

Product Standard

**Operating System and Languages:
COE Linux Platform Government-Supplied Kernel Source**

The Open Group

Copyright © February 2004, The Open Group

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owners.

Java[®] is a registered trademark of Sun Microsystems, Inc.

Linux[®] is a registered trademark of Linus Torvalds.

Boundaryless Information Flow is a trademark and UNIX and The Open Group are registered trademarks of The Open Group in the United States and other countries.

All other trademarks are the property of their respective owners.

Product Standard

Operating System and Languages: COE Linux Platform Government-Supplied Kernel Source

Document Number: X04CQ

Published in the U.K. by The Open Group, February 2004.

Any comments relating to the material contained in this document may be submitted to:

The Open Group
Apex Plaza
Forbury Road
Reading
Berkshire, RG1 1AX
United Kingdom

or by Electronic Mail to:

OGSpecs@opengroup.org

Product Standard

NAME

COE Linux Platform Government-Supplied Kernel Source

LABEL FOR LOGO

No label.

DESCRIPTION

The COE Linux Platform Government-Supplied Kernel Source (GSKS) Product Standard describes a common set of functionality that includes:

- Print Services
- System Management Services
- Accounts and Profile Manager (APM)
- Segment Installer
- Runtime Tools
- Developer Tools
- Audit Log File Manager
- Application Program Interface

The availability of COE Platform implementations providing this functionality assures that the Human-Computer Interfaces are functionally identical across multiple applications platforms. This reduces training costs and potential operator error.

CONFORMANCE REQUIREMENTS

Conformance includes the ability of the COE Platform implementation to build the GSKS code from source with no unauthorized modifications. The following requirements apply to authorized modifications:

- All changes to the Government-Supplied Kernel Source (GSKS) source code and/or objects shall comply with the appropriate standards; for example, ISO C, POSIX shell command language, or Java, as appropriate.
- All such changes shall require an approved Problem Report.

A COE Platform implementation shall be in conformance with the following.

Human-Computer Interface

1. User Interface

GNOME Human Interface Guidelines, Version 1.0, available at: <http://developer.gnome.org/projects/gup/hig/1.0>.

2. Print Services

A COE Platform implementation shall demonstrate the ability to print ASCII text and postscript graphics to both a locally attached printer (if a device is supported) and a printer attached directly to the network. The following functions are demonstrated:

- Attach a local printer
- Add a locally attached printer to the COE Platform implementation from the GUI
- Print text and graphics from the command line
- Add a network printer from the GUI
- Print text and graphics to the network printer
- Delete local and network printers
- Detach a printer from the COE Platform implementation

3. System Management Services

A COE Platform implementation shall support the COE Platform System Management Services providing the following functions:

- Network management including hosts, DNS administration, system name/IP address update, and routing administration
- Disk management
- System shutdown/reboot

4. Accounts and Profile Manager (APM)

The COE Platform Accounts and Profile Manager shall be provided on COE Platform implementations both on the Client and on the Master side. This software will:

- Set profile configurations
- Create and edit local and global user profiles
- Create and edit local and global user accounts
- Manage features and assignment of features to profiles
- Manage systems that are part of an APM administrative domain

5. Segment Installer

The COE Platform Segment Installer (Java) shall be provided on a COE Platform implementation (as Government Furnished Software). This is designed to install all COE segments (applications). It installs segments from disk, CD, tape, or the network on both local and remote machines. As an option, RPM format may be supported.

6. Runtime Tools

The system administrator uses the Runtime Tools support to install, configure, and de-install systems. The tools also provide the developers with a means to communicate with the operator during segment installation. The following runtime tools shall be supported on a COE Platform implementation:

COE_add_segment_features

Adds one or more features to a segment installed on the system.

COEAskUser

Displays a message to the user, and has the user click on a button (Yes/No, True/False, Accept/Cancel, and so on) in response to the message.

COE_feature_enabled

Determines whether a specified segment feature is currently enabled in the user's current login session.

COEFindSeg

Returns information about a requested segment. The tool sets status and writes the pathname, segment name, segment prefix, and segment type information to *stdout*.

COEGetProcessGroup

Returns the current setting of a process group.

COE_get_features

Returns the list of features assigned to a profile.

COEInstaller

Displays a list of variants or segments that may be installed from tape, disk, or other electronic media. It is normally executed by an operator who selects it from a System Administrator menu to install or de-install segments.

COEInstError

Displays an error message to the user from within a Pre-Install, Post-Install, or De-Install script signalling installation termination or de-installation of the segment.

COEListSegments

Displays a list of segments that are installed on the system.

COEListSegs

Reads and outputs information on installed segments from a computer.

COEMsg

Displays a message to the user and has the user click on the "OK" button to continue. The tool may be used by the Pre-Install, Post-Install, and De-Install scripts.

COEPrompt

Displays a message to the user and has the user enter a response to the message. The tool may be used by the Pre-Install, Post-Install, and De-Install scripts.

COEPromptPasswd

Prompts the user to enter a password. The tool may be used by the Pre-Install, Post-Install, and De-Install scripts.

COERegisterInterfaceEngine

Prompts the user to enter a password.

COESetProcessGroup

Changes the current setting of a process group.

COESegInstall Allows users to install a segment that already exists on disk without asking the user for input during installation. The **COESegInstall** API is essentially a stand-alone binary that can be executed from the command line, from a shell script, or from within an executable program. The program was developed to allow users to install a segment that already exists on disk without asking the user for input during installation. The API call was implemented as a binary to allow programmers to use the function in any type of program (for example, shell script, binary executable, or stand-alone tool) rather than tying it specifically to a limited set of C or X Window System library calls.

COEUpdateHome

Updates the *HOME* environment variable within a script file to point to where a segment was actually installed.

7. Developer Tools

COE Developer Tools support application software development and delivery, but are not delivered to operational sites. All interfaces to these tools are at the command line; none of them have a GUI interface. The following developer tools shall be supported on a COE Platform implementation:

CalcSpace Computes the space required for the segment specified and updates the hardware descriptor accordingly. The segment referred to must not be compressed and must not contain any files that do not belong with the segment (for example, source code) at runtime. The amount of space required is written to *stdout* in K bytes.

CanInstall Tests a segment to see whether it can be installed, which means that all required segments must already be on the disk, and the disk cannot have any conflicting segments.

ConvertSeg Examines segment descriptors and converts them to the latest format. The original segment descriptor directory is not modified. The output is in a directory created by the tool and called **SegDescrip.NEW**. This directory will be located directly underneath the segment's home directory at the same level as **SegDescrip**. **ConvertSeg** is not location-sensitive and may be moved to any directory desired for development.

MakeAttribs Creates the descriptor file **FileAttribs**. It recursively traverses every subdirectory beneath the segment home directory and creates a file containing permits, owner, group, and filename information.

MakeInstall Writes one or more segments to an installation medium, or packages the segments for distribution over the SIPRNET. **MakeInstall** checks to see whether **VerifySeg** has been run successfully on each of the segments, and aborts with an error if it has not.

TestInstall Temporarily installs a segment that already resides on disk. There must be no other COE process running when **TestInstall** is run. The reason for this restriction is that the tool may modify COE files already in use with unpredictable results.

TestRemove Removes a segment that was installed by **TestInstall**. There must be no other COE process running when **TestRemove** is run. The reason for this restriction is that the tool may modify COE files already in use with

unpredictable results.

TimeStamp Puts the current time and date into the VERSION descriptor.

VerifySeg Validates that a segment conforms to the rules for defining a segment. It uses information in the **SegDescrip** subdirectory and must be run whenever the segment is modified.

8. Audit Log File Manager

The COE Audit Log File Manager shall be available on a COE Platform implementation. It exercises the following functions:

- Default configuration settings
- Sample audit log files
- Disk usage parameters
- Verify monitoring of log files and system audit logs
- Display events

Portability Interface

1. Application Program Interface

A COE Platform implementation shall demonstrate conformance for the following APIs as described in the COE Programmer's Guide and Reference Manual (PGRM) for Kernel:¹

- COE Kernel APIs
- COE User Profiles APIs
- User Data APIs
- Profile Data APIs
- User/Profile Data APIs
- Application Data APIs
- Profile/Application Data APIs
- Current Profile Selection APIs
- Profile Locking APIs
- Miscellaneous APIs
- Common Data Store APIs
- COE Java Feature APIs

2. The system shall include the COE-provided standard Java runtime tools, class libraries, or jar files.

1. COE Programmer's Guide and Reference Manual (PGRM) for Kernel, Version 4.2.0.0, 4 February 2000, CM 30593.

Programming Language Environment

Not applicable.

Interoperability

Not applicable.

OPERATIONAL ENVIRONMENT

Not applicable.

PORTABILITY ENVIRONMENT

Java Support

A conforming system provides a set of services that permit the execution of pre-compiled applications that use the Java Runtime Environment (JRE) 1.2 or later.

OVERRIDING STANDARDS

Not applicable.

INDICATORS OF COMPLIANCE

The following are the required Indicators of Compliance:

- A report from the Validation Host Initial Validation Procedure
- A report from the Candidate Platform Initial Validation Procedure
- A report from the Kernel Overview Validation Procedure
- A report from the Print Services Validation Procedure
- A report from the Accounts and Profile Manager Validation Procedure
- A report from the Segment Installation Validation Procedure
- A report from the Remote Installation Validation Procedure
- A report from the Developers Toolkit and Runtime Validation Procedure
- A report from the Audit Log File Manager Validation Procedure

MIGRATION

Not applicable.