FOR IMMEDIATE RELEASE

FOR MORE INFORMATION

Sun Microsystems, Inc.
Carol Sacks (415) 336-0521

Hewlett-Packard Company
Diane Dennis (303) 229-2255

ROCK RIDGE GROUP SUBMITS PRELIMINARY CD-ROM SPECS TO NIST

Sixteen Companies Offer Their Support

SAN JOSE, Calif.— March 18, 1991 — Sixteen companies known as the Rock Ridge Group have submitted two new preliminary CD-ROM specifications -- System Use Sharing Protocol (SUSP) and the Rock Ridge Interchange Protocol Specification (RRIP) -- to the National Institute of Standards and Technology (NIST). These specifications offer industry-standard solutions for the distribution of data and software on CD-ROM media. They extend and are completely compliant with the current international standard format for CD-ROM, the ISO 9660-1988. CD-ROM media can significantly reduce cost and improve software installation and ease of use.

Members of the Rock Ridge Group include Anvil Software Limited, Digital Equipment Corporation, Fulcrum Technologies Inc., Hewlett-Packard Company, Highland Software, Interactive Systems Corporation, Lehman Brothers, Mentor Graphics Corporation, Meridian Data, Next Technology Corporation (UK), Philips Kommunikations Industrie, The Santa Cruz Operation, Silicon Graphics Computer Systems, Solbourne Computer, Sun Microsystems and Young Minds.

The goals of the Rock Ridge Group and its submission to NIST will be discussed by Andy Young, founder and president of Young Minds, on Wednesday, March 20, at the Sixth International Conference and Exposition on Multimedia and CD-ROM being held here.

The Rock Ridge CD-ROM specifications have been submitted to NIST for its review and consideration in developing a Federal Information Processing Standard (FIPS) for CD-ROM technology. NIST develops FIPS for use by United States federal agencies in the acquisition and management of computer and telecommunication systems.

"We are interested in developing standards-based solutions within an architectural framework that will give federal users faster access to data and software within open systems environments," said Allen Hankinson, chief of NIST's Systems and Software Technology Division. "The Rock Ridge specifications combined with the current ISO 9660 standard will deliver this to the user."

Rock Ridge Specifications

The SUSP extension to the ISO 9660 standard allows multiple file system extensions to coexist on one CD-ROM disc. The existing ISO standard allows for this, but does not define a mechanism.

The RRIP specification, which is built on top of the SUSP extension, lets POSIX files and directories be recorded on CD-ROM without requiring modifications to files, such as shortening file and directory names. POSIX is the Institute of Electrical and Electronic Engineers Portable Operating System Interface IEEE Std. 1003.1-1990. This recorded information can then be read by any computer system on which the new specifications or the current standard are implemented. RRIP makes it very easy for UNIX® applications to be executed directly from CD-ROM without copying them on the hard disk and using disk space. A standard CD-ROM holds more than 640 megabytes of data.

User Benefits

The Rock Ridge specifications primarily benefit software developers, software distributors and end users. Developers can now produce single CD-ROM discs that contain multiple applications or a single application that runs on systems from different vendors. In addition, developers can transfer their software to CD-ROM more easily because the Rock Ridge specifications eliminate the need to shorten file names.

With the Rock Ridge specifications, software distributors can now stock a single version of a product instead of multiple versions. End users save time because they now have direct access to data on their discs, eliminating the need to copy data to a hard disk first. In addition, users save money because they don't need to buy a hard disk on which to copy data.

Rock Ridge Information

Rock Ridge is a group of companies that began meeting in July 1990 to resolve difficulties in distributing software on CD-ROM. For companies interested in reviewing the specifications, please contact Robert J. Niland at Hewlett-Packard. He can be reached at (303) 229-4014 or at rjn@fc.hp.com. The Rock Ridge Group will hold its next meeting on April 16 in Mountain View, Calif. Please contact Larry Kluger, Sun Microsystems, at (415) 336-4708 for meeting time and location.

###

COMPANY CONTACTS:

Anvil Software Ltd, Frank Humphrey-Gaskin telephone: 011 44 71 729 8036

Digital Equipment Corporation, Dave Price telephone: (603) 884-3467

Fulcrum Technologies Inc., Peter Eddison telephone: (613) 238-1761

Hewlett-Packard Company, Diane Dennis telephone: (303) 229-2255

Highland Software, Mel Sibony/Jon Lane telephone: (415) 493-8567

Interactive Systems Corporation, Maury Domengeaux telephone: (213) 453-8649 x3106

Lehman Brothers, Scott Menter telephone: (212) 464-8737

Mentor Graphics Corporation, Rich Bruer telephone: (503) 685-7000 x1231

Meridian Data, Sheri Elpren telephone: (408) 438-3100

Next Technology Corp., Graham Brown-Martin telephone: 011 44 223 420 222

Philips Kommunications Industrie, Dr. K. Meissner telephone: 011 49 271 380 2574

The Santa Cruz Operation, Inc., Zee Zaballos telephone: (408) 425-7222 x5561

Silicon Graphics Computer Systems, Renee Wildman telephone: (415) 335-1026

Solbourne Computer, Steve Kowalski telephone: (303) 678-4382

Sun Microsystems, Inc., Carol Sacks telephone: (415) 336-0521

Young Minds, Andy Young telephone: (714) 335-1350

Rock Ridge Group Goals Document

I. Purpose

This document has the following purposes:

a. establish and document common goals for group participants
b. define a charter for the technical working group
c. act as a tool for gaining consensus within participating companies
d. act as a tool for gaining consensus with non-participating companies; this document may serve as the first formal release from the group
e. communicate our group's intentions and interest with respect to the CD-WO ad hoc group and other groups

II. Executive Summary

The companies participating in the Rock Ridge Group CD-ROM initiative desire the ability to use a CD-ROM as a *complete* implementation of X/Open filesystem directories. This would allow CD-ROM technology to be used for

+ software distribution in a heterogeneous environment
+ on-line access to CD-ROM executable, data and library files
+ database distribution in a heterogeneous environment

without restrictions in a complete X/Open environment.

Today, there is no commonly agreed CD-ROM format for accomplishing these goals. The purpose of the Rock Ridge initiative is to create and agree upon a common format that meets the needs presented above while maintaining compatibility with the installed base of ISO 9660:88 CD-ROM hardware and software.

The remainder of this document provides further information on the goals of the group.

III. Desired Benefits from different viewpoints

Publisher

Provide product distribution via 1 disc to all platforms--
The publisher should not be required to tailor CD-ROM products to different de livery platforms (PC vs. X/Open vs. OS/2 servers).

OS Vendor

Minimize software investment; single initial investment and reduce ongoing investments.
Allow one software solution which meets all X/Open needs.
Provide full X/Open File information.
Allow future extensions for security purposes.
Discourage ad hoc file format solutions.
Allow current software applications to run without modification using a CD-ROM.

Customers

Allow 1 disc to be used for all types of platforms.
Provide delivery of file data in a heterogeneous environment.
Provide access to all of a disc's data via ISO 9660:88 file data.
Optimize performance for operations involving X/Open extensions.

IV. Motivations for solving the problem co-operatively, not individually

Single vendor solutions are no longer a viable strategy in today's business climate.

Third party X/Open software vendors (ISV's) want to minimize the business risk of moving to CD-ROM as a new distribution format. They want a single solution for the open systems market, not multiple solutions or sub-solutions.

Facilitate ISV expansion to support X/Open systems by current DOS and Macintosh CD-ROM publishers.

Provide the basis for creating an international standard.

V. Definition of terms

'file data', 'file information', 'information' - the data contained within a file.
'file attributes' - the name of the file, dates associated with the data, permission modes, etc.

VI. Goals

1. Information interchange

   Maintain the ISO 9660:88 information interchange capability between all ISO 9660:88 and Rock Ridge systems.

2. File attribute interchange

   Allow interchange of file attribute information among X/Open class systems. The following file attributes should be supported as defined by X/Open: UID, GID, mode bits, major and minor device, UID, and GID numbers by receiving systems shall be provided.

3. Complete

   The format standard should meet the needs of distribution X/Open file data and file attribute information. No additional or optional tagged fields shall be required to deliver file information required for complete X/Open systems.

4. Execution

   The format shall allow direct execution of software from the CD-ROM in a heterogeneous environment.

5. Full backwards compatibility

   All file data on discs conforming to these extensions must be accessible on current Microsoft CD-ROM Extensions version 2.1 and other current ISO-9660:88/level 1 interchange drivers.

Discs conforming to this specification must meet the requirements of the proposed XCDR API [Philips]. (Assuming adoption of XCDR by X/Open.)

6. File names

   Extremely long name lengths shall be possible. Name lengths as long as possible shall be handled in an optimal manner.

   'Reader makes right' interchange: it is the responsibility of the receiving system to modify names as needed.

   Support ISO 8859/1:87 character set for file names and allow for support of future character set standards.

7. Path names

   Extremely long path names shall be possible. A system independent method for defining pathname components shall be provided. There shall not be any limitations on directory depth.

8. Application Programming Interfaces

   No impact on X/Open system APIs including XCDR. If necessary, an additional API will be proposed.

9. Service Interfaces

   Same as the following sections in XPG 3 v2 [88].
   2.1.19 Directory
   2.1.20 Directory Entry including symbolic links
   2.1.26 Empty Directory
   2.1.32 File
   2.1.33 File Access Permissions
   2.1.37 File Hierarchy
   2.1.38 File Mode
   2.1.39 Filename except, allow longer names. See VI.6.
   2.1.43 File Permission Bits
   2.1.46 File Times Update as defined for read-only filesystems
   2.1.71 Portable Filename Character Set-- use XPG 3 specified ISO 8859/1 character set except no reserved characters.

10. Performance

    The format shall be optimized for on-line access to a disc's executable, data and library files. In addition, the format shall optimize for directory information access.

VII. Non-Goal

Common Application data format. We do not seek to define a common data format. For example, requiring ASN.1 descriptions of file data. Binary executable formats are also outside the scope of the Rock Ridge Group.

Reference

[Philips] X/Open CD-ROM Support Component (XCDR). Version 2, June 1990. This document is available from Hans de Jong, Philips Information Systems. Telephone +31 55 43 27 74.

Appendix

The appendix contains additional information that may be of interest to the reader.

Discussion of information interchange goal. Reference: VI.1.
We define two types of information interchange between platforms.

Alternative 1 - Common file system format and all file data can be read by all platforms.
Alternative 2 - Common file system format and all file data can be read by some platforms.

Alternative 1: This type of interchange maintains the original ISO 9660:88 concept of "...[enabling] information to be interchanged between different systems..." [Section 1, ISO 9660:88].

This type of data interchange assures the accessibility of files in a heterogeneous work group environment. Any type of platform connected to a common LAN would be able to access all of the information on a disc, no matter the type of disc server.

This type of data interchange encourages the creation of "open" applications whose data is available to all users.

Counter argument: some architects feel that it would be desirable for the file labeled 'start' to refer to different file information dependent on the machine type or architecture. The user would only have to invoke 'start' no matter his or her machine type.

But this would make information inaccessible in many circumstances. In addition, the same effect can be gained within the X/Open market by writing a single 'shell script' which used the uname(1) command to execute the correct binary file.

It may be difficult or impossible for a common shell script to probe for and correctly handle the needs of X/Open, DOS, and other types of systems.

Alternative 2: This type of interchange would allow the CD-ROM to contain files which are available to some but not all types of receiving platforms. For example, in a network environment, an X/Open platform which acted as a server might not be able to export the DOS files on a CD-ROM disc to the networked DOS PC's since the files would be invisible to the X/Open platform.

Discussion of File attribute interchange goal. Reference: VI.2.

UID and GID mapping between values stored on the disc and values used by a particular X/Open system are defined by the XCDR proposal. Per section VI.5, XCDR's mechanisms shall be used whenever possible.

Discussion of Execution goal. Reference: VI.4.

Execution of software directly from a CD-ROM disc without first copying the software to magnetic disc is desirable for several reasons:

Ease of use: no installation of the software onto magnetic disc means that the user can "load and go" his or her software. By providing a complete runtime environment on the CD-ROM, a publisher can reduce the complexity of installations. The application would not have to be adapted to a specific system's file layout.

Ease of documentation: A CD-ROM can contain many software packages which are infrequently used or which are meant as demonstration/trial applications. Direct execution allows the user to demonstrate or use low-usage applications without having to load and then unload an application each time it is used.

No consumption of magnetic disc resources: often, magnetic disc resources are highly constrained on an operational system. Direct execution from the CD-ROM means that magnetic disc space does not need to be planned, allocated or consumed before using software.

Discussion of Backwards compatibility goal. Reference: VI.5.

Backwards compatibility with current ISO 9660:88 hardware and software is required for the following reasons:

It is unrealistic to expect that the embedded base of current CD-ROM drives will be updated to take advantage of a new CD-ROM format specification. Discs produced using this specification must be readable by the installed base or publishers will not make use of it.

Maintaining compatibility will reduce ISV concerns about migrating to use a new disc format.

Publishers wish to leverage their ISO 9660:88 experience and expertise.

Discussion of File name goal. Reference: VI.6.

Character and byte counts may be different when a multibyte character set is used. In particular, the 'space' and 'solidus' (/) characters may be used without restriction in file names. Also, any number (including none) 'full stop' (.) characters may be used.

Character names are from ISO 8859/1:87.

Discussion of Path names goal. Reference: VI.7.

Path names shall not be stored as a single character string that reserves some character such as 'solidus' (/) as a separator. Directory depths for X/Open systems are often very deep. For example, the widely used X11 window system from MIT includes directories that are 11 steps deep.

Discussion of Performance goal. Reference: VI.10.

Directory information access should be optimized by physically placing all directory information in one area. Directory information (file dates, etc) should not only be physically near the file information.

ROCK RIDGE INTERCHANGE PROTOCOL

VERSION 1

AN IS0 9660:1988 COMPLIANT APPROACH TO PROVIDING ADEQUATE CD-ROM SUPPORT FOR POSIX FILE SYSTEM SEMANTICS

ROCK RIDGE TECHNICAL WORKING GROUP

Revision 1.09

cdtec@dgdo.Eng.Sun.COM

PROPOSAL

(This page should be replaced with a blank page for back of title)

CONTENTS

(This page should be replaced with a blank page for back of TOC)

LIST OF TABLES

(This page should be replaced with a blank page for back of Tables)

1. PREFACE

1.1 Purpose and Scope

Producers and users of POSIX compliant systems and software have faced a significant, yet artificial, barrier to their effectively using CD-ROM technology for software distribution and information publishing -- ISO 9660:1988 alone provides inadequate support for delivery of POSIX file system information. The Rock Ridge Group was formed to generate a proposed standard for utilizing the System Use Areas provided by the ISO 9660 standard to record complete POSIX file system semantics. This proposal utilizes the System Use Sharing Protocol for recording this information.

1.2 Summary of Sections

Section 1	Contains this preface.
Section 2	Contains an overview of the Rock Ridge Interchange Protocol.
Section 3	Contains an overview of the notation used in this document.
Section 4	Contains the Rock Ridge Interchange Protocol.
Section 5	Contains the RRIP Application Programming Interface.
Section 6	Contains the bibliography.

2. OVERVIEW

The Rock Ridge Interchange Protocol (RRIP) specifies an extension to the ISO 9660 format for CD-ROM which enables the recording of POSIX File System semantics. The RRIP utilizes the System Use Sharing Protocol (SUSP) to specify the definition of a set of System Use Fields for this purpose.

The RRIP specifies the definition of a set of System Use Fields for recording:

- uid, gid, and permissions
- file mode bits, file types, setuid, setgid, and sticky bit
- file links
- device nodes
- symbolic links
- POSIX file names
- reconstruction of deep directories
- time stamps

3. TERMINOLOGY AND NOTATION

It is assumed that the RRIP is being recorded within an ISO 9660:1988 compliant volume using the System Use Sharing Protocol (SUSP:1991A). Unless defined herein, or otherwise specified, terms shall be as defined in ISO 9660:1988 or SUSP:1991A.

The following notation is used in this document.

3.1 Decimal and Hexadecimal Notation

Numbers in decimal notation are represented by decimal digits, namely 0 to 9.

Numbers in hexadecimal notation are represented by hexadecimal digits, namely 0 to 9 and A to F, shown in parentheses. E.g. the hexadecimal number 7F will be written as (7F).

3.2 File Naming Conventions

In all fields defined in ISO 9660:1988, the character set to be used shall be as specified in ISO 9660. The character set to be used in the System Use Fields defined herein shall depend upon whether the fields are recorded in a directory tree associated with a Primary Volume Descriptor or a Supplementary Volume Descriptor.

3.2.1 Primary Volume Descriptor File Naming Convention

Within a directory tree identified by a Primary Volume Descriptor of an ISO 9660 volume, the character set used in the System Use Fields defined for the RRIP shall be the ISO Standard 8859-1:1987, as specified in the X/Open Portability Guide Issue 3, XSI Supplementary Definitions. For general portability across most POSIX compatible systems, the 7-bit U.S. ASCII character set should be used.

The POSIX File Naming Convention states that the filename may be composed of any character except slash (2F) and the null byte (00). The special filename, dot (2E), refers to the directory specified by its predecessor. The special filename dot-dot (2E)(2E), refers to the parent directory of its predecessor directory.

For maximum portability across implementations conforming to the POSIX Standard, filenames should only contain the following characters:

(41) thru (5A)	'A' thru 'Z'
(61) thru (7A)	'a' thru 'z'
(30) thru (39)	'0' thru '9'
(2E)	period
(5F)	underscore
(2D)	hyphen

Upper and lower case letters shall retain their unique identities between conforming implementations.

3.2.2 Supplementory Volume Descriptor File Naming Convention

Within a directory tree identified by a Supplementary Volume Descriptor of an ISO 9660 volume, the character set used in the System Use Fields defined for the RRIP shall be the coded graphic character set(s) identified by the escape sequence(s) in the Supplementary Volume Descriptor (c-characters, section 7.4.2, ISO 9660:1988).

3.3 Reader Makes Right

Receiving systems which are capable of interpreting the System Use Fields defined herein, but which cannot handle the full extent of the file naming convention utilized by this specification may have to restrict themselves to the use of the ISO 9660 compliant file names stored in the File Identifier field of the ISO 9660 directory structure.

Alternatively, the developer of the receiving system may choose to provide file name conversion capability. Any such system must provide unique names for all unique files on the disc.

In general, whenever there is recorded a (potentially) system-specific identifier or numerical value, accomplishing any necessary modifications or mapping of these are the responsibility of the receiving system. This document provides for an Application Programming Interface (API) to support this function.

4. ROCK RIDGE INTERCHANGE PROTOCOL

The Rock Ridge Interchange Protocol (RRIP) utilizes System Use Areas provided by ISO 9660:1988. The System Use Area of the Directory Record is used to record the POSIX file system information. The System Use Sharing Protocol is used for recording information in each of these areas.

4.1 System Use Fields Provided by this Specification

The Rock Ridge Interchange Protocol defines the following fundamental System Use Fields:

"PX"	POSIX file attributes
"PN"	POSIX device modes
"SL"	Symbolic link
"NM"	Alternate name
"CL"	Child link
"PL"	Parent link
"RE"	Relocated directory
"TF"	Time stamp(s) for a file
"RR"	Flags indicating which fields are recorded

Additionally, this specification provides required identification information to be recorded in an "ER" (Extensions Reference) System Use Field for the purpose of identifying discs on which the Rock Ridge Interchange Protocol is implemented.

4.1.1 Description of the PX System Use Field

Recording of the "PX" System Use Field in the System Use Area of each ISO 9660 directory record shall be mandatory. No more than one "PX" System Use Field shall appear in (all) the System Use Area(s) for a single directory record.

If the file type in a directory record is of type directory, then the POSIX File Mode Field ([BP 4] in this section) and File Flags (ISO 9660 Format section 9.1.6) should both indicate such, with the exception of relocated directories, indicated by a "CL" field (section 3.1.5.1), for which the ISO file flags shall indicate a normal file, but the POSIX File Mode Field shall indicate a directory.

The format of the "PX" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "PX" type System Use Field. The bytes in this field shall be (50)(58) ("PX").

[2] "BP 3 - Length" shall specify as an 8-bit number the length in bytes of the "PX" System Use Field. The number in this field shall be 36 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "PX" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[4] "BP 5 to BP 12 - POSIX File Mode" has the same meaning as the st_mode defined in the header sys/stat.h by the IEEE Std 1003.1-1988. This field shall be recorded according to ISO 9660:1988 Format section 7.3.3. The valid mask values for this field are combinations of the following:

Octal Value	Meaning
0000400	read permission (owner)
0000200	write permission (owner)
0000100	execute permission (owner)
0000040	read permission (group)
0000020	write permission (group)
0000010	execute permission (group)
0000004	read permission (other)
0000002	write permission (other)
0000001	execute permission (other)
0004000	set user ID on execution
0002000	set group ID on execution
0002000	enforced file locking (shared w/ set group ID)
0001000	save swapped text even after use
0170000	type of file
0140000	socket
0120000	symbolic link
0100000	regular
0060000	block special
0020000	character special
0040000	directory
0010000	pipe or FIFO

[5] "BP 13 to BP 20 - POSIX File Links" has the same meaning as the st_nlink defined in the header sys/stat.h by the IEEE Std 1003.1-1988. If the file type described by the directory record is a directory, the value in this field shall equal the number of directories in the directory described by this directory record (i.e. any directory record which has the "directory" bit set, including the "." and ".." records). Otherwise, it shall be 1. This field shall be recorded according to ISO 9660:1988 Format section 7.3.3.

[6] "BP 21 to BP 28 - POSIX File User ID" has the same meaning as the st_uid defined in the header sys/stat.h by the IEEE Std 1003.1-1988. This field shall be recorded according to ISO 9660:1988 Format section 7.3.3.

[7] "BP 29 to BP 36 - POSIX File Group ID" has the same meaning as the st_gid defined in the header sys/stat.h by the IEEE Std 1003.1-1988. This field shall be recorded according to ISO 9660:1988 Format section 7.3.3.

TABLE 1. PX System Use Field - Version 1

'P' (BP1)	'X' (BP2)	LENGTH (BP3)	1 (BP4)	FILE MODE (BP5 to BP12)	LINKS (BP13 to BP20)

USER ID (BP21 to BP28)	GROUP ID (BP29 to BP36)

4.1.2 Description of the PN System Use Field

This field is mandatory if the file type recorded in the "PX" File Mode field for the given directory record indicates a character or block device. This field, if present, should be ignored for all other file types. No more than one "PN" System Use Field shall appear in (all) the System Use Area(s) for a single directory record.

If the receiving system records device numbers as 32-bit numbers, only the "Dev_t Low" field shall be used. If the receiving system records device numbers as 64-bit numbers, the "Dev_t High" and "Dev_t Low" fields shall be concatenated to make one 64-bit number.

The format of the "PN" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "PN" type System Use Field. The bytes in this field shall be (50)(4E) ("PN").

[2] "BP 3 - Length" shall specify as an 8-bit number the length in bytes of the "PN" System Use Field. The number in this field shall be 20 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "PN" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[4] "BP 5 to BP 12 - Dev_t High" shall contain as a 32-bit number the high order 32 bits of the device number. If the receiving system records device numbers as 32-bit numbers this field shall be zero and ignored. This field shall be recorded according to ISO 9660:1988 Format section 7.3.3.

[5] "BP 13 to BP 20 - Dev_t Low" shall contain as a 32-bit number the low order 32-bits of the device number. This field shall be recorded according to ISO 9660:1988 Format section 7.3.3.

TABLE 2. PN System Use Field - Version 1

'P' (BP1)	'N' (BP2)	20 (BP3)	1 (BP4)	DEV_T HIGH (BP5 to BP12)	DEV_T LOW (BP13 to BP20)

4.1.3 Description of the SL System Use Field

The purpose of the "SL" System Use Field is to store the content of a symbolic link. This System Use Field is mandatory if the file type recorded in the "PX" File Mode field for the given directory record indicates a symbolic link. For other file types, this System Use Field should be ignored. If the receiving system does not support symbolic links the system should invoke "Reader-Makes-Right".

If the file type recorded in the "PX" File Mode field for the given directory record indicates a symbolic link, the directory record shall point to a file, the contents of which are not specified by this document.

If more than one "SL" System Use Field is recorded in the System Use Area(s) for a single directory record, the Component Area (see section 4.1.3.1 below) of each should be concatenated together, in the order in which they were recorded, until a CONTINUE flag with value zero is encountered (see [4] below), to obtain the entire set of Component Records for the symbolic link.

The method of recording is system independent. Under reader makes right, the receiving system is responsible for supplying appropriate values and/or notations for the component delimiter and special cases provided for below.

The format of the "SL" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "SL" type System Use Field. The bytes in this field shall be (53)(4C) ("SL").

[2] "BP 3 - Length (LEN_SL)" shall specify as an 8-bit number the length in bytes of the "SL" System Use Field. The number in this field shall be 5 plus the length of the Component Area recorded in this "SL" field. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "SL" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[4] "BP 5 - Flags" shall contain bit field flags numbered 0 to 7 starting with the least significant bit as follows:

Position	Name	Interpretation if set to 1
0	CONTINUE	This Symbolic Link continues in next "SL"
all others	RESERVED	value must be 0

[5] "BP 6 to LEN_SL - Component Records" shall contain Component Records (described below).

TABLE 3. SL System Use Field - Version 1

'S' (BP1)	'L' (BP2)	LENGTH (BP3)	1 (BP4)	FLAGS (BP5)	COMPONENT RECORDS (BP6 to LEN_SL)

4.1.3.1 Description of the SL System Use Field Component Record

Within a "SL" System Use Field, each component of the pathname shall be recorded as one or more component records. A component does not contain the component separator (/ in POSIX). Recording a single component of a symbolic link may require multiple Component Records. If the component is greater than 255 bytes or will not fit into the current System Use Area or Continuation Area more than one Component Record will be recorded for the component. Multiple Component Records, specifying one or more separate components of the symbolic link may be recorded in the Component Area of a single "SL" field.

If a single Component Record is used to record a single component of a symbolic link, the CONTINUE flag must be set to zero. If multiple Component Records are used to record a single component of a symbolic link, the CONTINUE flag must be set to one in each Component Record except the last and zero in the last Component Record recording the given component.

Component Records shall be recorded contiguously within each Component Area, starting in the first byte of each Component Area. The last Component Record in the Component Area of an "SL" field may be continued in the Component Area of the next recorded "SL" field.

Each Component Record shall have the following format:

[A] "BP 1 - Component Flags" shall contain bit field flags numbered 0 to 7 starting with the least significant bit as follows:

Position	Name	Interpretation if set to 1
0	CONTINUE	Component recorded in this "SL" continues in next "SL" Component Record
1	CURRENT	Component refers to the current directory (. in POSIX)
2	PARENT	Component refers to the parent of the current directory (.. in POSIX)
3	ROOT	Component refers to the root of the current directory tree for this process (/ in POSIX)
4	VOLROOT	Component refers to the directory the current CD-ROM volume is mounted on
5	HOST	The local host name should be inserted as the value of the current component
all others	RESERVED	value must be 0

Bits 1 - 7 are mutually exclusive.

[B] "BP 2 - Component Length (LEN_CP)" shall specify as an 8-bit number the length in bytes of the (portion of) the component recorded in the current Component Record. This length shall not include the Component Record Flags byte or Length byte. If any of the 2^1 thru 2^5 bits are set, the value of this field shall be zero. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[C] "BP 3 to 2 + LEN_CP - Component" shall contain (the portion of) the component recorded in the current Component Record. The content of this field shall be recorded according to section 3.2 above.

TABLE 4. SL System Use Field - Component Record

COMP_FLAGS (BP1)	COMP_LEN (BP2)	COMPONENT (BP3 to 2+LEN_CP)

4.1.4 Description of the NM System Use Field

The purpose of the "NM" System Use Field is to store the content of an Alternate Name to support POSIX-style or other names. This System Use Field is optional. If no "NM" field(s) are recorded for a specific directory record, the ISO 9660 file identifier shall be used.

If more than one "NM" System Use Field appears in (all) the System Use Area(s) for a single directory record, the contents ([5] below) of each should be concatenated together, in the order in which they were recorded, until a CONTINUE flag with value zero is encountered (see [4] below), to obtain the entire Alternate Name.

"NM" System Use Fields recorded for the ISO 9660 directory records with names (00) and (01), used to designate the current and parent directories, respectively, should be ignored. Instead, the receiving system should convert these names to the appropriate receiving system-dependent designations for the current and parent directories.

No sorting of the directory records by Alternate Names is specified by the RRIP, nor can one necessarily be imposed by originating systems or assumed by receiving systems. The ISO 9660 specifies a sorting order based upon the ISO 9660 file identifier (see ISO 9660:1988, section 9.3).

The format of the "NM" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "NM" type System Use Field. The bytes in this field shall be (4E)(4D) ("NM").

[2] "BP 3 - Length (LEN_NAM)" shall specify as an 8-bit number the length in bytes of the "NM" System Use Field. The number in this field shall be 5 plus the length (of the portion) of the Alternate Name recorded in this "NM" field. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "NM" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[4] "BP 5 - Flags" shall contain bit field flags numbered 0 to 7 starting with the least significant bit as follows:

Position	Name	Interpretation if set to 1
0	CONTINUE	Alternate Name continues in next "NM" field
1	CURRENT	Alternate Name refers to the current directory (. in POSIX)
2	PARENT	Alternate Name refers to the parent of the current directory (.. in POSIX)
3	RESERVED	value must be 0
4	RESERVED	value must be 0
		of the current CD-ROM volume
5	HOST	The local host name should be inserted as the value of the Alternate Name
all others	RESERVED	value must be 0

Bits 1 - 7 are mutually exclusive.

[5] "BP 6 to LEN_NAM - Alternate Name" shall contain (a portion of) the content of the Alternate Name. The content of this field shall be recorded according to section 3.2 above.

TABLE 5. NM System Use Field - Version 1

'N' (BP1)	'M' (BP2)	LENGTH (BP3)	1 (BP4)	FLAGS (BP5)	ALTERNATE NAME (BP6 to LEN_NAM)

4.1.5 System Use Fields for Handling Deep Directory Trees

The ISO 9660:1988 mandates directory depths of no more than eight levels. Deeper directories must be reorganized to be recorded under the ISO 9660. The RRIP includes definitions of three System Use Fields to support logical reconstruction of deep directory trees while retaining complete ISO 9660 compliance.

For each specific directory, either all three of the following fields must be appropriately recorded, or none shall be recorded.

Table 9 and Table 10 at the end of this section have graphical examples of Deep Directory Trees.

4.1.5.1 Description of the CL System Use Field

The purpose of the "CL" System Use Field is to record the new location of a directory which has been relocated. The field contains the Logical Block number of the Logical Sector in which the first directory record of the moved Directory is stored.

The "CL" System Use Field is optional. If recorded, a "CL" System Use Field shall be recorded in the System Use Area of a ISO 9660 directory record which describes a file which has the same name as, and occupies the original position in the ISO 9660 directory tree of, the moved Directory. No more than one "CL" System Use Field shall appear in (all) the System Use Area(s) for a single directory record.

Except for the ISO 9660 name, the Alternate Name (recorded in an "NM" System Use Field), and the new location of the Directory, all other information stored in the directory for this file should be ignored. The contents of this file are not specified by this document. All attributes of the moved Directory shall be recorded in the first directory record ("dot" entry) of the moved Directory in its new location.

The format of the "CL" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "CL" type System Use Field. The bytes in this field shall be (43)(4C) ("CL").

[2] "BP 3 - Length" shall specify as an 8-bit number the length in bytes of the "CL" System Use Field. The number in this field shall be 12 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "CL" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[4] "BP 5 to BP 12 - Location of Child Directory" shall specify as a 32-bit number the Logical Block Number of the first Logical Block allocated to the moved directory. This field shall be recorded according to ISO 9660:1988 Format section 7.3.3.

TABLE 6. CL System Use Field - Version 1

'C' (BP1)	'L' (BP2)	12 (BP3)	1 (BP4)	LOC of CHILD DIRECTORY (BP5 to BP12)

4.1.5.2 Description of the PL System Use Field

The purpose of the "PL" System Use Field is to record the location of the original parent Directory of a Directory which has been relocated. The field contains the Logical Block number of the Logical Sector in which the first directory record of the original parent Directory of said moved Directory is stored.

For each moved Directory which is recorded using a "CL" System Use Field, a corresponding "PL" System Use Field is required. The "PL" System Use Field shall be recorded in the System Use Area of the second directory record ("dot-dot" entry) of the moved Directory. No more than one "PL" System Use Field shall appear in (all) the System Use Area(s) for a single directory record.

The format of the "PL" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "PL" type System Use Field. The bytes in this field shall be (50)(4C) ("PL").

[2] "BP 3 - Length" shall specify as an 8-bit number the length in bytes of the "PL" System Use Field. The number in this field shall be 12 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "PL" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[4] "BP 5 to BP 12 - Location of Parent Directory" shall specify as a 32-bit number the Logical Block Number of the first Logical Block allocated to the original parent directory of the moved directory. This field shall be recorded according to ISO 9660:1988 Format section 7.3.3.

TABLE 7. PL System Use Field - Version 1

'P' (BP1)	'L' (BP2)	12 (BP3)	1 (BP4)	LOC of PARENT DIRECTORY (BP5 to BP12)

4.1.5.3 Description of the RE System Use Field

The purpose of the "RE" System Use Field is to indicate to a receiving system which can understand the System Use Fields "CL" and "PL" that the directory record in which this "RE" System Use Field is recorded has been relocated from another position in the original directory tree.

An "RE" System Use Field shall not be recorded unless a corresponding "CL" System Use Field is recorded. If recorded, a "RE" System Use Field shall be recorded in the System Use Area of the directory record which describes the moved Directory in the new parent directory of the moved Directory.

The format of the "RE" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "RE" type System Use Field. The bytes in this field shall be (52)(45) ("RE").

[2] "BP 3 - Length" shall specify as an 8-bit number the length in bytes of the "RE" System Use Field. The number in this field shall be 4 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "RE" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

TABLE 8. RE System Use Field - Version 1

'R' (BP1)	'E' (BP2)	4 (BP3)	1 (BP4)

TABLE 9. Deep Directory Relocation

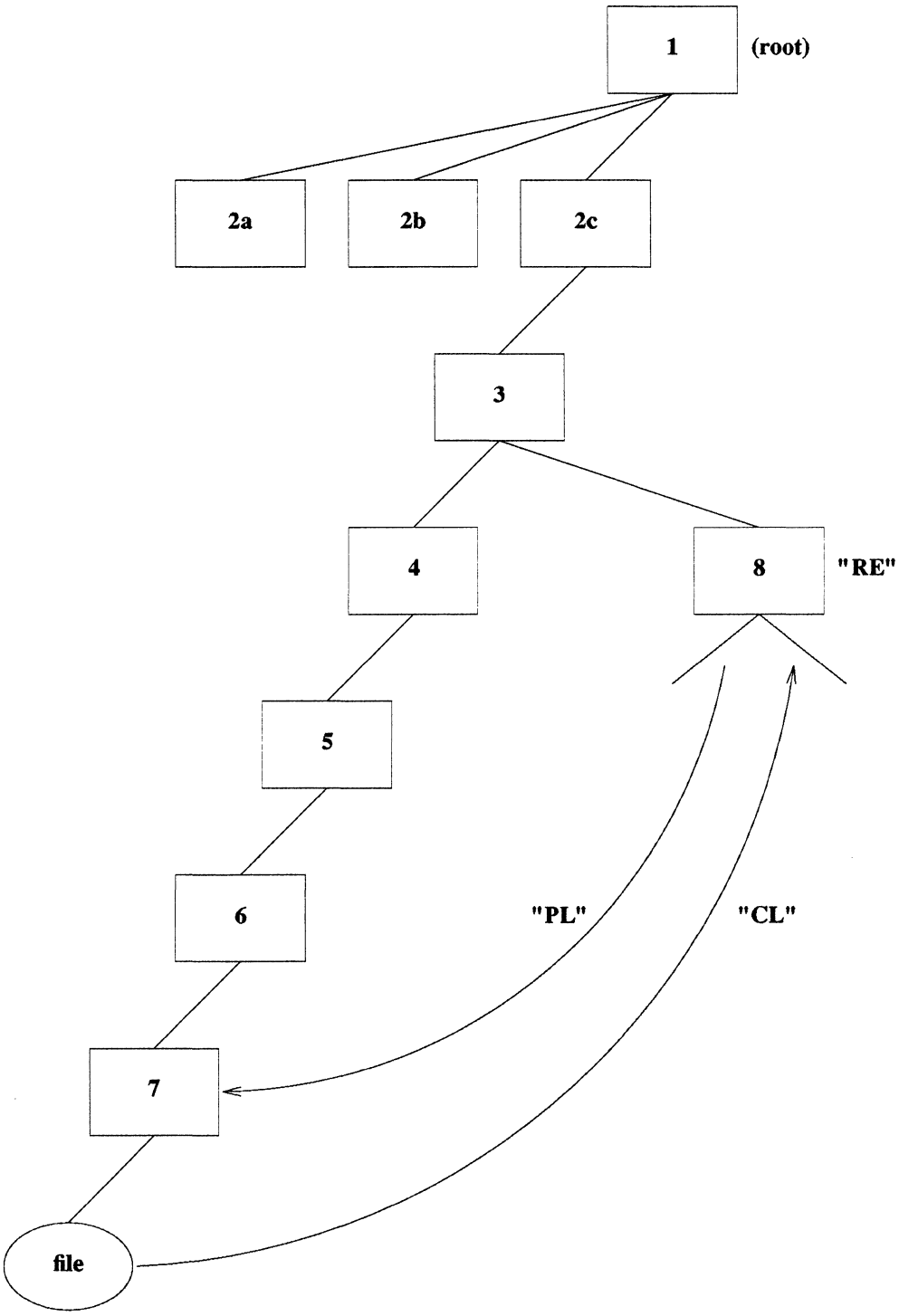
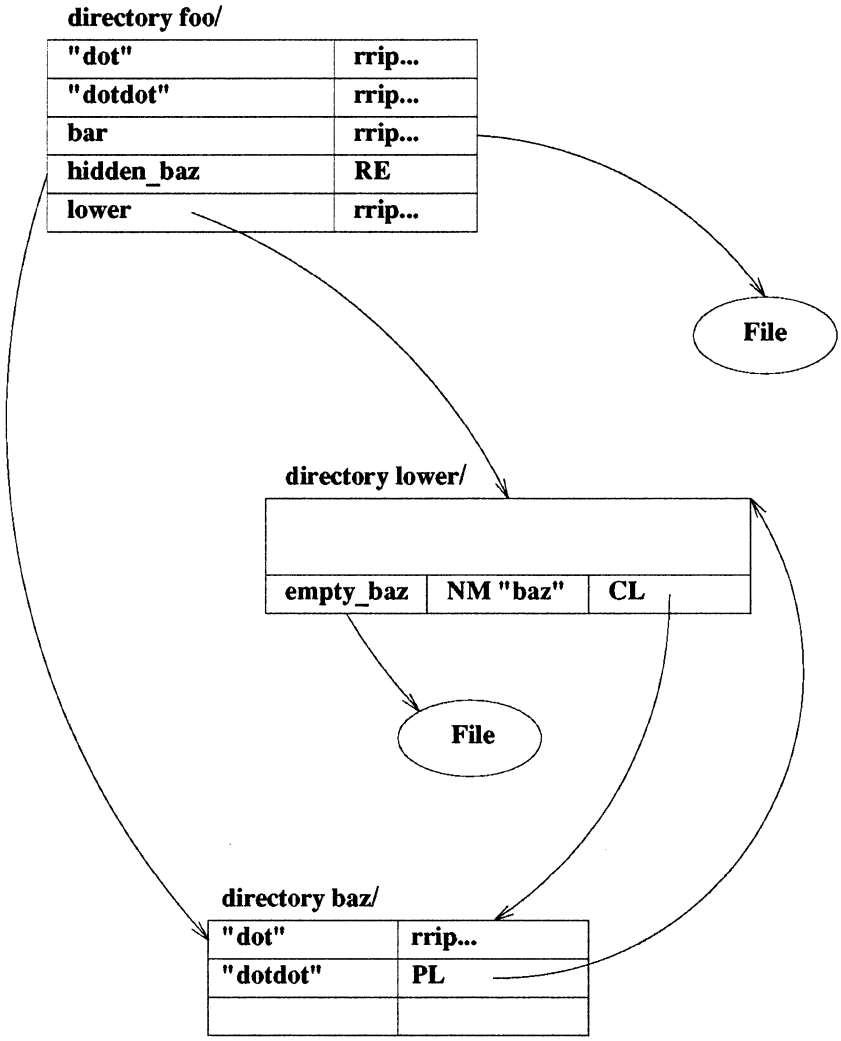


TABLE 10. Detailed Deep Directory Relocation



4.1.6 Description of the TF System Use Field

The purpose of the "TF" System Use Field is to allow the recording of a complete set of time stamps related to a file. This System Use Field shall be optional. If this field does not exist, the POSIX st_atime, st_ctime and st_mtime should have the same value as Recording Date and Time Field of the ISO 9660:1988 directory record. If both the "TF" System Use Field and the XAR are present, the time attributes stored in these two areas must be consistent. If only the XAR is present, the st_atime should have the same value as the Recording Date and Time Field of the ISO 9660 directory record.

Multiple "TF" fields may be recorded, using any combination of time stamps and time formats, but each individual time stamp may be recorded only once per directory record.

The format of the "TF" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "TF" type System Use Field. The bytes in this field shall be (54)(46) ("TF").

[2] "BP 3 - Length" shall specify as an 8-bit number the length in bytes of the "TF" System Use Field. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "TF" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[4] "BP 5 - Flags" shall contain bit field flags numbered 0 to 7 starting with the least significant bit as follows:

Position	Name	Interpretation if set to 1
0	CREATION	Creation time recorded
1	MODIFY	Modification time recorded
2	ACCESS	Last Access time recorded
3	ATTRIBUTES	Last Attribute Change time recorded
4	BACKUP	Last Backup time recorded
5	EXPIRATION	Expiration time recorded
6	EFFECTIVE	Effective time recorded
7	LONG_FORM	ISO 9660 17-byte time format used

If the LONG_FORM bit is set to one, all time stamps in this "TF" System Use Field shall be recorded using the format specified in Section 8.4.26.1 of ISO 9660:1988. If the LONG_FORM bit is set to zero, all time stamps in this "TF" System Use Field shall be recorded using the format specified in Section 9.1.5 of ISO 9660:1988.

[4+N] "BP 6+(X*(N-1)) to 5+(X*N) Time Stamp" shall contain the Nth time stamp indicated in [4] as being recorded, starting with the 0th bit and working sequentially through the list of recordable time stamps. The LONG_FORM bit does not indicate the presence or absence of any time stamp. The value of X in the expression above shall be 17 if the LONG_FORM bit is set to 1, and 7 otherwise.

The recorded time for each of the time stamps recorded in this field shall be local time. if recorded, CREATION, Creation Time, has the same meaning as in ISO 9660:1988 Format section 9.5.4.

If recorded, MODIFY, File Modification Date and Time, has the same meaning as in ISO 9660:1988 Format section 9.5.5. This field shall be used by the st_mtime for POSIX conformance.

If recorded, ACCESS, File Last Access Date and Time, shall specify the date and time of the day at which the information in the file was last accessed. This field shall be used by the st_atime for POSIX conformance.

If recorded, ATTRIBUTES, Last Attribute Change Time, shall be used by the st_ctime field for POSIX conformance.

If recorded, BACKUP, Last Backup Time, shall provide a time stamp for the most recent backup of this file. The utilization of this information is not restricted by this specification.

If recorded, EXPIRE, File Expiration Date and Time, has the same meaning as in ISO 9660:1988 Format section 9.5.6.

If recorded, EFFECT File Effective Date and Time" has the same meaning as in ISO 9660:1988 Format section 9.5.7.

TABLE 11. TF System Use Field - Version 1

'T' (BP1)	'F' (BP2)	LENGTH (BP3)	1 (BP4)	FLAGS (BP5)	TIME STAMPS (BP6 to LENGTH)

4.1.7 Description of the RR System Use Field

The purpose of the "RR" System Use Field is to indicate which System Use Fields defined by the Rock Ridge Interchange Protocol are actually recorded for the current directory record. This System Use Field is optional. No more than one "RR" System Use Field shall appear in (all) the System Use Area(s) for a single directory record.

The use of an "RR" field may allow some additional optimization in implementations which utilize this information to eliminate searching through the entire System Use Area (and all Continuations) for a System Use Field which may not even have been recorded. For this reason, if an "RR" field is recorded, it should be one of the first fields recorded in the System Use Area.

The format of the "RR" System Use Field is as follows:

[1] "BP 1 to BP 2 - Signature Word" shall indicate that the System Use Field is a "RR" type System Use Field. The bytes in this field shall be (52)(52) ("RR").

[2] "BP 3 - Length" shall specify as an 8-bit number the length in bytes of the "RR" System Use Field. The number in this field shall be 5 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[3] "BP 4 - System Use Field Version" shall specify as an 8-bit number an identification of the version of the "RR" System Use Field. The number in this field shall be 1 for this version. This field shall be recorded according to ISO 9660:1988 Format section 7.1.1.

[4] "BP 5 - Flags" shall contain bit field flags numbered 0 to 7 starting with the least significant bit as follows:

Position	If set to 1, indicates, for this Directory Record
0	"PX" System Use Field recorded
1	"PN" System Use Field recorded
2	"SL" System Use Field recorded
3	"NM" System Use Field recorded
4	"CL" System Use Field recorded
5	"PL" System Use Field recorded
6	"RE" System Use Field recorded
7	"TF" System Use Field recorded

TABLE 12. RR System Use Field - Version 1

'R' (BP1)	'R' (BP2)	LENGTH (BP3)	1 (BP4)	FLAGS (BP5)

4.2 Required Recording and Consistency

The "PX" System Use Fields shall be recorded in every directory record. All recorded instances of the "PX" and "TF" System Use Fields in directory records referring to a single directory must be consistent.

4.3 Specification of the ER System Use Field Values for RRIP

The Extension Version number for the version of the RRIP defined herein shall be 1. The content of the Extension Identifier field shall be "RRIP_1991A". The Identifier Length shall be 10.

The recommended content of the Extension Descriptor shall be "THE ROCK RIDGE INTERCHANGE PROTOCOL PROVIDES SUPPORT FOR POSIX FILE SYSTEM SEMANTICS." The corresponding Description Length is 84.

The recommended content of the Extension Source shall be "PLEASE CONTACT DISC PUBLISHER FOR SPECIFICATION SOURCE. SEE PUBLISHER IDENTIFIER IN PRIMARY VOLUME DESCRIPTOR FOR CONTACT INFORMATION." The corresponding Source Length is 135.

5. RRIP APPLICATION PROGRAMMING INTERFACE (API)

This section specifies an Application Programming Interface (API) for the Rock Ridge Interchange Protocol which is implemented on top of the System Use Sharing Protocol. This API is a supplement to the X/Open CD-ROM Support Component (XCDR).

5.1 Mapping Device Files

The major and minor numbers of device files as recorded in the System Use Area on the CD-ROM may not match the major and minor numbers of the physical devices. If that is the case, the command, *cddevsuppl* can be used to change the major and minor numbers of a specific device file on the CD-ROM.

If the system imposes a maximum value on the number of device file mappings, this will be defined via the symbolic name CD_MAXDMAP in *<sys/cdrom.h>*. At least 50 device file mappings will be supported.

5.2 Obtaining CD-ROM Specific Information

5.2.1 System Use Sharing Protocol Fields

The CD-ROM contains System Use Fields in the System Use Areas which are specific to the CD-ROM and cannot be obtained by standard XSI system interfaces. Using the command *cdsuf,* or the equivalent library function, all additional information in a file or directory System Use Field can be accessed.

5.2.2 Changing PX Field Information

The "POSIX File User ID" and "POSIX File Group ID" can be mapped on the receiving system by using commands and library functions supplied by the X/Open CD-ROM Support Component (XCDR).

5.2.3 File Name Resolution

The fields *file, filename, path,* and *pathname* shall be resolved according to the Rock Ridge Interchange Protocol, which may be the ISO 9660 name or RRIP name depending on whether an "NM" System Use Field is present for any component of that filename.

5.3 Definition of CD-ROM Specific User Commands

This sections provides manual pages which describe CD-ROM user commands for users in detail. The user commands are:

Name	Description
cdsuf	Retrieve a System Use Field
cddevsuppl	Set and get major/minor numbers of a device file

5.3.1 cdsuf command

NAME

cdsuf - read the System Use Fields from a System Use Area

SYNOPSIS

cdsuf [-s number] [-b] file

DESCRIPTION

This command is used to access the System Use Fields of the System Use Area associated with a File Section of a file or directory and to list its contents on standard output, following any Continuation Fields that may be present.

OPTIONS

The following options are available:

-s number This option specifies the File Section for which the System Use Area shall be read. The numbering starts with one. If this option is omitted the last File Section of that file is assumed.

-b With this option all of the System Use Fields of the System Use Area are copied from the CD-ROM to standard output in binary format.

OPERAND

The operand *file* is the name of any file or directory within the CD-ROM file hierarchy.

STDIN

Not Used.

INPUT FILES

None.

ENVIRONMENT VARIABLES

LC_TIME determines the format and contents of date and time strings. If LC_TIME is not set in the environment or is set to the empty string, the value of LANG will be used as a default. If LANG is not set or set to the empty string, the corresponding value from the implementation-specific default locale will be used. If LC_TIME or LANG contain an invalid setting, the utility will behave as if none of the variables had been defined.

STDOUT

The output is formatted in the form of a table which contains an entry for each System Use Field in the System Use Area as recorded on the CD-ROM. Each entry of the table shall have the fields *Signature, Length, Version,* and *Data* as specified in the System Use Sharing Protocol. Whether to break up the *Data* field into smaller fields according to the protocol which defined the *Signature* field is left up to the implementation.

If the **-b** option is applied, the contents of the full System Use Area are written to standard output in binary format as it is recorded on the CD-ROM.

STDERR

Used only for diagnostic messages.

OUTPUT FILES

None.

EXIT STATUS

The following exit values are returned:

0	successful completion
1	file not found or file is not a file or directory within a CD-ROM file hierarchy or access denied
2	File Section indicated by -s does not exist
3	File Section indicated by -s has no System Use Area

CONSEQUENCES OF ERRORS

None.

APPLICATION USAGE

The user must have read permission for *file* to execute the command successfully.

EXAMPLES

None.

FUTURE DIRECTIONS

None.

SEE ALSO

None.

CHANGE HISTORY

None.

5.3.2 cddevsuppl command

NAME

cddevsuppl - set and get the major and minor numbers of a device file

SYNOPSIS

cddevsuppl [-m mapfile | -u unmapfile] [-c]

DESCRIPTION

This command is used to map and unmap the major and minor numbers of a device file on a mounted CD-ROM.

If *cddevsuppl* is executed without the **-m** or **-u** option, it lists the current device file mappings on the system.

The **-m mapfile** and **-u unmapfile** options are mutually exclusive.

OPTIONS

The following options are available:

-m mapfile This option will map the major and minor numbers for device files. The mappings are specified in *mapfile*. This file has one entry for each device file mapping in the format:

device_file_path new_major new_minor

The fields are separated by white space. The entries are separated by newlines. Anything beyond the third field shall be treated as a comment.

The maximum number of mappings is defined in the header file *<sys/cdrom.h>*. A previous device file mapping for a specific device file is overridden if that device file that is mapped again.

-u unmapfile This option will unmap the major and minor numbers for device files. The mappings are specified in *unmapfile*. This file has one entry for each device file mapping in the format:

device_file_path

The entries are separated by newlines. Anything beyond the first field shall be treated as a comment.

-c This option is only useful when used in combination with the **-m mapfile** or **-u unmapfile** options. The **-c** option will cause *cddevsuppl* to continue processing the device file mappings if an error is returned for a specific device file mapping. An error message for that specific device file will be printed to standard error. The default action is to stop processing when an

error has occurred.

OPERAND
None.

STDIN
Not used.

INPUT FILES
The input files are text files.

ENVIRONMENT VARIABLES
No environment variables affect the execution of *cddevsuppl*. Note that LC_CTYPE will not be used in filename conversion.

STDOUT
If no options are used the current device file mappings are listed on standard output. In the case of **-m mapfile**, the new setting is listed if the mapping was completed successfully. In the case of **-u unmapfile**, the device file and the major/minor numbers as recorded on the CD-ROM are listed if the unmapping was completed successfully.

STDERR
Used only for diagnostic messages.

OUTPUT FILES
None.

EXIT STATUS
The following exit values are returned:

0	successful completion
1	file not found or file is not a file or directory within a CD-ROM file hierarchy or access denied
2	not user with appropriate privileges
3	too many mappings
4	parameter error or bad format in a mapping file
5	file is not a device file
6	file not previously mapped

CONSEQUENCES OF ERRORS
None.

APPLICATION USAGE
Only a user with appropriate privileges may change administrative CD-ROM features successfully. To read the current device file mappings, the user must have read permission on the device file.

Mappings should be established before affected device files are used. If the command is applied for device file mappings when device files have already been opened, the effect of this command on these

files is undefined.

The device file mappings for a mounted CD-ROM are eliminated when the CD-ROM is unmounted.

EXAMPLES

None.

FUTURE DIRECTIONS

None.

SEE ALSO

None.

CHANGE HISTORY

None.

5.4 Definition of CD-ROM Specific Library Functions for Users

This sections provides manual pages which describe CD-ROM library functions for users in detail. The library routines are:

Name	Description
cd_suf()	Retrieve a System Use Field
cd_setdevmap()	Set mappings of major/minor numbers
cd_getdevmap()	Get mappings of major/minor numbers

5.4.1 cd_suf library routine

NAME

cd_suf - read System Use Field from a specified System Use Area

SYNOPSIS

```
#include <sys/cdrom.h>

int cd_suf (path, fsec, signature, index, buf, buflen)
char        *path;
int         fsec;
char        signature[2];
int         index;
char        *buf;
int         buflen;
```

DESCRIPTION

Cd_suf returns a System Use Field in the System Use Area for *path*.

Path points to a file or directory within the CD-ROM file hierarchy.

Fsec specifies the File Section of that file. The numbering starts with one. If *fsec* is set to -1, the System Use Area of the last File Section of that file is assumed.

Signature is the 2 byte signature to look for and return from the System Use Area.

Index is the occurrence of *signature* to return. If *signature* is a NULL pointer, return the *index* System Use Field starting from the beginning of the System Use Area. Otherwise, return the *index* occurrence of *signature*. The *index* number of the first System Use Field of any *signature* is one.

Buf and *buflen* are the buffer and buffer length in which to place the System Use Field.

RETURN VALUE

Cd_suf will return the number of bytes placed in *buf* if successful. *Cd_suf* will return 0 if the *signature* field is not found. In case of error, -1 is returned and *errno* is set to indicate the error.

ERRORS

The *cd_suf()* function will fail if:

[EACCESS] Search permission is denied for a component of the *path* prefix or read permission on the file or directory pointed to by *path* is denied.

[ENAMETOOLONG] The length of the *path* string exceeds {PATH_MAX} or a pathname component is longer than {NAME_MAX} while {_POSIX_NO_TRUNC} is in effect.

[ENOENT] A component of *path* does not exist or the *path* argument points to an empty string. The File Section indicated by *fsec* has no System Use Area.

[ENOTDIR] A component of the *path* prefix is not a directory.

[EFAULT] The address of *buf, signature* or *path* is invalid.

[EINVAL]	The value of *fsec, index* or *buflen* is invalid. The argument *path* points to a file/directory not within a CD-ROM file hierarchy.
[ENODEV]	The Volume containing the File Section indicated by *fsec* is not mounted.
[ENXIO]	The CD-ROM is not in the drive or a read error occurred.
[EINTR]	A signal was caught during the *cd_suff()* function.
[EMFILE]	{OPEN_MAX} file descriptors are currently open in the calling process.
[ENFILE]	The system file table is full.

SEE ALSO

<sys/cdrom.h>

5.4.2 cd_setdevmap library routine

NAME

cd_setdevmap - set mappings of major/minor numbers

SYNOPSIS

#include <sys/cdrom.h>

int cd_setdevmap (path, cmd, new_major, new_minor)
char *path;
int cmd;
int *new_major;
int *new_minor;

DESCRIPTION

This function sets or unsets (based on *cmd)* the major and minor numbers of one device file on a mounted CD-ROM. The argument *path* points to a file or directory within the CD-ROM file hierarchy.

If *cmd* is CD_SETDMAP, this function maps the *new_major* major number and the *new_minor* minor number to the device file pointed to by *path. New_major* specifies the new major number for the device file. *New_minor* specifies the new minor number for the device file. Any device file mapping for the device file *path* set with a previous invocation of *cd_setdevmap()* is overridden by this invocation of *cd_setdevmap().*

If *cmd* is CD_UNSETDMAP, this function unmaps the mapped major and minor numbers of the device file pointed to by *path.* The value of the recorded major number on the CD-ROM shall be returned in *new_major.* The value of the recorded minor number on the CD-ROM shall be returned in *new_minor.*

See **Section 1.1, Mapping Device Files** for further information.

RETURN VALUE

For CD_SETDMAP, *cd_setdevmap* will return one if the device file is successfully mapped (a return value of zero means no more mappings allowed).

For CD_UNSETDMAP, *cd_setdevmap* will return one if the device file is successfully unmapped (a return value of zero means mapping not found).

In case of error, -1 is returned and *errno* is set to indicate the error.

ERRORS

[EACCESS] Search permission is denied for a component of the *path* prefix or read permission on the device file pointed to by *path* is denied.

[ENAMETOOLONG] The length of the *path* string exceeds {PATH_MAX} or a pathname component is longer than {NAME_MAX} while {_POSIX_NO_TRUNC} is in effect.

[ENOENT] A component of *path* does not exist or the *path* argument points to an empty string.

[ENOTDIR]	A component of the *path* prefix is not a directory.
[EFAULT]	The address of *path, new_major,* or *new_minor* is invalid.
[EINVAL]	The value of *cmd* is invalid. The argument *path* points to a file/directory not within a CD-ROM file hierarchy. The file pointed to by *path* is not a device file.
[EPERM]	User does not have appropriate privileges to set/unset device file major/minor values.
[ENXIO]	The CD-ROM is not in the drive or a read error occurred.
[EINTR]	A signal was caught during the *cd_setdevmap()* function.
[EMFILE]	{OPEN_MAX} file descriptors are currently open in the calling process.
[ENFILE]	The system file table is full.

APPLICATION USAGE

The use of *cd_setdevmap()* is restricted to a user with appropriate privileges. The maximum number of mappings is defined in <*sys/cdrom.h*>. Mappings should be established before affected device files are used. If this function is applied for device files that have already been opened, the effect of this function on these files is undefined. The device file mappings for a mounted CD-ROM are eliminated when the CD-ROM is unmounted.

SEE ALSO

<sys/cdrom.h>

5.4.3 cd_getdevmap library routine

NAME

cd_getdevmap - get mappings of major/minor numbers

SYNOPSIS

#include <sys/cdrom.h>

int cd_getdevmap (path, pathlen, index, new_major, new_minor)
char *path;
int pathlen;
int index;
int *new_major;
int *new_minor;

DESCRIPTION

This function gets the major and minor numbers of one device file on a mounted CD-ROM. The argument *path* points to a file or directory within the CD-ROM file hierarchy. The argument *index* refers to the *index*'th mapped device file. Mappings can be obtained by *path* or *index*.

If *index* is zero, this function gets the mapped major and minor numbers of the device file pointed to by *path*. The value of the mapped major number shall be returned in *new_major*. The value of the mapped minor number shall be returned in *new_minor*. The value of *pathlen* is not used.

If *index* is not zero, this function gets the major and minor numbers and pathname of the *index*'th mapped device file. Numbering for *index* starts at one. The value of the mapped major number shall be returned in *new_major*. The value of the mapped minor number shall be returned in *new_minor*. The pathname of the device file shall be returned in *path*. If the length of the pathname for the device file is longer than *pathlen* the pathname returned in *path* will be truncated to *pathlen* length and will not be NULL terminated.

See **Section 1.1, Mapping Device Files** for further information.

RETURN VALUE

cd_getdevmap will return the length of pathname if the device file is successfully returned (a return value of zero means mapping not found). Note: if the pathname is truncated, the return value will be larger than *pathlen*.

In case of error, -1 is returned and *errno* is set to indicate the error.

ERRORS

[EACCESS] Search permission is denied for a component of the *path* prefix or read permission on the device file pointed to by *path* is denied.

[ENAMETOOLONG] The length of the *path* string exceeds {PATH_MAX} or a pathname component is longer than {NAME_MAX} while {_POSIX_NO_TRUNC} is in effect.

[ENOENT] A component of *path* does not exist or the *path* argument points to an empty string.

[ENOTDIR] A component of the *path* prefix is not a directory.

[EFAULT]	The address of *path, new_major,* or *new_minor* is invalid.
[EINVAL]	The value of *index* or *pathlen* is invalid. The argument *path* points to a file/directory not within a CD-ROM file hierarchy. The file pointed to by *path* is not a device file.
[ENXIO]	The CD-ROM is not in the drive or a read error occurred.
[EINTR]	A signal was caught during the *cd_getdevmap()* function.
[EMFILE]	{OPEN_MAX} file descriptors are currently open in the calling process.
[ENFILE]	The system file table is full.

APPLICATION USAGE

The maximum number of mappings is defined in <*sys/cdrom.h*>. The device file mappings for a mounted CD-ROM are undone when the CD-ROM is unmounted.

The *index* numbers from 1 to *n* (where *n* is the number of the last device file mapping) are always guaranteed to have a device file mapping associated with the number. Thus if an application wishes to successively delete all device file mappings, one at a time, it would call *cd_getdevmap()* with *index* equal to 1, and then *cd_setdevmap()* with CD_UNSETDMAP in a loop until *cd_getdevmap()* returns zero.

SEE ALSO

<sys/cdrom.h>

5.5 Header

NAME

cdrom.h - RRIP definitions

SYNOPSIS

#include <sys/cdrom.h>

DESCRIPTION

The *cdrom.h* header contains the RRIP constant definitions for the RRIP library functions. If XCDR is supported, this header file will also contain the XCDR constants and structure declarations for the XCDR library functions.

The function *cd_setdevmap()* uses the following values for the argument *cmd:*

CD_SETDMAP	Set device file mapping
CD_UNSETDMAP	Unset device file mapping

If an implementation imposes a limit on the number of device file mappings, they will be defined by the following symbolic name. The definition of this symbolic name may be omitted from *<sys/cdrom.h>* if the actual value of the limit is indeterminate but greater than the stated minimum. Applications should therefore only use this symbol in code conditionally compiled on the existence of this symbol.

Name	Minimum Acceptable Value
CD_MAXDMAP	50

5.6 Recommendations for CD-ROM Publishers

Unless the CD-ROM is targetted at a specific collection of systems, the values for major and minor numbers of device files will not identify the correct values on the receiving system. Also the range of values for device file major and minor numbers that a system can handle might be smaller than what can be recorded on the CD-ROM. When producing a CD-ROM to be used on various systems, it is recommended that the publisher number the device major and minor numbers consecutively starting at 0 and provide the System Administrator with sufficient information to map each recorded major and minor number to the appropriate values for the target system.

6. BIBLIOGRAPHY

- ISO 9660:1988 - Information Processing - Volume and file structure of CD-ROM for information interchange
- IEEE Standard Portable Operating System Interface for Computer Environments, IEEE std 1003.1-1988 (a.k.a POSIX Standard), New York, New York:IEEE
- X/Open CD-ROM Support Component (XCDR), Version 4.0 December 1990
- X/Open Portability Guide, Volume 3, XSI Supplementary Definitions, Prentice Hall, 1989
- System Use Sharing Protocol, Rock Ridge Group
- Wong, T. September 12, 1989. Extensions to the ISO 9660 CD-ROM Volume and File Structure Format to support POSIX File System Semantics.