

Standards Reports

The Linux Standard Base: How Will It Benefit Linux?

by Bill Claybrook

Aberdeen Group

claybrook@aberdeen.com

[Editor's Note: This is the kind of document that IT VP's in large corporations read all the time. It's interesting to see how it's written and how it asserts various points of view in a persuasive way. It reminds me ever-so-slightly of the old /usr/group standardization process.

It is also interesting to try to discern where money is moving and to notice the reliance on brand names, logos, and trademarks. A huge part of the IT world relies on these sorts of assurances rather than employing their own technical gurus. Enjoy!

This article is reprinted with the permission of Aberdeen Group, and is the intellectual property of Aberdeen Group.

Linux is knocking on the enterprise door – and quickly making inroads. The speed with which Linux enters the enterprise depends to a large extent on the availability of enterprise applications running on Linux. The Free Standards Group is helping to accelerate this effort via its Linux Standard Base (LSB) and Li18nux initiatives. This *InSight* gives a preview of the LSB and outlines the benefits associated with deploying LSB-compliant software.

Linux Gets a Boost - Standards

Linux has the fastest market growth rate of any operating system (OS) in the world, and it is being developed faster than any OS in history. However, for Linux to move into the enterprise and serve as the host for enterprise applications, a set of common standards is needed. Without standards and common application programming interfaces (APIs), independent software vendors (ISVs) and other developers must decide which of the more than 250 Linux distributions to target for hosting their applications. On the upside, standards will bring a greater market share for Linux, allowing for its continuing growth. Developers, distributors, ISVs, systems vendors, and especially users will surely benefit.

In response to the need for Linux standards, the Free Standards Group (FSG) has initiated two key projects – the Linux Standard Base (LSB) and the Linux Internationalization Standard (Li18nux).

The LSB provides binary and behavioral protocols that increase compatibility among Linux distributions; it also enables LSB-compliant applications to run on any LSB-compliant Linux distribution. Li18nux is a standard for creating a foundation for language globalization of Linux-compliant distributions and applications.

Note that LSB 1.1 compliance and Li18nux 1.0 compliance are two separate concepts. Thus, a distribution can be LSB 1.1 compliant without being Li18nux 1.0 compliant – and vice versa. (The focus in this article is LSB 1.1.) The FSG Workgroups for LSB 1.1 and Li18nux 1.0 will deliver complete, written documentation articulating what is needed to make Linux distributions and other Linux-based software compliant. In addition, they will deliver compliance

Our Standard Report Editor, David Blackwood, welcomes dialogue between this column and you, the readers. Please send your comments to dave@usenix.org.

test suites for each standard. For additional LSB and Li18nux information, see <http://www.freestandards.org>.

The FSG is an independent, non-profit organization dedicated to accelerating the use and acceptance of open source technologies through the development and promotion of standards. The FSG is backed by important industry leaders including Caldera, Compaq, Debian, Dell, Hewlett-Packard (HP), Hitachi, IBM, Intel, Oracle, Red Hat, Sun, SuSE, Turbolinux, and key members of the open source development community.

Motivation for Linux Standards

For the past several years, ISVs have struggled with the task of porting their applications to multiple OS platforms including several Unix/RISC platforms. Porting efforts are often time-consuming and costly. Once a port is completed, the ISV must support it.

ISVs want access to as many markets as possible, but they do not want to repeat the Unix/RISC phenomena – making multiple ports to Linux. ISVs want to do a single port to Linux that will run on many Linux distributions. Red Hat is the largest Linux distributor by far, but other distributions find significant favor in many geographical sectors around the world. The LSB may be interpreted by some as a way to “level the playing field” for distributors, but the real significance of the LSB rests in making long-term enterprise Linux deployments possible.

Linux Standard Base 1.1

LSB 1.1 standardizes the core functionality of Linux and core/primary libraries. Figure 1 depicts how a Linux distribution will be viewed post-LSB 1.1. At the lowest level are the Linux kernel, drivers, and hardware (architecture)-specific code. None of this code is directly affected by LSB 1.1. Above the Linux kernel is the LSB component. It

also contains some hardware-specific code. The LSB component is expected to grow over time with the addition of libraries, e.g., C++, but only those that are stable (subject to infrequent change) will be added to the LSB component.

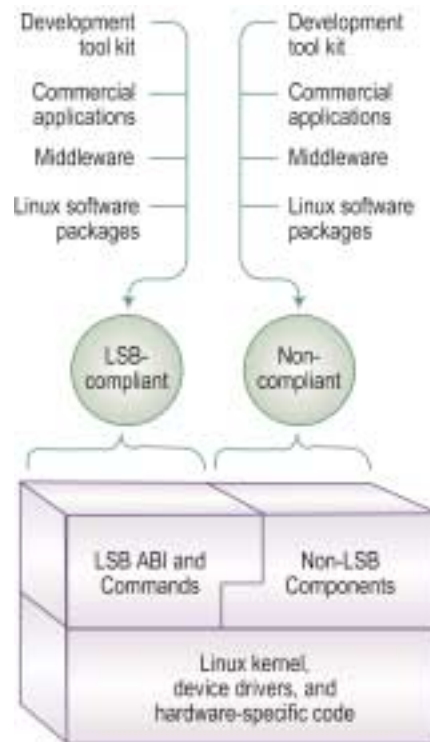


Figure 1: LSB View of Linux Distribution
Source: Aberdeen Group, March 2002

LSB 1.1 defines an Application Binary Interface (ABI), which is very similar to the POSIX.1 and POSIX.2 source specifications. The LSB's ABI defines the API for developers to program to, and it includes runtime definitions for binary compatibility.

Presently, an LSB-compliant Linux distribution must supply 15 specific libraries that define the ABIs. What LSB 1.1 does not define, applications must not use. For example, the vi (1) editor is not LSB-defined or LSB-compliant; therefore, it should not be called, via `popen`, by an application (`popen` is a

system call that opens a pipe to a process).

Outside the bounds of the Linux distribution, but with direct access to the LSB-specified ABI, are the development tool kit, commercial applications, etc. One of the features of compliant development tools is that they will restrict potential missteps in developing compliant applications.

Today, the GNU tool kit, which includes the gcc compiler, is the only tool kit that has been adopted for the development of compliant applications and middleware. In the future, however, other tool kits could be shown to be compatible for developing compliant applications.

Any software package or library that is not specifically included in the LSB 1.1 specification is considered to be outside the scope of the standard. Adding a nonconforming package to an otherwise conforming Linux distribution does not prevent the distribution from conforming, unless the package removes or replaces a conforming part of the distribution.

If an application developer goes outside the LSB ABI, there are special rules to follow. These rules may require the application developer to static-link modules or bundle them with the application to attain compliance.

LSB Certification

The LSB certification program is currently under development; therefore, some of the information provided below may differ slightly from the final published details. LSB 1.1 specifies separate sets of compliance test suites for Linux distributions, Linux applications, and Linux build environments. The LSB Workgroup is responsible for developing the test suites and making them available.

For application testing, the LSB checks to determine if the applications are using only LSB-specified ABIs and libraries, the LSB runtime linker, and executable and linking format (ELF). In addition, a set of Filesystem Hierarchy Standard (FHS) questions is asked about the installation of the application. Lastly, the application is required to be tested to work on the LSB Sample Implementation (a minimal Linux system that contains mostly just what is specified by the LSB) and two LSB-compliant distributions. ISVs and other application developers can get a preview of the types of problems that they may encounter in becoming LSB-compliant by running their application(s) on an LSB-compliant distribution (or the LSB Sample Implementation) and examining the error reports.

To successfully qualify as an LSB-compliant Linux distribution, a distribution must supply the 15 libraries that define the ABIs, adhere to the FHS, successfully run a set of test applications, etc. For the Linux build environment, the LSB-specified ABIs and header definitions are checked to determine if an LSB-compliant application can be built. (LSB 1.1 provides a sample build environment from which to create an LSB-compliant application).

How to (and Who Will) Do the Certification?

Certification will involve at least Linux distributors, systems vendors with middleware for Linux, and ISVs.

The FSG will not likely become involved in the actual LSB-compliant testing and auditing efforts, but this strategy could change over time. However, it will become involved in licensing third-party compliance testers – to ensure the quality of LSB compliance testing. LSB platinum members – large companies like IBM and HP – can complete compliance testing and stick the LSB-com-

pliant logo on their products, indicating that they have been certified. However, other companies could be required to use independent third-party companies to obtain the logo.

Linux distributors, such as Red Hat, SuSE, etc., are responsible for certifying their distributions on the various hardware architectures – such as IA-32, Itanium, and RISC – that they support. Systems vendors like HP, IBM, Compaq, Dell, etc., will “trust” that Linux distributors have certified these distributions on the architectures that they (the distributors) support. IBM, for example, will not need to repeat the SuSE Linux certification process when it ships (or supports) SuSE Linux on IBM servers.

There are two levels of certification. First-level certification involves, for example, an ISV like Oracle running the LSB-compliant application test suites against one of its applications on two LSB-compliant Linux distributions, registering the results with the LSB, and then proclaiming that the application is LSB-compliant.

Second-level certification involves hiring a third-party testing company to rerun the tests for an application on several other Linux distributions of choice. The testing results are registered, the provider signs an agreement to comply, and the FSG brands the product on behalf of the LSB.

Vendors who develop middleware or other Linux-based software could do self-compliance or third-party testing. The result is an LSB-compliant middleware product that is guaranteed to work on many LSB-compliant distributions.

Conclusions

The LSB has the support of many of the distributors and suppliers who develop Linux-based solutions. Credit Suisse First Boston (CSFB) has already announced that it will not deploy non-

LSB 1.1 software after the end of 2002. According to our research, any Linux distributor or Linux-based software supplier that wants to do enterprise business will have to produce LSB 1.1-compliant products. LSB 1.1 and Li18nux 1.0 are good starts by the FSG to produce standard interfaces for the development of software for Linux. Standards make the long-term deployment of Linux in the enterprise more attractive and less costly for suppliers and users.