
**Linux Standard Base (LSB) core
specification 3.1 —**

**Part 4:
Specification for AMD64 architecture**

*Spécifications 3.1 relatives au noyau de base normalisé Linux (LSB) —
Partie 4: Spécifications pour l'architecture AMD64*

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

This document is a preview generated by EVS

© ISO/IEC 2006

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Linux Standard Base Core Specification for AMD64 3.1

ISO/IEC 23360-4:2006(E)

Copyright © 2006 ISO/IEC

This standard includes material that has been provided by the Free Standards Group under the GNU Free Documentation License Version 1.1 published by the Free Software Foundation.

Portions of the text are copyrighted by the following parties:

- The Regents of the University of California
- Free Software Foundation
- Ian F. Darwin
- Paul Vixie
- BSDI (now Wind River)
- Andrew G Morgan
- Jean-loup Gailly and Mark Adler
- Massachusetts Institute of Technology

These excerpts are being used in accordance with their respective licenses.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

UNIX is a registered trademark of The Open Group.

LSB is a trademark of the Free Standards Group in the United States and other countries.

AMD is a trademark of Advanced Micro Devices, Inc.

Intel and Itanium are registered trademarks and Intel386 is a trademark of Intel Corporation.

PowerPC is a registered trademark and PowerPC Architecture is a trademark of the IBM Corporation.

S/390 is a registered trademark of the IBM Corporation.

OpenGL is a registered trademark of Silicon Graphics, Inc.

Contents

Foreword	vii
Introduction	viii
I Introductory Elements	0
1 Scope.....	1
1.1 General.....	1
1.2 Module Specific Scope.....	1
2 References	2
2.1 Normative References	2
2.2 Informative References/Bibliography	4
3 Requirements	7
3.1 Relevant Libraries	7
3.2 LSB Implementation Conformance	7
3.3 LSB Application Conformance.....	8
4 Definitions	10
5 Terminology	11
6 Documentation Conventions	13
II Executable and Linking Format (ELF).....	14
7 Introduction.....	15
8 Low Level System Information.....	16
8.1 Machine Interface.....	16
8.2 Function Calling Sequence.....	17
8.3 Operating System Interface	18
8.4 Process Initialization.....	18
8.5 Coding Examples	18
8.6 C Stack Frame	19
8.7 Debug Information.....	19
9 Object Format	20
9.1 Introduction	20
9.2 ELF Header	20
9.3 Sections	20
9.4 Symbol Table	21
9.5 Relocation.....	21
10 Program Loading and Dynamic Linking	22
10.1 Introduction	22
10.2 Program Header	22
10.3 Program Loading	22
10.4 Dynamic Linking.....	22
III Base Libraries	24
11 Libraries	25
11.1 Program Interpreter/Dynamic Linker	25
11.2 Interfaces for libc	25
11.3 Data Definitions for libc	40
11.4 Interfaces for libm	52
11.5 Data Definitions for libm.....	57
11.6 Interface Definitions for libm	58
11.7 Interfaces for libpthread	59
11.8 Data Definitions for libpthread	62
11.9 Interfaces for libgcc_s	62
11.10 Data Definitions for libgcc_s.....	63
11.11 Interface Definitions for libgcc_s.....	64
11.12 Interfaces for libdl	69
11.13 Data Definitions for libdl	70
11.14 Interfaces for libcrypt.....	70

IV Utility Libraries.....	71
12 Libraries	72
12.1 Interfaces for libz.....	72
12.2 Data Definitions for libz.....	72
12.3 Interfaces for libncurses.....	72
12.4 Data Definitions for libncurses.....	73
12.5 Interfaces for libutil.....	73
V Package Format and Installation.....	75
13 Software Installation	76
13.1 Package Dependencies	76
13.2 Package Architecture Considerations	76
A Alphabetical Listing of Interfaces.....	77
A.1 libgcc_s.....	77
A.2 libm.....	77

This document is a preview generated by EVS

List of Tables

2-1 Normative References	2
2-2 Other References	4
3-1 Standard Library Names.....	7
8-1 Non Conforming Instructions.....	16
9-1 ELF Special Sections	20
9-2 Additional Special Sections	21
11-1 libc Definition	25
11-2 libc - RPC Function Interfaces	25
11-3 libc - System Calls Function Interfaces	26
11-4 libc - Standard I/O Function Interfaces	28
11-5 libc - Standard I/O Data Interfaces	29
11-6 libc - Signal Handling Function Interfaces	29
11-7 libc - Signal Handling Data Interfaces	30
11-8 libc - Localization Functions Function Interfaces	30
11-9 libc - Localization Functions Data Interfaces	31
11-10 libc - Socket Interface Function Interfaces	31
11-11 libc - Wide Characters Function Interfaces.....	32
11-12 libc - String Functions Function Interfaces	33
11-13 libc - IPC Functions Function Interfaces	34
11-14 libc - Regular Expressions Function Interfaces	34
11-15 libc - Character Type Functions Function Interfaces.....	34
11-16 libc - Time Manipulation Function Interfaces	35
11-17 libc - Time Manipulation Data Interfaces	35
11-18 libc - Terminal Interface Functions Function Interfaces	36
11-19 libc - System Database Interface Function Interfaces.....	36
11-20 libc - Language Support Function Interfaces	37
11-21 libc - Large File Support Function Interfaces	37
11-22 libc - Standard Library Function Interfaces.....	38
11-23 libc - Standard Library Data Interfaces.....	40
11-24 libm Definition	52
11-25 libm - Math Function Interfaces.....	53
11-26 libm - Math Data Interfaces	57
11-27 libpthread Definition.....	59
11-28 libpthread - Realtime Threads Function Interfaces.....	59
11-29 libpthread - Posix Threads Function Interfaces	60
11-30 libpthread - Thread aware versions of libc interfaces Function Interfaces	61
11-31 libgcc_s Definition	62
11-32 libgcc_s - Unwind Library Function Interfaces.....	63
11-33 libdl Definition	69
11-34 libdl - Dynamic Loader Function Interfaces.....	69
11-35 libcrypt Definition.....	70
11-36 libcrypt - Encryption Function Interfaces	70
12-1 libz Definition.....	72
12-2 libncurses Definition	73
12-3 libutil Definition.....	73
12-4 libutil - Utility Functions Function Interfaces	74
A-1 libgcc_s Function Interfaces	77
A-2 libm Function Interfaces	77

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 23360-4 was prepared by the Free Standards Group and was adopted, under the PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*.

ISO/IEC 23360 consists of the following parts, under the general title *Linux Standard Base (LSB) core specification 3.1*:

- *Part 1: Generic specification*
- *Part 2: Specification for IA32 architecture*
- *Part 3: Specification for IA64 architecture*
- *Part 4: Specification for AMD64 architecture*
- *Part 5: Specification for PPC32 architecture*
- *Part 6: Specification for PPC64 architecture*
- *Part 7: Specification for S390 architecture*
- *Part 8: Specification for S390X architecture*

Introduction

The LSB defines a binary interface for application programs that are compiled and packaged for LSB-conforming implementations on many different hardware architectures. Since a binary specification includes information specific to the computer processor architecture for which it is intended, it is not possible for a single document to specify the interface for all possible LSB-conforming implementations. Therefore, the LSB is a family of specifications, rather than a single one.

This document should be used in conjunction with the documents it references. This document enumerates the system components it includes, but descriptions of those components may be included entirely or partly in this document, partly in other documents, or entirely in other reference documents. For example, the section that describes system service routines includes a list of the system routines supported in this interface, formal declarations of the data structures they use that are visible to applications, and a pointer to the underlying referenced specification for information about the syntax and semantics of each call. Only those routines not described in standards referenced by this document, or extensions to those standards, are described in detail. Information referenced in this way is as much a part of this document as is the information explicitly included here.

The specification carries a version number of either the form $x.y$ or $x.y.z$. This version number carries the following meaning:

- The first number (x) is the major version number. All versions with the same major version number should share binary compatibility. Any addition or deletion of a new library results in a new version number. Interfaces marked as deprecated may be removed from the specification at a major version change.
- The second number (y) is the minor version number. Individual interfaces may be added if all certified implementations already had that (previously undocumented) interface. Interfaces may be marked as deprecated at a minor version change. Other minor changes may be permitted at the discretion of the LSB workgroup.
- The third number (z), if present, is the editorial level. Only editorial changes should be included in such versions.

Since this specification is a descriptive Application Binary Interface, and not a source level API specification, it is not possible to make a guarantee of 100% backward compatibility between major releases. However, it is the intent that those parts of the binary interface that are visible in the source level API will remain backward compatible from version to version, except where a feature marked as deprecated in one release may be removed from a future release.

Implementors are strongly encouraged to make use of symbol versioning to permit simultaneous support of applications conforming to different releases of this specification.

This is version 3.1 of the Linux Standard Base Core Specification. This specification is part of a family of specifications under the general title "Linux Standard Base (LSB) core specification 3.1". Developers of applications or implementations interested in using the LSB trademark should see the Free Standards Group Certification Policy for details.

I Introductory Elements

This document is a preview generated by EVS

This document is a preview generated by EVS

Linux Standard Base (LSB) core specification 3.1 —

Part 4: Specification for AMD64 architecture

1 Scope

1.1 General

The Linux Standard Base (LSB) defines a system interface for compiled applications and a minimal environment for support of installation scripts. Its purpose is to enable a uniform industry standard environment for high-volume applications conforming to the LSB.

These specifications are composed of two basic parts: A common specification ("LSB-generic" or "generic LSB"), ISO/IEC 23360-1, describing those parts of the interface that remain constant across all implementations of the LSB, and an architecture-specific part ("LSB-arch" or "archLSB") describing the parts of the interface that vary by processor architecture. Together, the LSB-generic and the relevant architecture-specific part of ISO/IEC 23360 for a single hardware architecture provide a complete interface specification for compiled application programs on systems that share a common hardware architecture.

ISO/IEC 23360-1, the LSB-generic document, is to be used in conjunction with an architecture-specific part. Whenever a section of the LSB-generic specification is supplemented by architecture-specific information, the LSB-generic document includes a reference to the architecture part. Architecture-specific parts of ISO/IEC 23360 may also contain additional information that is not referenced in the LSB-generic document.

The LSB contains both a set of Application Program Interfaces (APIs) and Application Binary Interfaces (ABIs). APIs may appear in the source code of portable applications, while the compiled binary of that application may use the larger set of ABIs. A conforming implementation provides all of the ABIs listed here. The compilation system may replace (e.g. by macro definition) certain APIs with calls to one or more of the underlying binary interfaces, and may insert calls to binary interfaces as needed.

The LSB is primarily a binary interface definition. Not all of the source level APIs available to applications may be contained in this specification.

1.2 Module Specific Scope

This is the AMD64 architecture-specific Core part of the Linux Standard Base (LSB). It supplements the generic LSB Core module with those interfaces that differ between architectures.

Interfaces described in this part of ISO/IEC 23360 are mandatory except where explicitly listed otherwise. Core interfaces may be supplemented by other modules; all modules are built upon the core.

2 References

2.1 Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

Note: Where copies of a document are available on the World Wide Web, a Uniform Resource Locator (URL) is given for informative purposes only. This may point to a more recent copy of the referenced specification, or may be out of date. Reference copies of specifications at the revision level indicated may be found at the Free Standards Group's Reference Specifications (<http://refspecs.freestands.org>) site.

Table 2-1 Normative References

Name	Title	URL
ISO/IEC 23360-1	ISO/IEC 23360-1:2006, <i>Linux Standard Base (LSB)</i> <i>core specification 3.1—</i> <i>Part 1: Generic</i> <i>Specification</i>	http://www.linuxbase.org/spec/
AMD64 Architecture Programmer's Manual, Volume 1	AMD64 Architecture Programmer's Manual, Volume 1: Application Programming 24592 3.08	http://www.amd.com/us-en/Processors/DevelopWithAMD/
AMD64 Architecture Programmer's Manual, Volume 2	AMD64 Architecture Programmer's Manual, Volume 2: System Programming 24593 3.08	http://www.amd.com/us-en/Processors/DevelopWithAMD/
AMD64 Architecture Programmer's Manual, Volume 3	AMD64 Architecture Programmer's Manual, Volume 3: General Purpose and System Instructions 24594 3.03	http://www.amd.com/us-en/Processors/DevelopWithAMD/
AMD64 Architecture Programmer's Manual, Volume 4	AMD64 Architecture Programmer's Manual, Volume 4: 128-bit Media Instructions 26568 3.04	http://www.amd.com/us-en/Processors/DevelopWithAMD/
AMD64 Architecture Programmer's Manual, Volume 5	AMD64 Architecture Programmer's Manual, Volume 5: 64-bit Media and x87 Floating-Point Instructions 26569 3.03	http://www.amd.com/us-en/Processors/DevelopWithAMD/

Name	Title	URL
ISO C (1999)	ISO/IEC 9899: 1999, <i>Programming Languages — C</i>	
ISO POSIX (2003)	ISO/IEC 9945-1:2003, <i>Information technology — Portable Operating System Interface (POSIX) — Part 1: Base Definitions</i> ISO/IEC 9945-2:2003, <i>Information technology — Portable Operating System Interface (POSIX) — Part 2: System Interfaces</i> ISO/IEC 9945-3:2003, <i>Information technology — Portable Operating System Interface (POSIX) — Part 3: Shell and Utilities</i> ISO/IEC 9945-4:2003, <i>Information technology — Portable Operating System Interface (POSIX) — Part 4: Rationale</i>	http://www.unix.org/ version3/
Large File Support	Large File Support	http://www.UNIX-syst ems.org/version2/wha tsnew/lfs20mar.html
SUSv2	CAE Specification, January 1997, System Interfaces and Headers (XSH), Issue 5 (ISBN: 1-85912-181-0, C606)	http://www.opengrou p.org/publications/cat alog/un.htm
SVID Issue 3	American Telephone and Telegraph Company, System V Interface Definition, Issue 3; Morristown, NJ, UNIX Press, 1989. (ISBN 0201566524)	
SVID Issue 4	System V Interface Definition, Fourth Edition	

Name	Title	URL
System V ABI	System V Application Binary Interface, Edition 4.1	http://www.caldera.com/developers/devspecs/gabi41.pdf
System V ABI Update	System V Application Binary Interface - DRAFT - 17 December 2003	http://www.caldera.com/developers/gabi/2003-12-17/contents.html
System V Application Binary Interface AMD64 Architecture Processor Supplement	System V Application Binary Interface AMD64 Architecture Processor Supplement, Draft Version 0.95	http://www.x86-64.org/documentation/abi-0.95.pdf
X/Open Curses	CAE Specification, May 1996, X/Open Curses, Issue 4, Version 2 (ISBN: 1-85912-171-3, C610), plus Corrigendum U018	http://www.opengroup.org/publications/catalog/un.htm

2.2 Informative References/Bibliography

In addition, the specifications listed below provide essential background information to implementors of this specification. These references are included for information only.

Table 2-2 Other References

Name	Title	URL
DWARF Debugging Information Format, Revision 2.0.0	DWARF Debugging Information Format, Revision 2.0.0 (July 27, 1993)	http://refspecs.freestdards.org/dwarf/dwarf-2.0.0.pdf
DWARF Debugging Information Format, Revision 3.0.0 (Draft)	DWARF Debugging Information Format, Revision 3.0.0 (Draft)	http://refspecs.freestdards.org/dwarf/
IEC 60559/IEEE 754 Floating Point	IEC 60559:1989, <i>Binary floating-point arithmetic for microprocessor systems</i>	http://www.ieee.org/
ISO/IEC TR 14652	ISO/IEC TR 14652:2004, <i>Information technology — Specification method for cultural conventions</i>	

Name	Title	URL
ITU-T V.42	International Telecommunication Union Recommendation V.42 (2002): Error-correcting procedures for DCEs using asynchronous-to-synchronous convention ITUV	http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-V.42
LI18nux Globalization Specification	LI18NIX 2000 Globalization Specification, Version 1.0 with Amendment 4	http://www.li18nux.org/docs/html/LI18NIX-2000-amd4.htm
Linux Allocated Device Registry	LINUX ALLOCATED DEVICES	http://www.lanana.org/docs/device-list/devices.txt
PAM	Open Software Foundation, Request For Comments: 86.0, October 1995, V. Samar & R. Schemers (SunSoft)	http://www.opengroup.org/tech/rfc/mirror-rfc/rfc86.0.txt
RFC 1321: The MD5 Message-Digest Algorithm	IETF RFC 1321: The MD5 Message-Digest Algorithm	http://www.ietf.org/rfc/rfc1321.txt
RFC 1831/1832 RPC & XDR	IETF RFC 1831 & 1832	http://www.ietf.org/
RFC 1833: Binding Protocols for ONC RPC Version 2	IETF RFC 1833: Binding Protocols for ONC RPC Version 2	http://www.ietf.org/rfc/rfc1833.txt
RFC 1950: ZLIB Compressed Data Format Specification	IETF RFC 1950: ZLIB Compressed Data Format Specification	http://www.ietf.org/rfc/rfc1950.txt
RFC 1951: DEFLATE Compressed Data Format Specification	IETF RFC 1951: DEFLATE Compressed Data Format Specification version 1.3	http://www.ietf.org/rfc/rfc1951.txt
RFC 1952: GZIP File Format Specification	IETF RFC 1952: GZIP file format specification version 4.3	http://www.ietf.org/rfc/rfc1952.txt
RFC 2440: OpenPGP Message Format	IETF RFC 2440: OpenPGP Message Format	http://www.ietf.org/rfc/rfc2440.txt

Name	Title	URL
RFC 2821: Simple Mail Transfer Protocol	IETF RFC 2821: Simple Mail Transfer Protocol	http://www.ietf.org/rfc/rfc2821.txt
RFC 2822: Internet Message Format	IETF RFC 2822: Internet Message Format	http://www.ietf.org/rfc/rfc2822.txt
RFC 791: Internet Protocol	IETF RFC 791: Internet Protocol Specification	http://www.ietf.org/rfc/rfc791.txt
RPM Package Format	RPM Package Format V3.0	http://www.rpm.org/max-rpm/s1-rpm-file-format-rpm-file-format.html
SUSv2 Commands and Utilities	The Single UNIX Specification (SUS) Version 2, Commands and Utilities (XCU), Issue 5 (ISBN: 1-85912-191-8, C604)	http://www.opengroup.org/publications/catalog/un.htm
zlib Manual	zlib 1.2 Manual	http://www.gzip.org/zlib/