source software through standards. Today it is led by executive director Jim Zemlin and a board of directors representing key community and corporate individuals directly involved in the world of Linux. The Free Standards Group has two primary working groups, focusing respectively on:

- **The Linux Standard Base (LSB):** a binary standard for interoperability between applications and the Linux platform
- **Open18N:** a standard that creates a foundation for language globalization of compliant distributions and applications

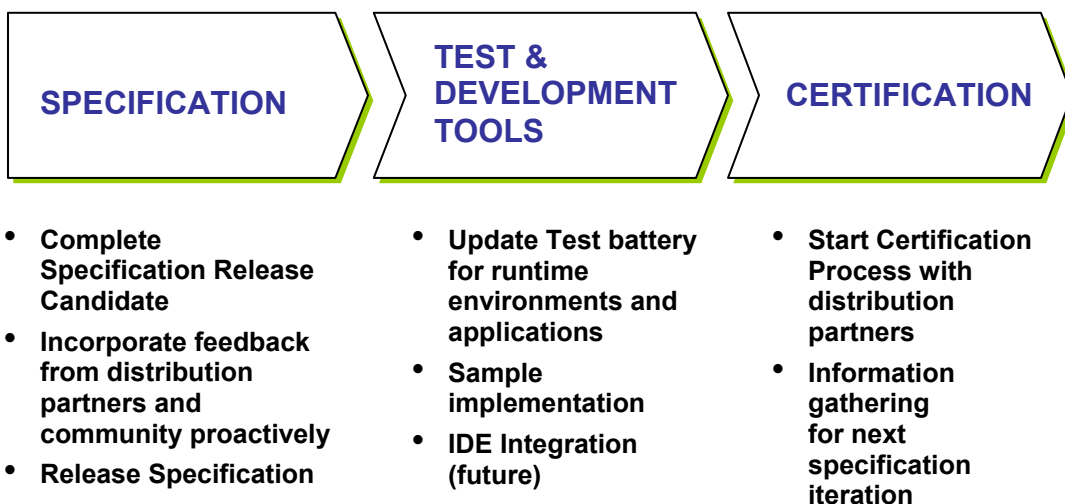
Today, the FSG concentrates its efforts in the following areas:

- Developing and improving existing standards
- Developing and implementing testing and certification programs in support of its standards
- Conducting outreach and education campaigns to encourage ISVs to target the Linux platform, providing technical support and resources
- Enforcing the LSB brand with compliant distributions and applications

Currently the FSG is supported by major vendors in the world of Linux including Red Hat, Novell, MandrakeSoft, Conectiva, Progeny, TurboLinux, Red Flag Linux, Miracle Linux, Beijing Co-Create Software Company, Sun Wah Linux, Thiz Linux, IBM, Intel, HP, AMD, Dell, Sun Microsystems, Veritas, BakBone, Google, Trolltech, Cray Computer, Montavista, VA Software, UGS, and many more.

### The LSB:

The Linux Standard Base is a core standard for the Linux operating system that encourages interoperability between applications and the platform. It includes a written binary interface specification, a set of test suites for both distributions and applications writing to the standard, and a sample implementation for testing purposes. The release process of the specification is indicated in figure A1.



**Figure A1:** Major LSB components and release process

The LSB makes use of existing standards whenever possible, including liberally using the POSIX set of specifications widely used for UNIX. Unlike other API standards, however, the LSB concerns

itself with the binary level and thus specifies binary interfaces only. Open systems and open API standards have enjoyed source portability for years. Unfortunately, source portability is not the solution for creating a large consumer market. Without binary portability, applications used by the mainstream market will not increase<sup>1</sup>.

Since it is a binary specification, the LSB is divided into both general and processor-specific components.

|  |
|--|
| The LSB Specifies:   |
| Common Packaging and Install Guidelines                          |
| Common Shared Libraries and their Selection                      |
| Configuration Files  |
| File Placement (FHS)   |
| System Commands  |
| ABIs for System Interfaces (both application and platform level) |

**Figure A1.2:** LSB Description

The LSB currently covers seven computing architectures including: IA32, 32-bit PowerPC, Itanium, 64-bit PowerPC, 31-bit S390, 64-bit z/Architecture.

The Free Standards Group also coordinates testing and certification programs that verify software compliance with existing standards. Currently, the Open Group administers both application and distribution certification tests on behalf of the FSG.

**What about a Reference Implementation?** Everyone agrees that the simplest technical solution is for ISVs to have one Linux implementation to target and test against. This is commonly referred to as a reference implementation. While this is a technically elegant solution, there are simply too many technical, competitive and political realities specific to open source that make this approach unfeasible. The LSB binary interface approach allows distribution vendors to comply with the standard, control their own engineering and release cycles while still allowing them to differentiate their product offerings. It allows them to provide the value and service they offer today (and maintain a robust ecosystem driving innovation around Linux) but still deliver a common set of interfaces for ISVs. With the support of the distribution vendors for LSB, this approach clearly has the support of the entire community.

**How does the LSB Approach differ from the Failed Attempts to unify UNIX?**

*“Without the LSB, Linux could end up, as so many of the Unixes have, permanently parked on the side of the operating system road.”  
- eWeek*

**Linux is not UNIX.** Is Linux doomed to the same fate as UNIX? Is resistance futile? Happily, the answer to both questions can be “no”. The reason lies not only with Linux, but with timing and the LSB.

**Timing is Everything.** Most obviously, the efforts to unify UNIX occurred after fragmentation was not only in evidence, but after numerous large vendors had vested interests in maintaining that fragmentation to greater and lesser degrees. Moreover, each vendor was also hedging its bets – while publicly expressing their support for UNIX, many vendors had internal Windows NT programs with equal (or greater) funding and human resources. The result was ambivalence at best.

At this juncture in time for Linux, real fragmentation has not yet occurred, and the best interests of the vast majority of players still demonstrably lies with maintaining coherence – there is no “plan B”.

<sup>1</sup> See **Building Applications with the Linux Standard Base** for more information on this topic.

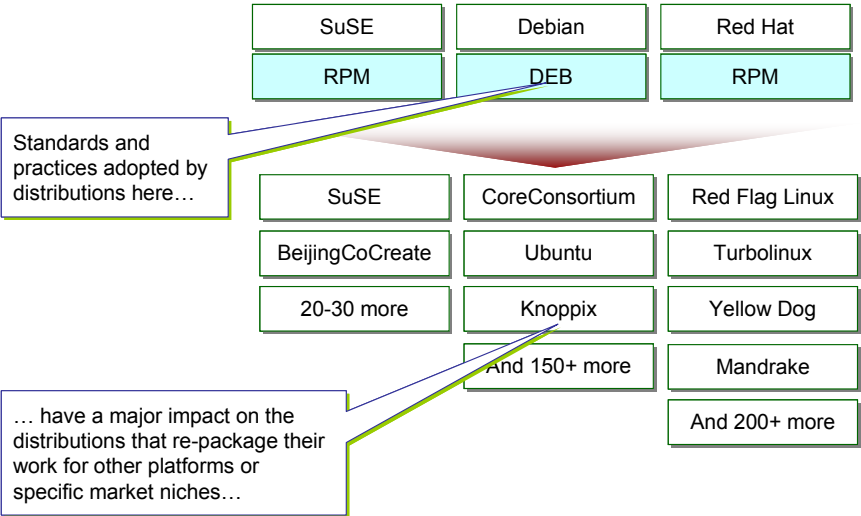
**A Different Technical Approach:** Community groups undertook many initiatives to standardize the UNIX operating system: most notably, the Single UNIX Specification (SUS) and Portable Operating System Interface (POSIX). Most UNIX standardization efforts focused on application programming interfaces. Eventually, the POSIX standards (administered by the Open Group) became widely adopted and useful for the industry, even though they did not deliver a unified UNIX. And while these were worthy, state of the art processes, they pre-dated the open source technical and legal models that provide for greater protection against proprietary, closed systems, and for greater enthusiasm at the engineering level.

At the same time, the LSB builds on earlier efforts that attempted to prevent Unix fragmentation, such as the POSIX and the Single Unix Specification (SUS). In fact, it uses some of POSIX's own source code standards and SUS' interface definitions.

But while the LSB has incorporated the durable aspects of these precursors, the FSG has learned from the UNIX experience, and the LSB has therefore not adopted the limitations of POSIX and SUS. Notably, POSIX only defined programming interfaces and therefore could not guarantee binary compatibility. At the other end of the spectrum, standards such as OSF/1, which aimed for binary compatibility, were found to be too restrictive. The LSB aims to strike a balance between the two approaches -- it includes a binary compatibility layer that splits the difference between the approaches taken with POSIX and OSF/1.<sup>23</sup>

**A New Socioeconomic Environment:** Unlike the failed attempts to provide binary interoperability for UNIX, the LSB lives in a very different computing and business environment. A maturing of the systems software market combined with the open source software movement has caused system software to become a commodity. Companies no longer wish to differentiate wildly in order to generate the large software margins that characterized the age of UNIX. This discourages distribution vendors from diverging from an interoperable Linux. In fact, the greater the number of pieces of a distribution that are provided in standardized form, the more the vendor of that distribution can focus its attention and resources (both technical and marketing) on providing the sort of differentiating value that customers are still willing to pay for: service, support, management software, and so on.

**The Open Source Difference:** A dramatic differentiator between UNIX and Linux lies in the (then unknown) licensing structure of Linux: the GPL. By its very nature, the GPL allows distribution vendors to copy the work of other distributions. This results in much of the same code base between divergent distributions, appreciably narrowing the technical challenge, and wholesome entropy against change. The key challenge for open source and Linux is to prevent the difference in distributions that the GPL was not designed to solve: file hierarchy, configuration language, runtime environment compatibility, etc.



<sup>2</sup> CNET, "Linux: the Forking Fight Back," Feb 7, 2005

<sup>3</sup> For more on the technical approach taken by the LSB, see Appendix B.



**Figure 2:** GPL and Standards: the trickle down effect.

**The Odds:** In summary, the odds are simply much, much better that efforts to prevent the fragmentation of Linux will be successful. The GPL, the relative uniformity among Linux distributions, the fact that FSG and other organizations with similar concerns got off to an early start, the fact that there is a common, proprietary operating system enemy, all make the vision of the LSB easier to achieve – and therefore make it much easier for stakeholders to buy into, and commit to, making the vision real.

# Section Three: What is to be Done

## A. Market Realities

**An Ecosystemic Strategy:** A standardized Linux environment will benefit a rainbow of stakeholders: most obviously, systems vendors, distribution vendors, ISVs and end users. However, in order for those benefits to be realized, the buy-in of all of these same stakeholders is required. The best way to assure the actual use of the LSB and its attendant tools is to directly involve each group in its creation, so that they have invested their resources and made their strategic commitments at an early date.

**Chickens, Eggs and Distro Vendors:** Early implementation of interoperability standards, by its nature, involves a gamble, since the value of such a standard is based on “network effects” (i.e., a telephone is at best ornamental until someone you want to talk to buys one as well). Consequently, before vendors will support a standard, they want to see everyone else supporting it. The LSB must address this same reality.

In the particular case of Linux, as the market exists today, the distribution vendors are the enablers of the standard. Without their participation, the standard cannot achieve any success. [See figure 3.1] And without their participation in the creation of the LSB, their support for it would be unlikely. It is vitally important that *all* major distribution vendors have pledged to certify on LSB 3.0. This is a critical factor that will assure all other stakeholders that they, too, should become involved not only in development, but with implementation – thus solving the first situational conundrum of the chicken and the egg.

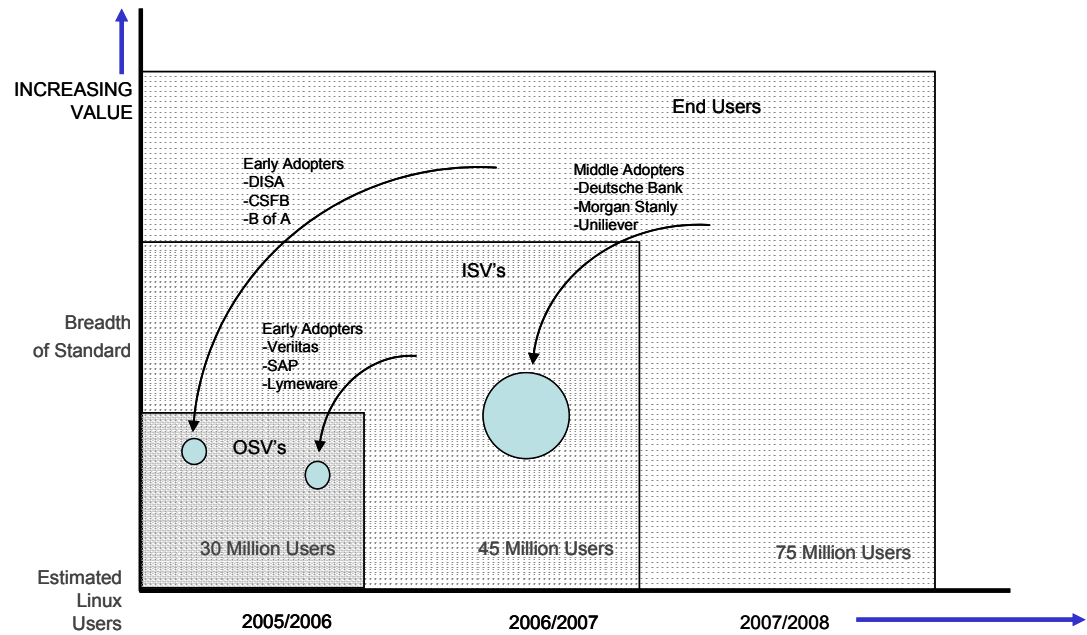


Figure 3.1: The necessary order of LSB adoption

**ISVs and the Party:** But ISVs are also vital to the success of the LSB and the standardization of Linux. There are strong incentives for ISVs to support the LSB effort as well, since they will directly benefit in lower porting, testing and support costs and through an increase in market reach. Since Linux is already an established and growing presence in the marketplace, the issue is not whether ISVs will port to the Linux platform, but how they can do so most quickly and cheaply – the LSB provides the best answer to that question. Not surprisingly, the Free Standards Group has also made good process with obtaining early support from many key ISVs: Oracle, Bakhone, VERITAS, IBM Software, Novell, Levanta, MySQL, Hyperic and many others have recently joined or given their support openly.

It is critical that the FSG recognizes that application certification is not an early indicator of success but rather a later milestone when shrink-wrapped Linux applications reach critical mass in the consumer market. At that point, consumers will look to the LSB branded mark for assurance of interoperability. It is imperative that FSG builds awareness of the realities today and how ISVs can benefit by building an application to the LSB without requiring certification.

**Risk Management for End Users.** End users are the tailing adopters of the standard, yet they can assert significant influence on both distribution vendors and ISVs. End users need to see the LSB as a form of risk management in their Linux strategy. There is proof that acceptance has begun with early adopters. Media reports have stated that a handful of large Fortune 500 companies have stipulated LSB compliance in their support contracts. This contractual obligation will accelerate the adoption of the LSB. The Free Standards Group has just begun engaging proactively with this group. Google, the largest Linux end user, has recently joined the Free Standards Group. Their support and interest in using the standard for both their custom distribution and applications on top of their distribution is a significant milestone in the FSG's recent accomplishments.

**Seven Architectures.** Hardware vendors benefit from the standard by enabling application vendors to easily target their architectures. Currently the LSB supports seven architectures. This allows for robust competition among a variety of architectures without locking those hardware vendors into any third party or proprietary vendor's software offering. What's required of this constituency? They must write LSB compliance into their OEM agreements with distribution partners. This gives the standard a meaningful economic driver while preventing the hardware vendor from being beholden to any third party now or in the future.

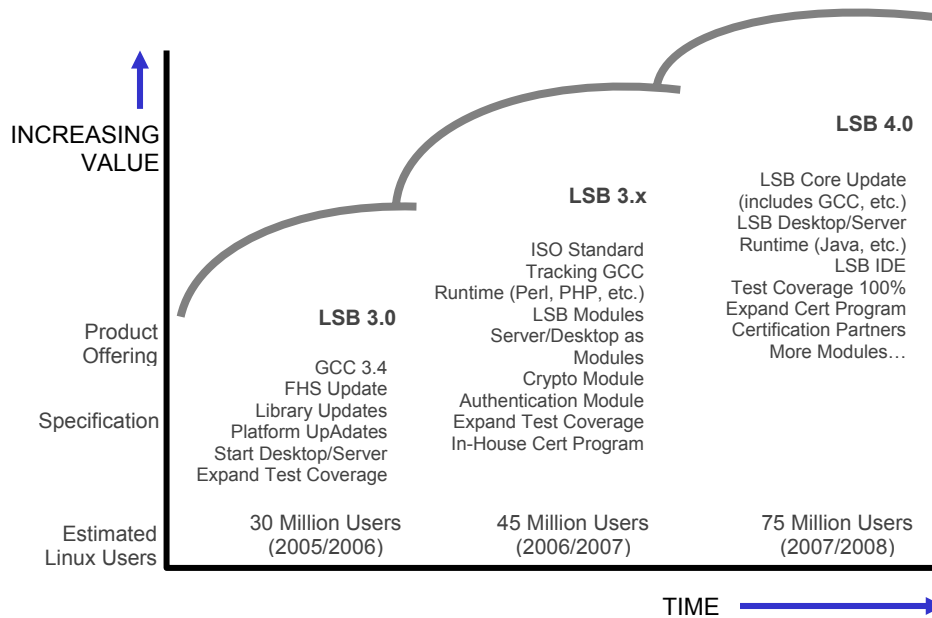
**The Role of the Community.** The open source community represents an amalgamation of software projects which are integrated into a single computing solution. It is important that the maintainers of those projects are aware of existing computing standards such as the LSB so they can work in a cooperative fashion to accelerate the adoption of their technology. Currently industry groups like the OSDL support the LSB. In fact, the OSDL has made the LSB a key part of their DCL and CGL offerings.

## **B. The Road Map**

The Linux Standard Base – supported by compliance testing and a certification program – is the firewall that we can build that will protect vendors and users alike from both deliberate as well as negligent degradation and fragmentation of Linux.

Our vision is to achieve the goal of binary compatibility while being sensitive to the needs of the distribution companies; the LSB aims to look at only what needs to be standardized to minimize the effort required by ISV's to market across Linux yet allows the distribution companies to differentiate and add value.

**The Detailed Roadmap.** In 2004, the Free Standards Group conducted a survey of all key members of its community: ISVs, distribution vendors, end users, system vendors. It looked at their requirements for the Linux Standard Base and created a detailed roadmap of what needs to be standardized to achieve Linux interoperability. At the same time, it became clear that this perfect world would take time to achieve. Standards – just like technology – are never done. The LSB roadmap is driven by both the needs of the various members of the consortium and the ability to gain consensus around key components. For instance, C++ was a key element missing from LSB 1.0. This has been addressed in 2.0 and 3.0, but it's very clear that we need to go farther. We've summarized the core elements of our roadmap below.



**Figure 3.2:** LSB Roadmap

Simply as a strategy to break the problem down, we have separated the work into **server** and **desktop** initiatives.

**Server.** Future work on the LSB Server module consists of tracking the LSB core and adding modules that ISV's have requested. We expect to start on this work mid-year 2005 and progress until complete. We expect to have the most commonly requested runtime libraries and the server portion of the work completed through the issuance of LSB 3.x point releases. These releases will expand but not change existing aspects of the overall specification. This is a significant amount of hard work. Key components include:

- *LSB Core, Maintain lsb-core and track evolution of lsb-core with ISO*

We do not plan to initiate major changes to lsb-core, but we need to track changes to upstream projects and upstream specs. The former includes glibc; there are features in glibc/kernel which will mature to where they are ready for the specification. Identified areas include message queues, timers, and advanced threading interfaces. The latter includes the POSIX spec (entering a revision cycle), and ELF and DWARF which need revision to reflect current practice (LSB will need to drive some of the latter).

- *LSB Runtimes*

The LSB must include languages which don't (usually) run native binaries but have some kind of a runtime environment platform: Java, Python, Perl, PHP.

- *LSB Identity Management module(s)*

Identity management is a key requirement. LSB must contain security-related features centering around cryptography and authentication such as access control lists; ssl, ssh, Kerberos, LDAP, and possibly PAM.

- *LSB Systems Management module(s)*

Systems admin and manageability has been excluded from LSB to date. While much of this area is still seen as an area for distribution differentiation, we can add significant value at lower levels by addressing a few basic sys admin enablers (like device mounting and unmounting), as well as SNMP, NIS/YP, and authentication.

**Desktop.** The desktop effort actually builds on the work done for the server. The focus is currently on graphics, though some device, sound and game controller issues are currently being discussed. Key components include:

- *LSB Graphics module: update list of core X11 libraries*

The current list of X11 libraries is quite limited, New libraries have emerged as widespread "best practice" since the original X11 snapshot was taken, and these are needed as enablers for higher-level standards, particularly Desktop. There is also a set of small image libraries that are needed either in this module or in Desktop for the same reason (enablers). The final component of lsb-graphics is OpenGL, where the base specs have evolved and LSB needs to realign.

- *LSB Desktop module*

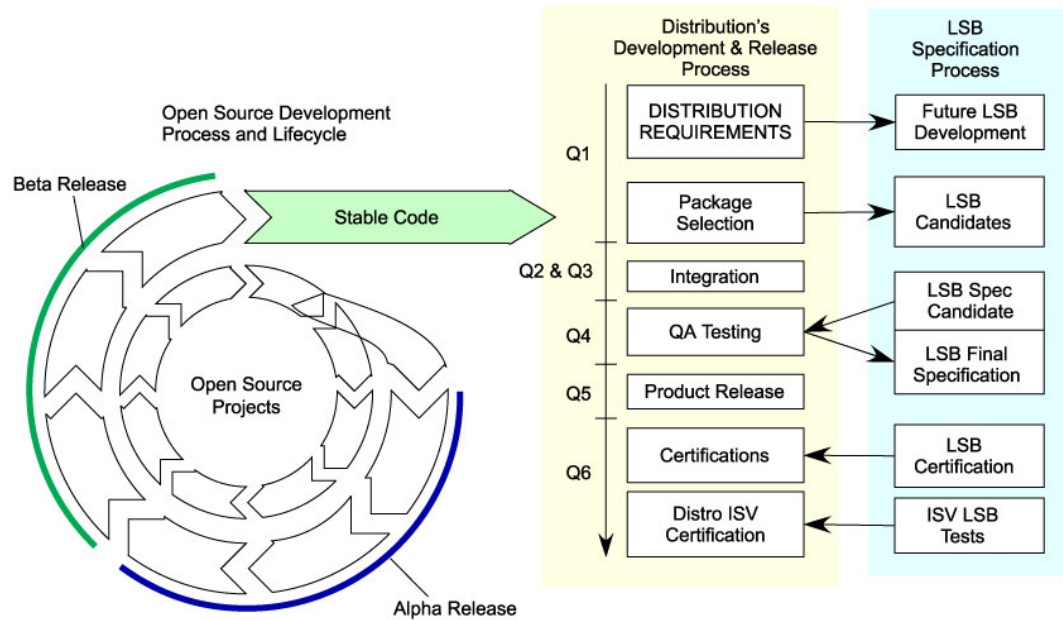
We must create enough of a baseline to make building desktop applications feasible. This will entail work on core Gtk/Gnome libraries, and depending on the ability to remove an IP-encumbrance issue, the same for KDE.

### **Accountability**

**Measuring Success.** Success metrics are objective, measurable targets for the value customers want to experience. In terms of the specification, we will always be balancing the needs of our member partners versus the goal of documenting common standards. Members, partners and participants contributing to the LSB effort want to see their energy put to good use – they want “buyer satisfaction.” Simply put, while they see the wisdom in supporting the community they want to see strategic value for their invested resource.

The Free Standards Group's customers and key beneficiaries are the distribution vendors, independent software vendors, system vendors, and independent software developers. This consortium sets the expectations by which we are bound to deliver value in the form of specifications, test suites and a sample implementation.

In order to win the business of these customers, the LSB requires a predictable, cyclical process whereby feedback can be integrated into the specification. This process must always be examined from the customers' perspective. It must have a unique, well defined methodology whereby value is added at each stage in the cycle. See figure 2.0



**Figure 2.0:** Where the LSB fits into the Linux development and release process.

The FSG will use the following as key measures of success of the technical roadmap:

- Number of LSB compliant distributions by both quantity and market share
- Number of total dollars saved by ISVs by writing to the LSB
- Reduced complexity encountered by ISVs
- Number of applications targeting the Linux platform

## Section Five: Summary

To summarize the points made in this paper:

- The LSB and Free Standards Group have made great strides in the last 12 months, cultivating new levels of support and consensus within the community and achieving several significant milestones.
- Aggressive action is imperative at this time to build upon success to date to achieve the next critical milestones. This is required in order to prevent the fragmentation of Linux.
- The organization has developed a detailed roadmap through an ISV and system vendor audit conducted in 2004 that we believe will be highly effective in reaching those milestones.

The Free Standards Group calls on members of the Linux ecosystem to join the organization and use the standard. Without a commonly supported standard, Linux will fragment: software applications will decrease while usage of the operating system grinds to a halt. By supporting the organization and working toward the goal of increase portability between applications and Linux, your organization and the entire Linux community will benefit.