



Standards: Bridging the Generation Gap between Linux® and UNIX® Systems

Andrew Josey
Director of Certification
The Open Group
Email: a.josey@opengroup.org

UNIX is a registered trademark of The Open Group
Linux is a registered trademark of Linus Torvalds
30-01-2002

Agenda

□ **This talk covers...**

- A review of the latest open systems standards
- How Linux shapes up to them

Agenda (Cont'd)

- Standards Projects
 - POSIX
 - The Single UNIX Specification
 - The Linux Standard Base (LSB)

"Despite their well earned reputation as a source of confusion, standards are one of the enabling factors behind the success of Linux. If it weren't for the adoption of the right standards by Linus Torvalds and other developers, Linux would likely be a small footnote in the history of operating systems. "

- Dan Quinlan, Free Standards Group Chairman

The Free Market

- ❑ The key to the growth of the Linux market is the free-market demands placed upon suppliers by Open Standards
- ❑ These systems ultimately compete on quality and added value features to retain customers
- ❑ Dissatisfied customers can move onto another supplier

Source vs Binary Standards

**Source
API Stds**

**Specify
Linux
Behavior**

**Binary
Stds**

**Source
Portability**

**Shrink
Wrapped
Applications**

Agenda

- **Standards Projects**

- **POSIX**

- **The Single UNIX Specification**
 - **The Linux Standards Base**

POSIX® /pahz-icks/

- ❑ *POSIX* , pronounced *pahz-icks* as in positive, not *poh-six*, or other variations
- ❑ POSIX is a registered trademark of the IEEE
- ❑ An acronym for Portable Operating System Interface

What is POSIX?

- POSIX is the codification and de jure standardization of the common core of UNIX® practice, and now of Realtime OS practice
 - The basic objective is portability of both programmers and application source code
 - Portability of the OS kernel itself and/or application binary code are not objectives

What is POSIX (Cont'd)?

- POSIX is a set of books specifying APIs
 - It is neither a piece of code nor an operating system

What is an API?

- ❑ Application Program Interface
- ❑ A written contract between system developers and application developers
- ❑ Its not a piece of code, it is a piece of paper defining what the two sets of developers are guaranteed to receive and are in turn responsible for providing

POSIX® /pahz-icks/

- ❑ Today POSIX is a family of standards developed by the Portable Applications Standards Committee (PASC) of the IEEE Computer Society
- ❑ Main subject areas:
 - System Interfaces (C, Fortran, ADA Bindings)
 - Commands & Utilities
 - Test Methods

POSIX Deliverables

- ❑ *POSIX.1 - 1990 System Interfaces (C language binding)*
- ❑ POSIX.2 - 1992 Shell & Utilities
- ❑ POSIX.5 - 1992 Ada bindings
- ❑ POSIX.1b - 1993 Realtime Extension
- ❑ POSIX.1c - 1995 Threads Extension
- ❑ POSIX.5b - 1996 Ada Realtime Extension
- ❑ POSIX 1003.13 - 1998 Realtime Profiles

POSIX *Latest* Deliverable

- *POSIX.1-2001*
 - *Approved December 6th 2001*
- Developed by the *Austin Group (see later)*
- Supersedes POSIX.1-1990 and its amendments and POSIX.2-1992 and its Amendments

The Success of POSIX

- ❑ The majority of UNIX system suppliers today support POSIX
- ❑ A core component of Linux
- ❑ Many other systems also now support POSIX, for example VMS, MVS, MPE and even Microsoft Windows NT
- ❑ It is now so successful that it has lost its role as a *market differentiator*

POSIX *building blocks*

- ❑ POSIX provides the foundations and building blocks for Open Systems
- ❑ Not all systems support all POSIX functionality
- ❑ Neither is POSIX functionally complete

POSIX.1 and Linux

- ❑ Linux mostly compatible at the source level with IEEE POSIX 1003.1 - 1990
- ❑ Early versions were certified for FIPS 151 - 2 compliance
- ❑ Test suites are available to measure compliance:
 - <http://www.opengroup.org/testing/downloads.html>

POSIX.1c and Linux

- ❑ Linux has a partial *pthread*s implementation
 - Most of the APIs are present
 - Some semantic differences
 - NGPT Threads implementation (see next slide) shows a solution possible but not yet accepted
- ❑ The *gLSB* includes a section giving advice to the programmer on how to work around the differences.

Next Generation POSIX Threads (NGPT)

- ❑ Userlevel library providing the POSIX threads API
- ❑ Uses a modified clone() interface
- ❑ Integrates into glibc as a LinuxThreads replacement
- ❑ NGPT 1.0 released June 2001
- ❑ NGPT 1.1.2 due to be released Jan 25 2002
- ❑ *A solution for some markets, but implementation concerns by some mean not yet accepted into mainstream Linux*

POSIX.2 and Linux

- ❑ Recent distributions are getting closer to POSIX.2 and provide most of the required utilities
- ❑ Usually differences hidden by `POSIXLY_CORRECT` environment variable
- ❑ Known differences: `split`, `du`, `df`
- ❑ The *gLSB* includes a set of man pages for the commands and utilities listing the differences when applicable

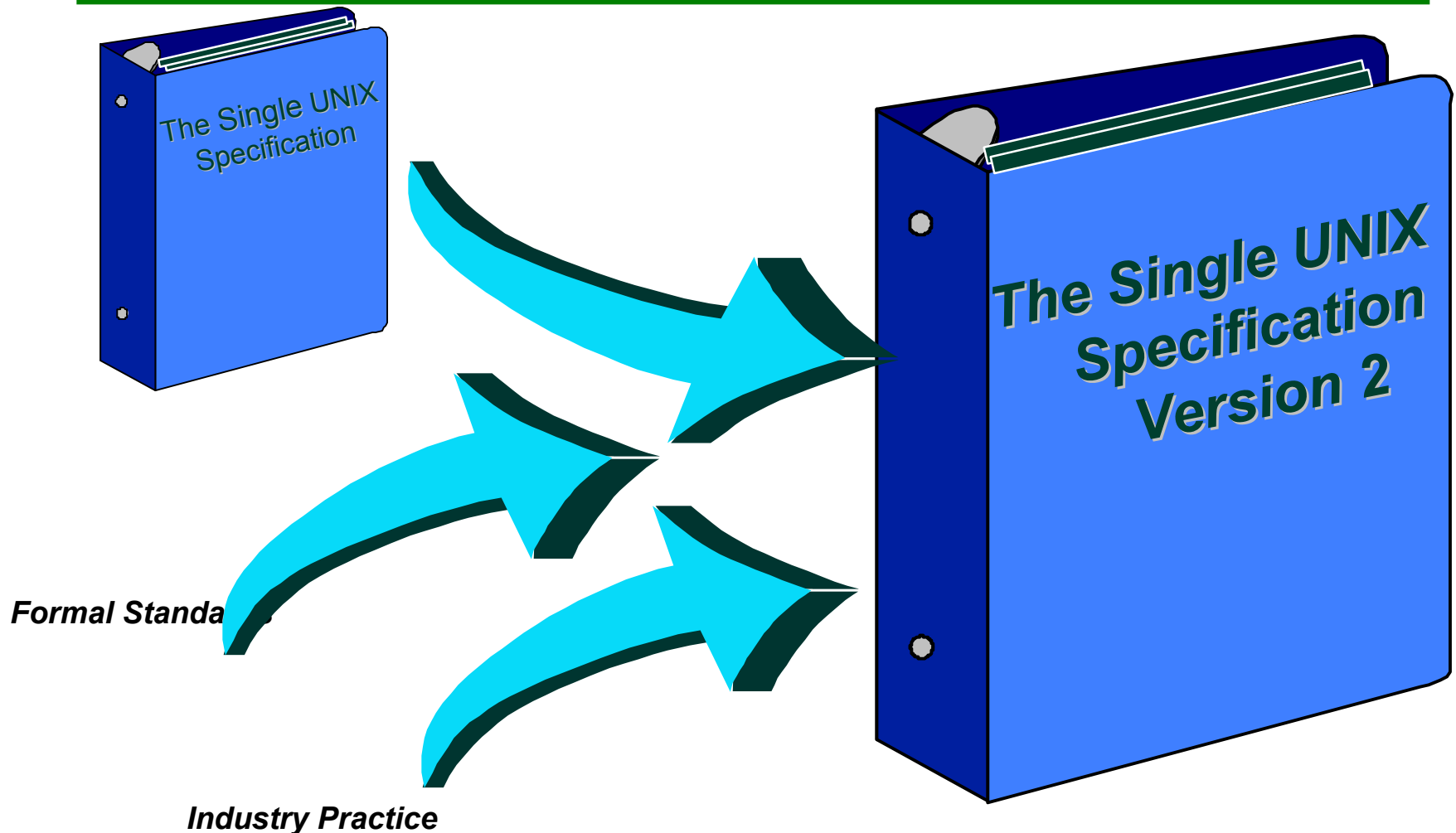
Agenda

- **Standards Projects**
 - **POSIX**
 - **The Single UNIX Specification**
 - **The Linux Standards Base**

The Single UNIX Specification

- ❑ Designed to give software developers a single set of APIs to be supported by every UNIX system
- ❑ Shifts the focus from incompatible UNIX system product implementations to compliance to a single set of APIs
- ❑ If an OS meets the specification and commonly available applications run on it then it is open.

The Single UNIX Specification Version 2



30-01-2002

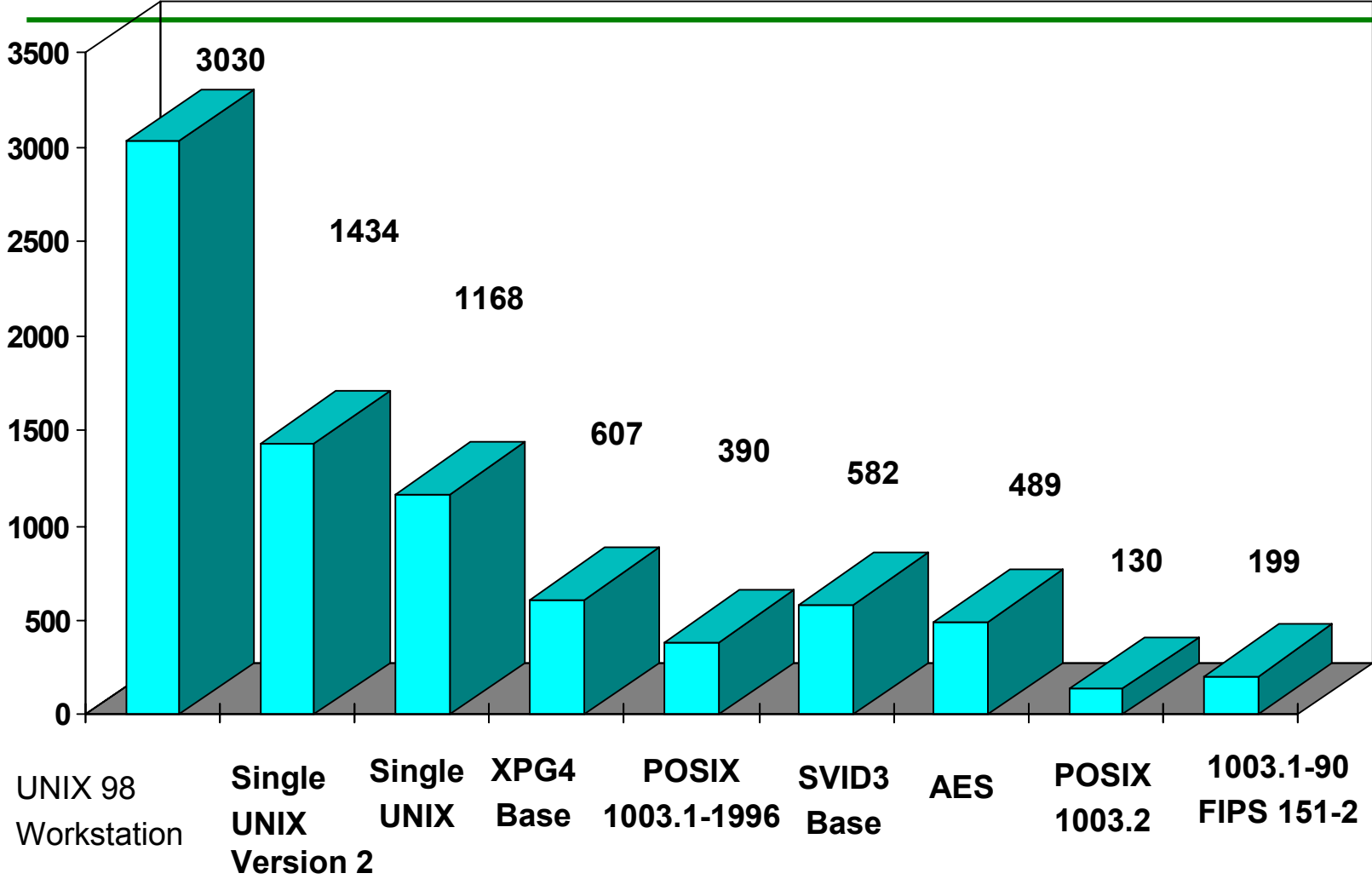
POSIX and UNIX Synergy

- The Single UNIX Specification builds on the foundation of international standards:
 - mandates POSIX options to form a rich foundation which you can depend on
 - ISO/IEC 9945-2:1992 (Shell & Utilities)
 - ISO/IEC 9899:1990/Amendment 1:1995 (Multibyte support)

Linux Status

- Recent versions of glibc have been getting closer to full coverage of the *Single UNIX Specification Version 2*
- Closer still to the *Single UNIX Specification Version 3*

Interface Counts



30-01-2002

Linux Status (Cont'd)

- Some features not implemented:
 - STREAMS (isastream, getmsg, putmsg, etc.)
 - XTI (t_alloc, etc)

Linux Status

- Some features partially implemented :
 - Wide characters (complete support in glibc 2.2 and later)
 - Pthreads

Linux Utilities

- ❑ Some differences
- ❑ Mainly due to the standards being aligned with System V historical definitions
 - notably the shell (sh)
 - Some efforts made with *bash* to align with POSIX as a result of the LSB test efforts (see the *VSC-lite* test suite which contains tests for the utilities and shell)

Linux Utilities (Cont'd)

- In general, where Linux provides an alternative functionality to that in the Single UNIX specification, the standard version is not provided
 - Compress/uncompress vs gzip/gunzip
 - SCCS utilities vs RCS
- As per *POSIX.2*; see the *gLSB specification* for details of the differences

The Single UNIX Specification Version 3

- ❑ Developed by the Austin Common Standards Revision Group
- ❑ An open industry initiative to revise the core *POSIX standard* and the *Single UNIX Specification*; standards that lie at the heart of the Linux operating system

The *Austin Group*

- ❑ Electronic participation
- ❑ Participation in the group is free
- ❑ Deliverables:
 - IEEE Std 1003.1-2001 (POSIX.1)
 - The Open Group Base Specifications Issue 6
 - (they are the same document!)

About the *Austin Group*

- Over 700 participants on the mailing list
- Wide industry support:
 - *AT&T, Compaq, Fujitsu, HP, IBM, Lucent, Microsoft, Red Hat, SGI, Siemens, Sun*
 - *DoD, USENIX, Canada revenue*
- Participation in the *Austin Group* from the open source community includes:
 - *The Linux Standard Base, NetBSD, FreeBSD, and many others*

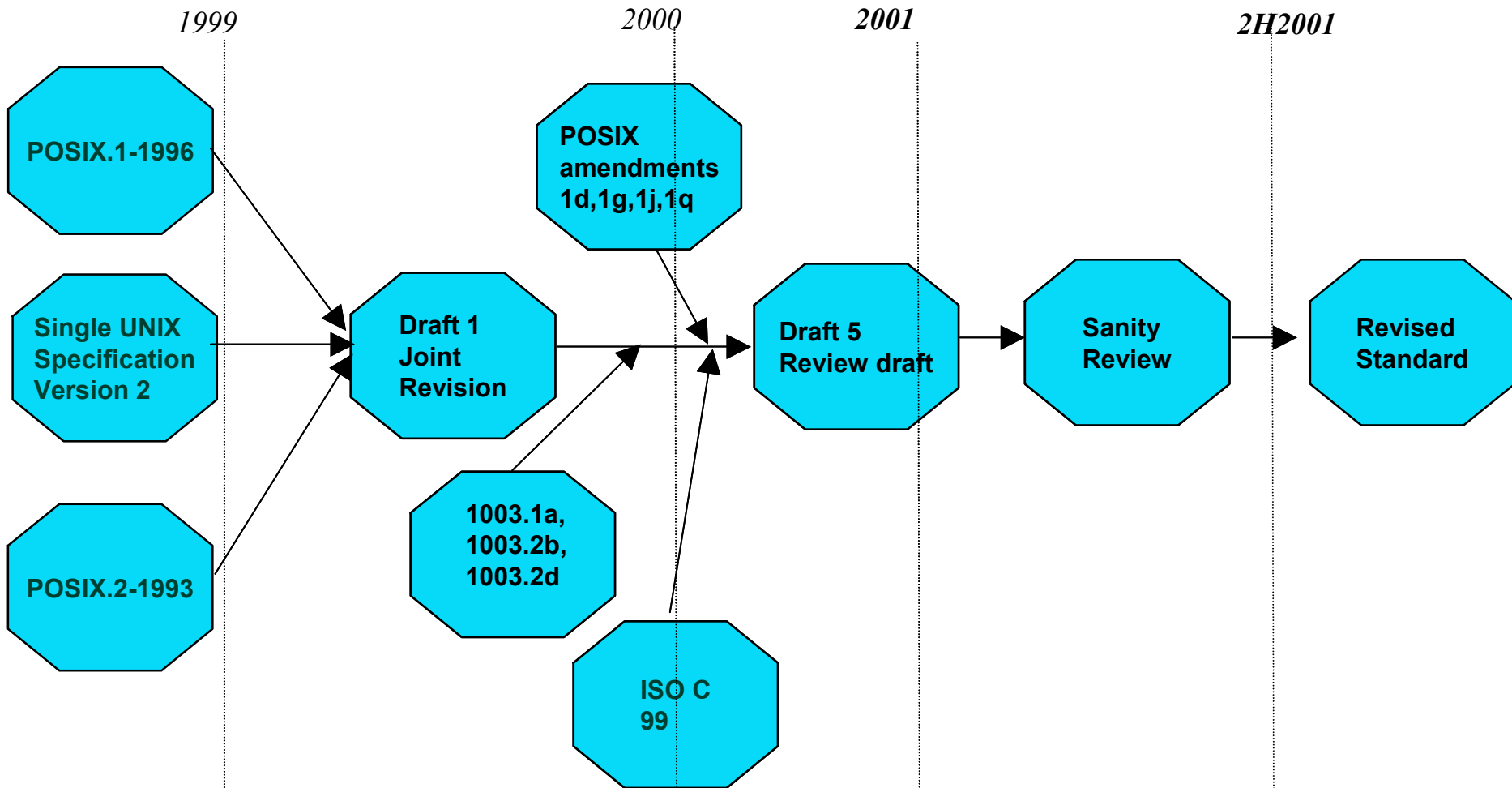
Objectives

- ❑ To target the joint specification at the programmer / user rather than the system implementer
- ❑ Organization based on the Core volumes of the Single UNIX Specification, organized alphabetically, and including Rationale
- ❑ To Produce a standard on schedule

Scope of the revision

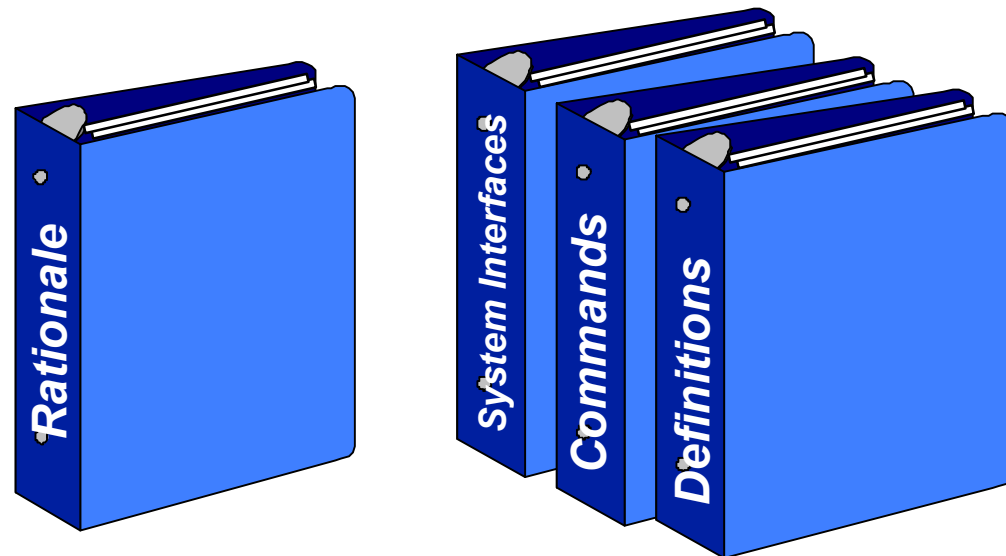
- ❑ Production of a single document to be adopted by multiple parties
- ❑ Minimize the number of changes required to implementations of earlier versions of the Base documents for the revision
- ❑ Limit new work items to those related to integration and consistency, resolving any conflicts
- ❑ Alignment with the ISO C 1999 standard

Roadmap



30-01-2002

The New Common Specification

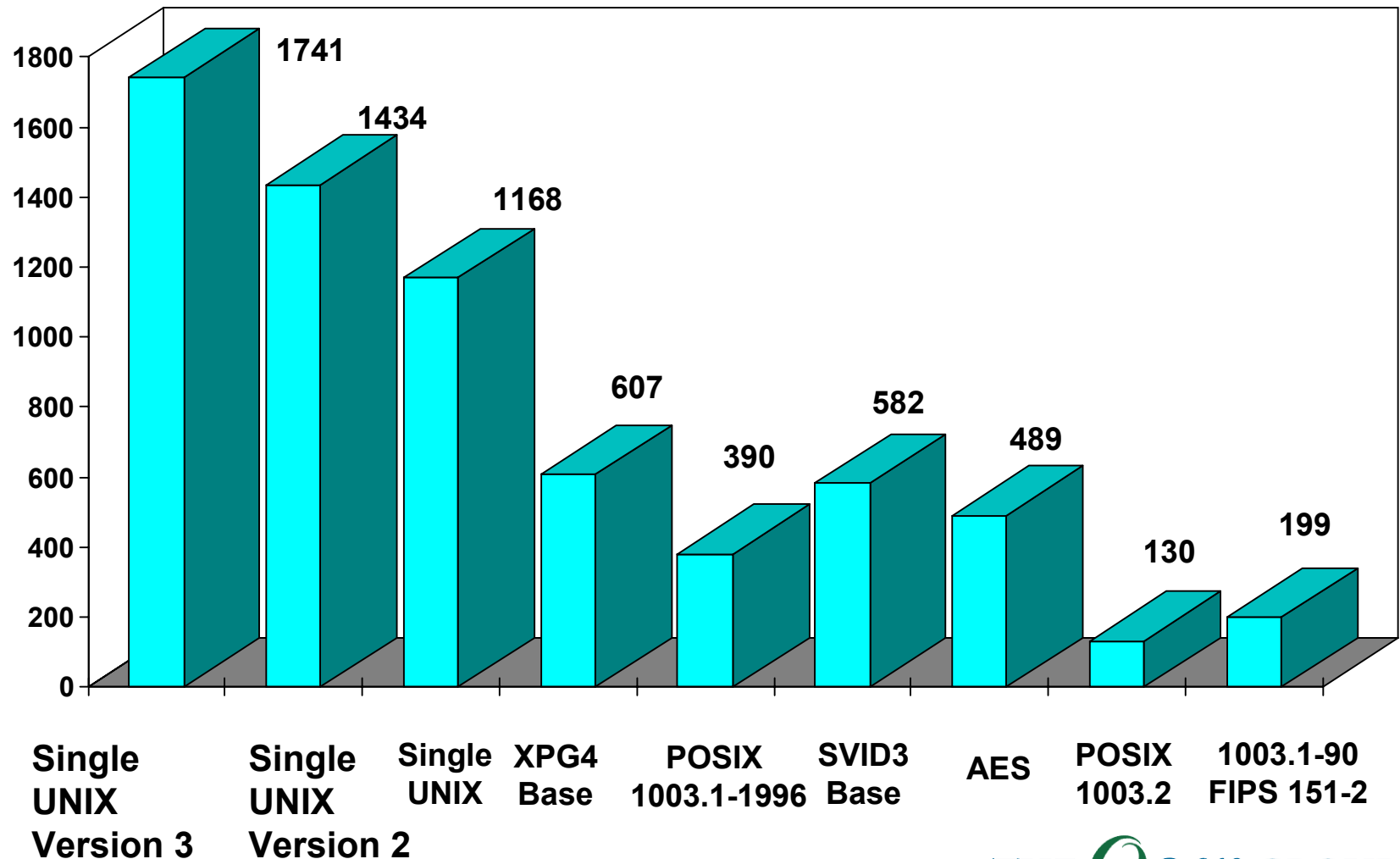


IEEE Std 1003.1-2001,
The Open Group Base
Specifications Issue 6

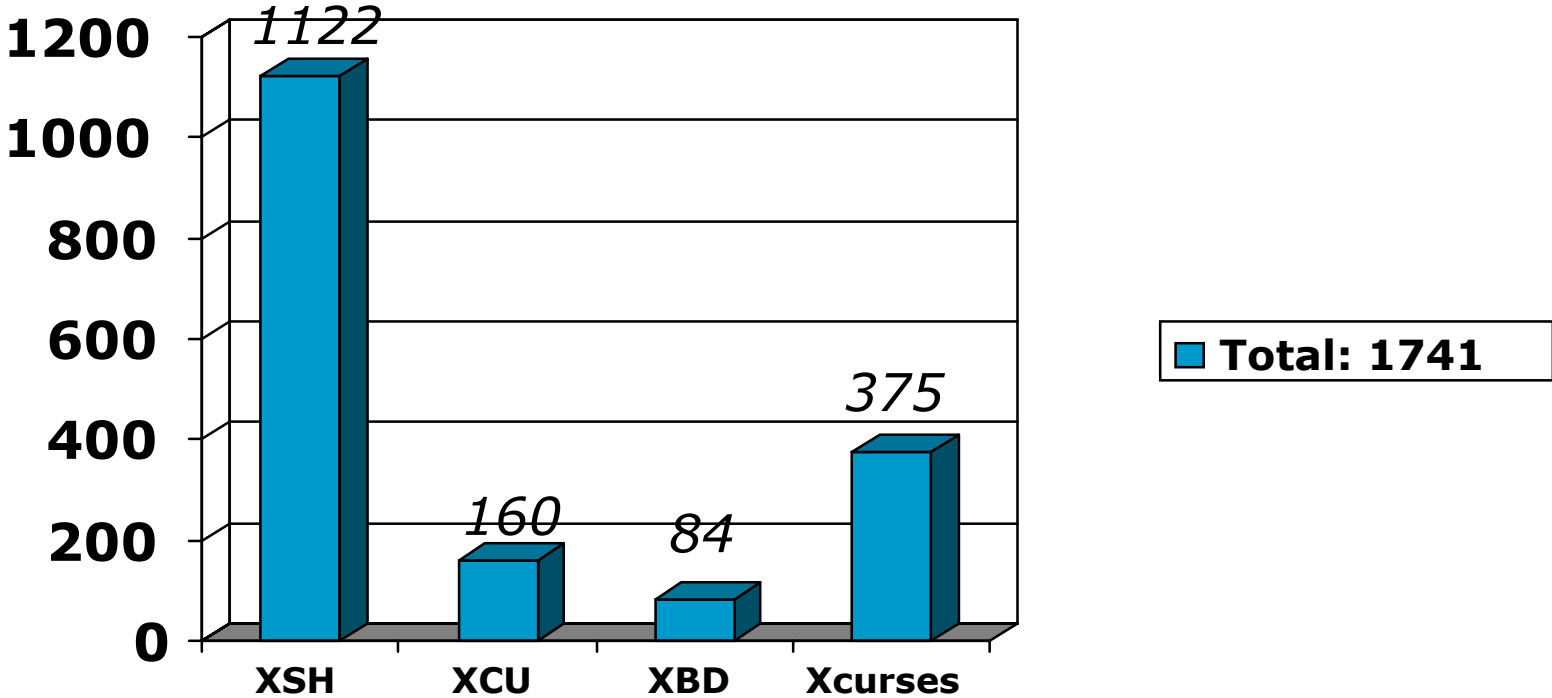
Approvals Status

- ❑ The Open Group September 12th 2001
- ❑ IEEE December 6th 2001
- ❑ Expected ISO/IEC Approval in 1H2002

Portability Functions



Interface Count



Formal Standards Alignment

- IEEE Std 1003.1-2001 (POSIX.1)
 - The Base Specifications Issue 6 is technically identical to POSIX.1, they are one and the same document
- ISO/IEC 9899:1999, Programming Languages – C (ISO C)

Options

- ❑ Encryption
- ❑ Realtime
- ❑ Realtime Threads
- ❑ Advanced Realtime
- ❑ Advanced Realtime Threads
- ❑ Tracing
- ❑ XSI STREAMS
- ❑ Legacy

Feature Test Macros

- `_XOPEN_SOURCE=600`
 - Used by applications to request the implementation make available the symbols and prototypes from the Single UNIX Spec V3
 - Subsumes the definition of the POSIX macro `_POSIX_C_SOURCE=200112L`

Compiling an Application

```
c99 -D_XOPEN_SOURCE=600 myapp.c \  
-o myapp -l c
```

```
c99 -D_XOPEN_SOURCE=600 myrtapp.c \  
-o myrtapp -l c -l rt
```

```
c99 -D_XOPEN_SOURCE=600 myrtthreadsapp.c \  
-o myrthreadsapp -l rt -l pthread
```

New Features

1003.1d

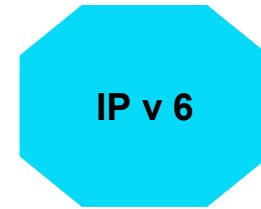
- ❑ Process creation via *posix_spawn()*
- ❑ *Sporadic Server Scheduling policy*
- ❑ *Execution time monitoring*
- ❑ *Time-outs for selected blocking functions*

New Features

- Enhanced threads functions:
 - barriers
 - spin locks
- Additional realtime functionality:
 - monotonic clock
 - synchronized clock

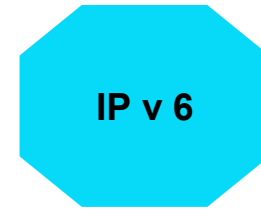
New Features

- ❑ 1003.1q , Tracing
- ❑ Provides tools to access and manage a stream of event data :
 - Traced process can record a trace event
 - Controller process can manage a trace stream
 - Analyzer process can retrieve traced events



New Features

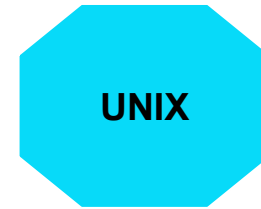
- IP version 6
 - Overcomes the shortage of address space
 - Designed for better manageability
 - *Security enhancements*
 - ¥ IPSEC
 - *Quality of service*



New Features

□ API Enhancements

- new functions `if_freenameindex()`, `if_indextoname()`, `if_nameindex()` and `if_nametoindex()` have been added to the Sockets Interfaces.
- New functions `inet_ntop()` and `inet_pton()` have been added to the IP Address Resolution Interfaces.



New Features

- ❑ The UNIX feature set is now available as an option within POSIX
- ❑ The new document set forms the core of the Single UNIX Specification Version 3
- ❑ Extended pthreads functions

New Features

- ISO/IEC 9899:1999 ISO C (c99)
 - *Language changes*
 - *New keywords: inline, restrict, _Bool, _Complex, _Imaginary, long long*
 - *Complex number and complex maths*
 - *Floating point environment support*
 - *Type generic math*
 - *Other library changes*

Key Changes

- ❑ Legacy and obsolescent features dropped
 - Much of this was to compromise between System V and BSD
- ❑ Job control and certain options in POSIX.1 mandated (FIPS 151-2 alignment)
 - ditto
- ❑ Corrigenda incorporated

Key Changes (Cont'd)

- ❑ XTI dropped
 - Is retained a separate specification but not required for POSIX or UNIX conformance
- ❑ STREAMS optional
 - An optional feature group for those systems that wish to support the functionality

Where to Obtain the Specification?

- ❑ The html version is online at
 - <http://www.UNIX-systems.org/version3>
- ❑ PDF either electronically or on CDROM can be ordered from <http://www.opengroup.org/pubs/catalog/un.htm>
 - The PDF is free to members of The Open Group
- ❑ Available also on CDROM with *The Authorized Guide to the Single UNIX Specification Version 3* (coming soon!)

Linux Status

- Latest versions of *glibc* have implemented much of the Austin Group specification
 - C99 Alignment
 - Some differences in the I18N behavior
 - See <http://gcc.gnu.org/c99status.html> for the latest status
 - Concentrated on the XSI mandatory parts of the XSH document
 - Remaining differences due to underlying kernel

Agenda

- **Standards Projects**
 - **POSIX**
 - **The Single UNIX Specification**
 - **The Linux Standards Base**

Why the LSB is important

- ❑ Linux is *the* binary standard for the Intel processor architecture
- ❑ The LSB has the promise to deliver shrink-wrapped commodity binary applications

LSB Overview

- ❑ Mission & Goals
- ❑ Why is the LSB needed?
- ❑ Organization
- ❑ What LSB does NOT attempt to do & why
- ❑ Architectural Overview - How will it work?
- ❑ Current LSB Roadmap

Mission

- ❑ The LSB develops and promotes a set of standards that increases compatibility among Linux distributions and enables software applications to run on any compliant Linux system.
- ❑ In addition, the LSB helps coordinate efforts to recruit software vendors to port and write products for Linux.

Goals

- ❑ To preserve backwards compatibility without locking out future progress
- ❑ To allow distributions to still be unique (and provide added value) by only standardizing the base
- ❑ To avoid fragmentation of the base functionality

Why is the LSB Needed?

- ❑ Allows ISVs to:
 - Minimize issues in porting code from another Linux platform
 - Allows a package to perform the same way regardless of the Linux distribution or emulation of such

Organization

- Steering Committee Chair - George Kraft IV
 - FSG liaisons -Scott McNeil, Dan Quinlan
 - Technical Sub-Committees
 - Written Spec - Technical Lead - Stuart Anderson
 - Test Suites - Technical Lead - Andrew Josey
 - Sample Implementation - Technical Lead - Ralf Flaxa
 - LSB Futures – Technical Lead – Matt Taggart
 - Many volunteers

What is not covered?

- ❑ It does not mean there will only be one "Linux"
 - One port - multiple platform choices
- ❑ It does not specify the kernel level
 - The kernel can be any version that provides conforming interfaces
- ❑ It does not cover languages other than C
 - Other languages will be covered over time

What is not covered?

- ❑ It does not specify a desktop environment
 - Desktop integration is a work in progress
 - Use desktop independent toolkits (Qt, Gtk)
- ❑ It does not fully cover system administration
 - Settled areas are specified - Others will be added

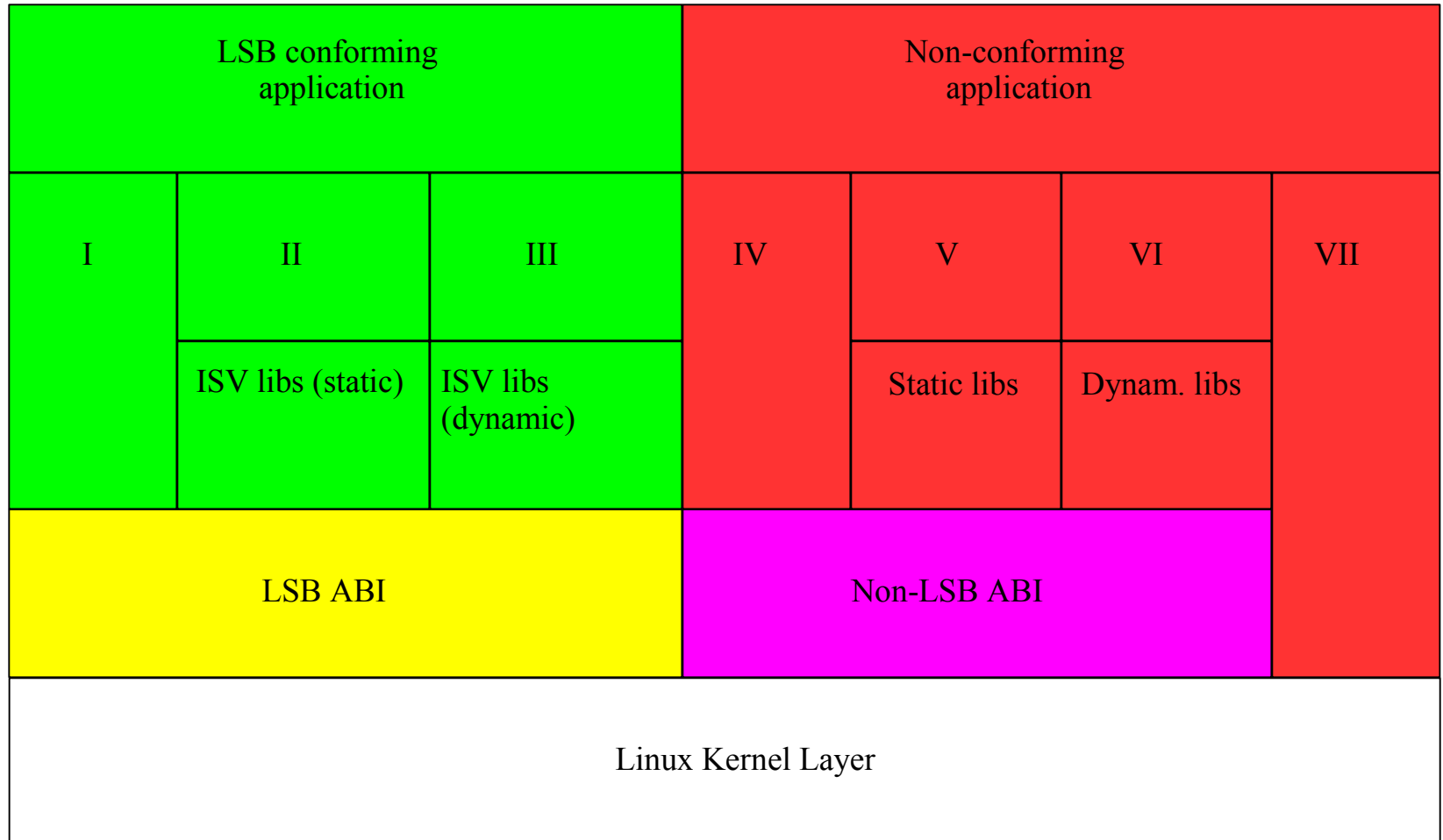
LSB Specification

- A *generic, processor-independent* specification
 - The Generic LSB Specification known as the *gLSB*
- One or more *processor-specific* supplemental specifications that build upon the *gLSB* to provide a complete binary specification
 - Intel x86, Intel ia64

LSB Specification

- ❑ Object file formats
- ❑ Dynamic linking
- ❑ Base libraries
- ❑ Utility libraries
- ❑ Graphics Libraries
- ❑ Packaging
- ❑ Commands and Utilities
- ❑ Standard Shell
- ❑ Users and Groups
- ❑ Filesystem Hierarchy
- ❑ System Initialization

Architectural Overview



Standards Alignment

- Built on industry standards
 - IEEE Std POSIX 1003.1-1996 (POSIX.1)
 - IEEE Std POSIX 1003.2-1992 (POSIX.2)
 - The Open Group Single UNIX Specification Version 2 (*aka* UNIX 98)
 - AT&T (Caldera/SCO) SVID, Issue 3 1989-2001 (SVID.3)
 - See www.linuxbase.org/spec/gLSB/gLSB/rstandards.html
 - ***36 Specifications referenced at this time***

Current Status

- ❑ Written 1.0 *gLSB* Spec Released June 2001
- ❑ Pilot Beta test program August 2001
- ❑ *gLSB* 1.1 specification December 31 2001
- ❑ *psLSB* for Intel x86 December 31 2001
- ❑ *psLSB* for Intel ia64 expected end 1Q2002
- ❑ *gLSB* 1.2 4Q2002?

gLSB 1.1 vs 1.0

- ❑ Designed to be a superset
- ❑ Some exceptions
 - Bug fixes
 - Additions of functions identified as needed by applications
 - Improved data definitions
 - Further changes needed to support *psLSBs*

Current Status (Cont'd)

- ❑ Test Suite 1.0 *golden* release December 3rd 2001
 - LSB-FHS
 - VSX-PCTS
 - LSB-OS
 - LSB-USRGROUPS
- ❑ Binary Test Suite release December 31st 2001
- ❑ Certification Pilot launch January 2002
- ❑ Certification program to launch 2Q2002

Application Tools

- *lsbappchk*
 - Used during the build process, similar to *gcc*
 - Uses same flags, but instead of producing compiled code it checks that the binary file(s) being produced comply with the LSB spec
- FHS Checklist
 - 6 basic questions about the applications directory usage
 - Allowance for deviation when rationale justifies it

LSB Resources

- ❑ LSB main site - www.linuxbase.org
- ❑ Free Standards Group - www.freestandards.org/
- ❑ The LSB Specification - www.linuxbase.org/
- ❑ LSB Test
 - www.linuxbase.org/test/
- ❑ LSB Sample Implementation
 - www.linuxbase.org/impl/
- ❑ LSB Futures
 - <http://www.linuxbase.org/futures/>

Summary

- Standards activities
 - alive and ongoing
 - free to participate in
- Significant cross-fertilization of projects
- Significant opportunities to grow the market and prevent fragmentation

Further Information

- *The Linux Standards Base*
 - <http://www.linuxbase.org>
- *The Austin Group*
 - <http://www.opengroup.org/austin>
- *The Single UNIX Specification*
 - <http://www.UNIX-systems.org>
- *These slides*
 - <http://www.UNIX-systems.org/slides.html>

How You Can Help?

- ❑ To participate in the *Austin Group*, join the mailing list by visiting the web site at:
 - <http://www.opengroup.org/austin/>
- ❑ To participate in the Linux Standards Base visit:
 - <http://www.linuxbase.org/>
- ❑ To join the Free Standards Group, visit:
 - <http://www.freestandards.org/>



Standards: Bridging the Generation Gap between Linux® and UNIX® Systems

Andrew Josey
Director of Certification
The Open Group
Email: a.josey@opengroup.org

UNIX is a registered trademark of The Open Group
Linux is a registered trademark of Linus Torvalds
30-01-2002