CDE Release Notes

CDE 2.1.0 Release Notes

- Introduction
- Unloading the Distribution
- Building CDE
- Installing CDE
- X Print Services
- Documentation
- Building Tests
- Installing Tests
- Running Tests
- Compatibility
- Pluggable Authentication Module (PAM)
- New Functionality
- Reference Platforms
- Open/Closed Defect Lists
- Additional Note

CDE 2.1.0 Release Notes

December 1996

Legal Notices

© Copyright 1993, 1994, 1995, 1996 Hewlett-Packard Company
© Copyright 1993, 1994, 1995, 1996 International Business Machines Corp.
© Copyright 1993, 1994, 1995, 1996 Sun Microsystems, Inc.
© Copyright 1993, 1994, 1995, 1996 Novell, Inc.
© Copyright 1995, 1996 Digital Equipment Corp.
© Copyright 1995, 1996 Fujitsu Limited
© Copyright 1995, 1996 Hitachi, Ltd.

ALL RIGHTS RESERVED

All marks are trademarks or registered trademarks of their respective owners.

Introduction

This document describes the Common Desktop Environment, CDE version 2.1.0 and Motif version 2.1.0. Wherever "CDE" is used in this document, it means CDE 2.1.0. Wherever "Motif" is used, it means Motif 2.1.0.

The release of X11 provided with this CDE distribution is based on X11R6.2, which includes additional support for Vertical Writing, User−Defined Characters, and the X Print service.

This document and accompanying text files contain important information. Please read them carefully before starting the build and installation procedures.

The command line instructions provided here are for **csh**. They may vary depending on the particular shell you are using.

Unloading the Distribution

If you received this distribution on physical media (tape or CD) rather than electronically, the distribution includes appropriate instructions in hardcopy form. The hardcopy information pertains to your particular distribution format.

The instructions in this document can be used for unloading from electronic distribution or tape.

Distribution Format

The distribution consists of a single tape that contains nine tar files. These files contain the source code, test code, and documentation of CDE, Motif, and X11R6.2

To unpack the distribution, create a directory to hold all of the sources and untar everything into that directory:

```
% mkdir $CDESOURCEDIR
% cd $CDESOURCEDIR
% tar xf tar-file-name-or-tape-device (nine times)
```

where $CDESOURCEDIR designates the CDE master source directory. If you are restoring from a tape, you should choose a device name that will not cause the tape to rewind between **tar** commands.

The directories stored in the nine sections are:

```
$CDESOURCEDIR/cde and $CDESOURCEDIR/cde-contrib
$CDESOURCEDIR/cde-test
$CDESOURCEDIR/cde/doc
$CDESOURCEDIR/motif
$CDESOURCEDIR/motif/tests
$CDESOURCEDIR/motif/doc
$CDESOURCEDIR/x11/xc
$CDESOURCEDIR/x11/xc/test
$CDESOURCEDIR/x11/xc/doc
```

If you unpack the entire distribution, you will have the following five subdirectories under $CDESOURCEDIR:

```
cde
cde-contrib
cde-test
motif
x11
```

Note that directory permissions stored in the tar file are generally set world−writable. Make sure your **umask** is set the way you want before unpacking the distribution.

The `cde-contrib` directory contains some unsupported contributed CDE software.

The DocBook SGML source for these release notes can be found in the file

`$CDESOURCEDIR/cde/ReleaseNotes.sgm`. The PostScript output can be found in `$CDESOURCEDIR/cde/ReleaseNotes.ps`.

Disk Space Requirements

The following amounts of disk space are required to unpack the various parts of the CDE distribution. All amounts are in MB.

```
cde, cde-contrib    93
cde-test          1361
cde/doc             91
motif               19
motif/tests         39
motif/doc           15
x11/xc              90
x11/xc/test         13
x11/xc/doc          39
```

Approximately 1Gb of additional disk space is required to build the release.

Depending on the platform, approximately 140MB to 160MB of additional disk space is required to install the release (C locale only). Sample systems that were installed with several locales required between 170MB and 230MB.

Building CDE

This chapter provides a generic description of the build process for CDE. The x11, motif, and cde libraries and headers must be installed on your system prior to building cde−test. These libraries and headers are included in the CDE Distribution.

CDE has been built on the following reference platforms:

```
Digital   AlphaStation 200    Digital UNIX V4.0

Fujitsu   DS/90 7000          UXP/DS V20L10

HP        HP9000/7xx          HP-UX 10.01

IBM       RS/6000             AIX 4.2

Novell    Intel 486/Pentium   UnixWare 2.02

Sun       SPARCstation        Solaris 2.4
```

For a detailed definition of the reference platforms, see "Reference Platforms."

To build CDE, you must have an ANSI C compiler and a C++ compiler. You must also have **perl** version 4.0 or higher for all platforms (except SUN, which needs version 5.0 or higher). **perl** is publicly available via anonymous **ftp** at ftp.uu.net. **perl** must be installed on your system and available in your $PATH. You must not have `/usr/ucb` in your $PATH, or the build may fail.

If you expect to build the distribution on more than one machine using a shared source tree, or if you just want to keep the binaries out of the source tree, you may want to use the X11 program **lndir** to create a symbolic link tree on each build machine. The links require additional disk space but take less disk space than multiple copies of the source tree.

Create a build directory that is not a subdirectory of the CDE master source directory:

```
% mkdir $CDEBUILDDIR
```

Then, create a symbolic link tree:

```
% lndir $CDESOURCEDIR $CDEBUILDDIR
```

where $CDESOURCEDIR is the CDE master source directory. Note that the directory containing the **lndir** program must be in your $PATH.

Note that the builds described below may not be done in parallel; each subsystem depends upon the results of builds in previous subsystems.

Building X11

1. Customize `$CDEBUILDDIR/x11/xc/config/cf/site.def` to reflect local preferences for install locations, compilers, paths, etc.

2.Federal law prohibits the X Consortium from shipping Data Encryption Standard (DES) source code outside the United States. If you have a copy of this source (`x11/xc/lib/Xdmcp/Wraphelp.c`) and wish to allow use of the **XDM −AUTHORIZATION−1** protocol, enable **HasXdmAuth** in your `site.def` file.

3.Set your current directory as follows:

```
% cd $CDEBUILDDIR/x11/xc
```

4.See if there is a BootstrapCFlags mentioned in the comments in the `<vendor>.cf` file. This file is in the directory `$CDEBUILDDIR/x11/xc/config/cf`, and the <vendor> string identifies the platform on which you are building. If there isn't a definition for BootstrapCFlags, enter:

```
% make World >& world.log
```

If there is a BootstrapCFlags definition, take its value and enter:

```
% make World BOOTSTRAPCFLAGS="value" >& world.log
```

Building Motif

1.Customize `$CDEBUILDDIR/motif/config/cf/site.def` to reflect local preferences for install locations, compilers, paths, etc.

2.If you are building with imports (the default setup in `site.def`), construct the necessary links, as follows:

```
% cd $CDEBUILDDIR/motif
% mkdir imports
% cd imports
% ln −s ../../x11/xc/exports x11
```

3.Set your current directory as follows:

```
% cd $CDEBUILDDIR/motif
```

4.See if there is a BootstrapCFlags mentioned in the comments in the `<vendor>.cf` file. This file is in the directory `$CDEBUILDDIR/motif/config/cf`. If there isn't a definition for BootstrapCFlags, enter:

```
% make World >& world.log
```

If there is a BootstrapCFlags definition, take its value and enter:

```
% make World BOOTSTRAPCFLAGS="value" >& world.log
```

Building CDE

1.Customize `$CDEBUILDDIR/cde/config/cf/site.def` to reflect local preferences for install locations, compilers, paths, etc.

2.If you are building with imports (the default setup in `site.def`), construct the necessary links, as follows:

```
% cd $CDEBUILDDIR/cde
% mkdir imports
% cd imports
% ln −s ../../motif/imports/x11 x11
% ln −s ../../motif/exports motif
```

3.Set your current directory as follows:

```
% cd $CDEBUILDDIR/motif
```

4. See if there is a BootstrapCFlags mentioned in the comments in the `<vendor>.cf` file. This file is in the directory `$CDEBUILDDIR/cde/config/cf`. If there isn't a definition for BootstrapCFlags, enter:

```
% make -k World >& world.log
```

   If there is a BootstrapCFlags definition, take its value and enter:

```
% make -k World BOOTSTRAPCFLAGS="value" >& world.log
```

On Solaris, documents build with a large number of SGML errors that can be ignored, as a result of using **sgmls** rather than **nsgmls**.

Installing CDE

CDE should be installed or re−installed in single−user mode. It may not be sufficient merely to stop the CDE−related processes on a system that already has a CDE version installed.

Before installing CDE, check to see if `/usr/dt` exists. If this directory or link already exists and you want a completely new installation (which is recommended), then rename it or back it up and then remove it.

HP−UX only:

If CDE is running, use **init**(1M) to change the runlevel to a level that will stop **dtlogin** and other CDE processes (see `/etc/inittab` for more information). For example:

```
% /sbin/init 2
```

AIX only:

If "lslpp −h | grep X11.Dt" shows an AIX installation of CDE, use **smit** to remove the filesets. Otherwise, if a tar installation of CDE was made in `/usr/dt`, remove or rename it.

If CDE is running, do the following to stop all of the CDE processes:
Kill all running **dtlogin** processes
Kill the X server

The installation process creates directory hierarchies rooted at `/etc/dt` and `/var/dt`, but does not install any files there. Delete files in these directories with care. Important customizations made by applications are stored here, and this information may be "unrecoverable" if removed.

Installing X11

To install X do the following as root:

```
% cd $CDEBUILDDIR/x11/xc
% make install >& install.log
```

You can install manual pages with:

```
% make install.man >& man.log
```

Installing Motif

To install Motif do the following as root:

```
% cd $CDEBUILDDIR/motif
% make install.cde >& install.log
```

Note this is NOT the same command used to install Motif without CDE.

Installing CDE

To install CDE do the following as root (since **inetd** is affected, you should do this at the console, rather than remotely):

```
% cd $CDEBUILDDIR/cde
% ./admin/IntegTools/dbTools/installCDE -s $CDEBUILDDIR/cde
```

where $CDEBUILDDIR is the path to the directory where CDE was built. This installs CDE into `/usr/dt/`.

You must reboot the system after installation.

This script requires the user to have root permissions.

To print the usage message for the script, run:

```
% installCDE '-?'
```

Note that many of the options for **installCDE** have not been tested.

You must install into `/usr/dt`. You can use the *-pI* option to specify an alternative directory for the actual file storage, but `/usr/dt` will be created as a symbolic link to that directory because CDE will not operate without that link.

To start CDE manually, run the following command as user root:

```
% /usr/dt/bin/dtlogin
```

Installing Localized Filesets

After the base install of CDE, you can install localized filesets. Each locale has three filesets: message catalogs, resource and config files, and on-line help. The filesets for the C locale are installed by default. This section describes how to install filesets for supported locales other than C.

To install the localized filesets for a locale, run **installCDE** as root as follows (the quotes are required):

```
% $CDEBUILDDIR/cde/admin/IntegTools/dbTools/installCDE \
        -s $CDEBUILDDIR/cde \
        -f "CDE-<LOCALE> CDE-MSG-<LOCALE> CDE-HELP-<LOCALE>"
```

where "<LOCALE>" is "DE" (German), "ES" (Spanish), "FR" (French), "IT" (Italian), or "JP" (Japanese). For example, to install Japanese filesets:

```
% $CDEBUILDDIR/cde/admin/IntegTools/dbTools/installCDE \
        -s $CDEBUILDDIR/cde \
        -f "CDE-JP CDE-MSG-JP CDE-HELP-JP"
```

Known Installation Problems

Some known installation problems are listed below (additional information may be found in the

files `$CDESOURCEDIR/cde/OPENBUGS` and `$CDESOURCEDIR/motif/OPENBUGS`):

- This problem occurs on Digital only.
  Deconfiguration will attempt to undo the installation operations. If it fails or needs to be undone manually, you can restore the non−CDE login with:

  ```
  # ln −f −s ../init.d/xdm /sbin/rc3.d/S95xdm
  ```

  Rebooting will return the workstation to xdm mode.
- If the Japanese filesets are installed, several files are reported as missing during the installation process. The Japanese translations of some documents are not part of the initial release of CDE 2.1. This is not harmful to the installation.
- On Solaris, the file `/proj/cde/programs/nsgmls/nsgmls` is reported as missing. This is expected.

X Print Services

To print with CDE, a print server must be running, and the proper name of the server must be supplied to client applications.

In addition, to enable users to select a printer setup dialog from within client applications, a print dialog manager (dtpdm) daemon must be running.

Running an X Print Server

Running an X Print Server is identical to running any other X server.

Here is a sample command line for use with a typical configuration:

```
% Xprt :1 -ac -fp /usr/lib/X11/fonts/misc,\
/usr/lib/X11/fonts/Type1,/usr/lib/X11/fonts/100dpi,\
/usr/dt/config/xfonts
```

The following options are available when running the print server:

:1

Use a different display from the default.

−ac

Disable access control, since no simple mechanism for sharing keys is provided.

−fp

Point to the standard video fonts. Note that the only reason for doing this is to add the CDE aliases to the default set.

In addition, the following new arguments are available:

−XpFile

Points to file listing of available printers instead of default.

XPCONFIGDIR

Environment variable specifying alternative location of configuration files.

The server can be run within a user's session as an individual command or at boot time. These two methods of invocation represent two distinct styles. The latter style provides a centralized departmental service, while the former style usually serves a single user. However, because an X print server is no different from any other X server, there are many other configuration possibilities available.

Informing Clients

To select an X Print Server from a client that employs the Xprint server, simply type the name of the printer like this:

```
printerName@host:display
```

A default printer can be specified by providing a value for either the XpPrinter resource or the **XPRINTER** environment variable.

An alternative, and probably the simplest, way to provide an entire set of printers to all clients is to set the **XPSERVERLIST** variable (or XpServerList resource) to the name of a X Print Server (for example, print_host:1). The user's default value of **LPDEST** will be combined with the defined value to create a full printer name, allowing the user to use the GUI on the print setup box dialog to access other printers.

You can set all of the above environment variables in the `.dtprofile` file in a user's home directory.

Print Dialog Manager

In order for users to access the print setup dialogs, a special client called the Print Dialog Manager must be running. The name of the daemon that implements the Print Dialog Manager is **dtpdmd**. You can invoke **dtpdmd** in either of two ways:

- Point it at a print server, as follows:

```
dtpdmd -d print_host:1 &
```

- Point it at your video display, as follows:

```
dtpdmd &
```

Note that in this case, the environment variable **XPDMDISPLAY** must be set to the value *video*.

You can apply either of these methods using the `.dtprofile` file in the user's home directory.

Documentation

The documentation available with this release consists of:

- CDE and Motif man pages
- CDE and Motif guides
- CDE online help

Man Pages

This release provides English versions of the CDE and Motif man pages in traditional man format. The man pages are generated from DocBook source during the build process. Man pages are also viewable online with **dtinfo** using the English CDE information library.

Guides

The guides provided with this release include:

- English CDE guides, which have been converted to DocBook. These guides are viewable online with **dtinfo** using the English CDE information library. The DocBook source is located in `$CDEBUILDDIR/cde/doc/C/guides`. New volumes in this release are the Information Manager User's Guide, Information System Author's and Programmer's Guide, and Guide to the DocBook DTD.
- English Motif guides, which have been converted to DocBook. These guides are viewable online with **dtinfo** using the English CDE information library. The DocBook source is located in `$CDEBUILDDIR/motif/doc/C/guides`.
  With this release, the Motif and CDE style guides have been combined and reworked into three new volumes: Motif and CDE Style Guide, Motif and CDE Style Guide Reference, and Motif and CDE Style Guide Certification Checklist.
- Translations of selected CDE and Motif guides, which have been converted to DocBook. The CDE User's Guide, Information Manager User's Guide, and Advanced User's and System Administrator's Guide are available in all languages. Additional guides are available in Japanese. These guides are viewable online with dtinfo using the localized CDE information library. The DocBook source is located in `$CDEBUILDDIR/cde/doc/LANG/guides`, where LANG corresponds to one of the following locales:

  de_DE.ISO8859−1 (German)

  es_ES.ISO8859−1 (Spanish)

  fr_FR.ISO8859−1 (French)

  it_IT.ISO8859−1 (Italian)

  ja_JP.dt−eucJP (Japanese)

Online Help

This release contains complete English and translated CDE online help, which has been converted to DocBook. The online help is viewable with the Help Manager. New material includes online help for dtinfo and for printing. The English online help source can be found in `$CDEBUILDDIR/cde/doc/C/help`. The translated online help can be found in `$CDEBUILDDIR/cde/doc/LANG/help`, where LANG corresponds to the locales listed in the "Guides" section.

Building Tests

This chapter provides a generic description of the build process for the CDE Test Suite.

You must build and install CDE (including X11 and Motif) on your system prior to building these tests. Also, the actual CDE build tree must still be accessible.

The CDE test suite has been built on the following reference platforms:

```
Digital  AlphaStation 200   Digital UNIX V4.0

Fujitsu  DS/90 7000         UXP/DS V20L10

HP       HP9000/7xx         HP-UX 10.01

IBM      RS/6000            AIX 4.2

Novell   Intel 486/Pentium  UnixWare 2.02

Sun      SPARCstation       Solaris 2.4
```

For a detailed definition of the reference platforms, see "Reference Platforms."

Configure the CDE Test Suite Build Environment

Before building, you must make sure that symbolic links for the `OSVersion.tmpl` and the `localtree.tmpl` files are created and that these links are correct for your system. You must also modify the `site.cf` file. You can not skip this step, even though you have built CDE for your system, because the config directory for the CDE Test Suite is separate from the CDE build directory.

Builds have been performed using the optimized shared library templates for `localtree.tmpl`.

To create the links, go to the CDE Test config directory in the build directory:

```
cd $CDEBUILDDIR/cde-test/config
```

Make sure that cdesrcRoot is defined to point to `$CDEBUILDDIR/cde` in the file `site.cf`. For example, if $CDEBUILDDIR is `/proj`, the following lines should appear in the file (do not use environment variables here):

```
#ifndef cdesrcRoot
#define cdesrcRoot /proj/cde
#endif
```

Create the correct links for your system. For example, if your system is HP–UX 10.01 you will need to execute the following commands to create these links:

```
% ln -s OSVersiontemplates/10.01 OSVersion.tmpl
% ln -s localtemplates/hpux/10.01/optimized.shared localtree.tmpl
```

The following are the correct links for the other platforms:

Digital Unix 4.0

```
OSVersion.tmpl -> OSVersiontemplates/4.0
localtree.tmpl -> localtemplates/dec/alpha/osf/optimized.sharedlibs
```

Fujitsu UXP/DS V20L10

```
OSVersion.tmpl -> OSVersiontemplates/20.10
localtree.tmpl -> localtemplates/uxp/optimized.sharedlibs
```

IBM AIX 4.2

```
OSVersion.tmpl -> OSVersiontemplates/4.2
localtree.tmpl -> localtemplates/ibm/rs6000/aix4.2/optimized.sharedlibs
```

Sun Solaris 2.4

```
OSVersion.tmpl -> OSVersiontemplates/5.4
localtree.tmpl -> localtemplates/sun/optimized.sharedlibs
```

UNIXWARE

```
OSVersion.tmpl -> OSVersiontemplates/4.2
localtree.tmpl -> localtemplates/sun/optimized.sharedlibs
```

Building the CDE Test Suites

Change to the directory containing the **test_build** program:

```
% cd $CDEBUILDDIR/cde-test/admin/BuildTools/master_build
```

Run **test_build** to build the CDE Test Suite on your system:

```
% ./test_build -build $CDEBUILDDIR/cde-test \
        -log $CDEBUILDDIR/cde-test/logs -c
```

The **test_build** command makes the Makefiles, includes, depend, and all targets in the specified build directory and creates build logs in the specified log directory. For a more complete description of the **test_build** options, see the **master_build**.1 man page in `$CDEBUILDDIR/cde/admin/man/man1` (**test_build** and **master_build** support the same set of options).

Installing Tests

Test execution is fully functional inside the cde−test directory structure. No installation is required.

Running Tests

The following sections give basic instructions for executing the various testing protocols and information on test results.

Test Execution

Automated Regression Tests

The test execution cycle is divided into five phases based on dependencies. The phases are listed below in order of execution. The cycle outlined here is intended to accommodate the testing of multiple test suites automatically.

1.Configure test environment.

The test environment requires a complete installation of CDE, a complete test build that includes TET, and a default user environment with the following X resources set:

```
*enableButtonTab: False
*blinkRate: 0
dtterm*blinkRate: 250
```

The tests can be run either from the test display itself or from a remote display. The following environment variables must be set:

```
TET_ROOT=$CDEBUILDDIR/cde-test
TET_TMP_DIR=<temporary space dir>
DISPLAY=<test display>
PATH=$PATH:.
```

Then, general configuration data must be loaded:

```
$ . $TET_ROOT/DTConfig/`uname -s`/dt_env.sh -l
```

2.Configure the component–specific test environment.

Component–specific test environment configuration is set by:

```
$ cd $TET_ROOT/comp_suites/<test suite name>
$ . config/`uname -s`/env.sh -l
```

where <test suite name> corresponds to the directory under `$TET_ROOT/comp_suites` for a CDE component (for example, dtfile, DtSvc/dragdrop, dthelp/dtHelpApi).

Some test suites require additional setup that will have to be done manually. The instructions for additional setup (if any) can be found in

```
$TET_ROOT/comp_suites/<test suite name>/README[.test suite name]
```

3.Execute test suite.

Tests are organized under "scenarios" (e.g. REGRESSION, all, TOUCH) which are listed in the file

```
$TET_ROOT/comp_suites/<test suite name>/tet_scen
```

The tests are run through TET's **tcc** (test case controller):

```
$ tcc -i <output dir> -j <journal file> \
      -e comp_suites/<test suite name> <scenario>
```

where <journal file> is the path name of the file where test journal reports and results are stored. Any test data for inspection (for example, images) is stored under <output dir>.

4. Clean up test suite.

   Component−specific test environment configuration cleanup is accomplished by:

```
$ cd $TET_ROOT/comp_suites/<test suite name>
$ . config/`uname −s`/env.sh −c
```

5. Clean up general test environment.

   General test environment configuration cleanup is accomplished by:

```
$ . $TET_ROOT/DTConfig/`uname −s`/dt_env.sh −c
```

Alternatively, the test execution script **testCDE**, available in `$TET_ROOT/bin`, can be used. Usage of the script is:

```
$ $TET_ROOT/bin/testCDE −tet_root $CDEBUILDDIR/cde−test \
        { −suite <test suite name> | −list <file with list of suites> }
        [ −scen <test scenario> ] \
        [ −engr <email address to notify> ] \
        [ −record ]
        [ −res <directory path name> ]
```

The −record option is used to execute relevant suites in record mode (to generate golden data). The −res option is used to specify an alternative results directory (`$HOME/test−results` is the default). When the scenario is not specified using the −scen option, the REGRESSION scenario is selected by default.

In addition to performing the configuration setup, test execution, and configuration cleanup steps (listed above for the first method of running tests), this script utilizes setup scripts (if any) found under `$TET_ROOT/setup/<test suite name>`. These setup scripts automate additional manual setup required for some test suites.

Note: In many cases these setup scripts use site specific data such as user−ids. Make sure the site−specific data is updated before test execution.

Manual Regression Tests

The following CDE components have predefined manual tests, with instructions located in the specified files. Some suites have a number of instruction files, indicated below with appropriate wildcards.

Suite	File (relative to $CDEBUILDDIR/)
DefConfig	cde−test/comp_suites/DefConfig/manual/*_tests
DtSvc/DtMsgLog	cde−test/comp_suites/DtSvc/DtMsgLog/manual/manualtests
DtSvc/DtWsm	cde−test/comp_suites/DtSvc/DtWsm/manual/ws*
DtSvc/dragdrop	cde−test/comp_suites/DtSvc/dragdrop/manual/assertion_mis
DtSvc/visuals	cde−test/comp_suites/DtSvc/visuals/manual/icons
RunTime	cde−test/comp_suites/RunTime/manual/runtime*.des
dtaction	cde−test/comp_suites/dtaction/manual/test.checklist
dtappgather	cde−test/comp_suites/dtappgather/manual/README
dtbuilder	cde−test/comp_suites/dtbuilder/manual/Instructions

dtcalc	cde-test/comp_suites/dtcalc/manual/*/*
dtcm	cde-test/comp_suites/dtcm/manual/assertions.all
dtfp	cde-test/comp_suites/dtwm/fptests/manual/*/README
dthelp	cde-test/comp_suites/dthelp/manual/DtHelpManualTest.mif
dthelp/CREInline	cde-test/comp_suites/dthelp/manual/CREInline/README
dtinfo	cde-test/comp_suites/dtinfo/manual/README
dtlp	cde-test/comp_suites/dtlp/manual/README
dtmail	cde-test/comp_suites/dtmail/manual/Assertions.manual
dtpad	cde-test/comp_suites/dtpad/manual/dtpad.manual_checklist
dtterm	cde-test/comp_suites/dtterm/manual/manual
dtwm	cde-test/comp_suites/dtwm/wmtests/manual/*/README

For each suite, simply following the instructions listed in the specified files. Any suite−specific instructional information is listed below.

dthelp

This file is in FrameMaker format.

dtmail

There are two other files in the same directory called `Mailer.assertion` and `README`, the latter of which explains which file should be used.

Motif Quality Assurance Test Suite Testing

Most of the information required to build and run the CDE/QATS can be found in the following files:

```
$CDEBUILDDIR/motif/tests/docs/Output/draft/ps/chap1
$CDEBUILDDIR/motif/tests/docs/Output/draft/ps/chap2
$CDEBUILDDIR/motif/tests/docs/Output/draft/ps/chap3
$CDEBUILDDIR/motif/tests/docs/Output/draft/ps/chap4
$CDEBUILDDIR/motif/tests/environment/README
```

While these documents describe the original OSF test suite, they are applicable with some minor modifications to CDE.

- The following environment variables need to be set for running in playback mode:
  - **VFA_TOP** to the top of the Motif tree (`$CDEBUILDDIR/motif`)
  - **VISUAL_PATH** to the parent of the directories containing the recorded images
  - **VFA_RESULTS_DIR** to the parent of the output directories (the directories must exist)
- You can run the automated test suite from the top of the test tree by invoking the RUN_auto located in `$CDEBUILDDIR/motif/tests`.
- When recording, the **VISUAL_PATH** and **VFA_RESULTS_DIR** should be the same.

The QATS must be run on a display controlled by a Motif 2.x version of **mwm**, not **dtwm**. A suitable version of **mwm** is built in the `$CDEBUILDDIR/motif/clients/mwm` directory.

Interplatform Tests

Interplatform testing is the testing of CDE application functionality between platforms. The following CDE capabilities are tested:

- The ability of CDE applications to execute and interoperate in distributed fashion across a network of heterogeneous, interoperating hosts.
- The ability of CDE users to execute the desktop on different heterogenous platforms in a consistent manner.

The general process used in the Interplatform tests is to startup CDE components from a client host and a daemon host (on the same display) using **dtaction** before running actual tests involving drag and drop on/between these components.

The tests are located in `$CDEBUILDDIR/cde-test/sys_suites/Intop/`: The information required to initially configure the test systems before a test run is in the following files in the same directory:

- `HowToRun.interplatform`
- `README.interplatform`
- `README`

The information in these files includes steps to configure the test machines to use the dtspc protocol, which is required by the tests.

Once the test machines have been correctly setup, the tests can be executed in a similar fashion to other CDE tests described earlier in this document, with the exception that these tests are located in the `sys_suites` rather than the `comp_suites` sub-directory and so the pathnames will have to be correspondingly altered.

Test Execution Results

The following sections give details on the results obtained with the various test protocols on the reference platforms.

Automated Regression Tests

The following is a partial list of expected results for the six tested platforms. The first number indicates the number of passes for the given test suite and the second number indicates the total number of assertions that were run in that test suite.

Note that not all results are available on all platforms. Defect reports have been submitted as appropriate for non-passes.

Suite	IBM	Sun	Digital	HP	Fujitsu	Unixware
CREPrinting	3/7	5/7	3/7	6/7	3/7	6/7
ConfigDB	0/1	1/1	0/1	1/1	1/1	0/1
DefConfig	3/5	3/16	3/23	3/4	2/28	3/4
Desktop		55/57	55/57	55/57	56/57	38/51
DevEnv	1/2	1/3	0/0	2/3	0/0	1/3

DtAction	97/109	63/77	70/96	103/125	101/117	43/47
DtClientPrinting_dtcm		13/26	13/26	13/26	14/26	13/26
DtClientPrinting_dtmail		21/28	21/28	20/24	22/25	17/26
DtClientPrinting_dtpad		12/14	11/14	12/14	9/10	12/14
DtEditor	32/35	41/42	35/42	41/42	17/40	30/35
DtHelp	0/0		0/0			0/6
DtInit	8/8	8/8	8/8	8/8	8/8	8/8
DtMrm	8/8	8/33				2/6
DtMsgLog	29/37	27/37	26/35	28/37	31/37	27/35
DtSvc	12/23	0/43	0/2			0/43
DtTerm	1/3	0/7				0/8
DtWidget	98/113	113/113	113/113	113/113	98/113	113/113
DtWsm	9/9	9/9	8/11	9/9	9/9	9/11
Intop	158/233					
Dtinfo printing		19/32	18/32	19/32	3/12	18/27
XmMT	142/148	138/141	49/50			83/85
XmPrinting	15/22	16/22	17/22	17/22	7/7	17/22
dragdrop	8/18	12/18	8/18	18/18	11/18	14/18
dtGui	17/34	21/34	15/21	22/34	0/15	0/21
dtHelpApi	5/104	92/104	67/75	76/104	0/59	77/104
dtHelpCanvas	1/2	2/2	1/2	1/2	2/2	2/2
dtHelpGui	52/100	103/126	27/100	58/126	57/82	51/126
dtPdm	1/3	1/5	3/5	1/5	3/5	0/5
dtaction	34/36	35/36	33/36	31/36	33/36	32/36
dtappgather	0/4	4/4	4/4	4/4	3/4	3/4
dtappintegrate	16/17	16/17	13/17	17/17	7/17	17/17
dtcalc	1/2	1/3	1/2	1/2	1/3	1/3
dtcm				2/126		
dtdatatyping	21/27	21/27	18/27	21/27	21/27	18/27
dtdocbook	0/3	2/4	2/4	2/4	2/4	2/4
dtfile	10/228	82/229	30/123	8/226	9/218	6/77
dtfp	17/23	20/21	2/21	10/21	7/13	15/20
dthelpgen	18/23	20/23	20/23	0/19	19/23	13/19
dthelpprint	2/10	2/10	6/10	0/10	7/10	7/10
dthelptag	9/9	4/9	9/9	4/9	7/9	9/9
dthelpview	20/42	29/42	42/42	33/42	32/42	33/42
dtksh	43/302	285/302	288/302	285/302	297/302	200/302
dtlp	13/14	13/14	12/14	13/14	12/14	12/14
dtmail	60/78	61/82	61/77	67/78	71/78	65/77
dtpad	128/190	141/157	139/157	141/157	128/145	138/153
dtscreen	49/55	53/55	31/51	53/55	49/55	49/55
dtsearchpath	26/26	26/26	24/26	26/26	0/26	24/26
dtsession	39/40	39/40	41/41	40/41	40/41	41/41
dtspcd	11/11			11/11	0/0	9/11
dtsr	8/12	7/12	8/12	8/12	8/12	7/12
dtstyle	10/10	10/10	10/10	10/10	0/9	8/10
dtterm	131/178	63/85	130/181	133/182	100/137	65/82
dttypes	0/7	0/7	0/0	0/7	0/0	0/7
dtwm	129/129	122/129	118/138	124/133	90/108	94/101
mtDtHelp	6/6	0/6	0/0			0/6
mtDtMrm	69/74	14/67	0/0			6/58

```
mtDtSvc                 38/43    0/43     0/2                        0/43
mtDtTerm                14/19    0/13     0/0                        0/12
mtDtWidget              101/107  6/152    6/8                        3/147
mttt                             1/2
sm-wm                            1/6               0/0
tt                      2/30     27/34    4/31     22/31    4/32     10/13
xpApi                   77/86    75/86    73/86    77/93    73/86    71/82
xpConfig                18/18    8/8      15/18    5/30     15/18    15/18
xpddx                   87/250   187/221  143/243  245/252  209/227  60/252
-------------------------------------------------------------------------
```

Because QC/Replay was not available on AIX 4.2, the QC/Replay tests were not run on the IBM platform.

Motif QATS tests

The Motif/QATS automated tests and Motif window manager were built as described in an earlier section.

Automated tests were built and run on all reference platforms. Sun, HP, and SGI displays were used for displaying tests. Motif window managers built on different platforms were used to manage the windows while running the tests. The complete QATS is executed on Sun, HPUX, and SGI platforms while only the new tests or suites were executed on the remaining reference platforms.

Missing Keys WARNINGS

The keys on the keyboards used on the HP and Sun platforms are insufficient for the needs of the tests (keys missing from the keyboard that were expected to be mapped for the tests), causing "AUTOWARNING" errors in every one of the tests. The standard report generation scripts detect these benign errors and flag the entire test as an error. Since these were not considered to be fatal errors (after inspection of the test code), the analysis scripts filtered out AUTOWARNING errors that came from missing keys at the very beginning of a test run.

New Test Clients

Several new test clients were added to the QATS during this release, some automated and some manual. Test clients were added to the following directories:

Automated:

```
tests/Toolkit/Buttons
tests/uil/widgets
```

Manual:

```
tests/Manual/DtWidgets
```

Golden data was generated where necessary.

Golden Data Generation

Some new QATS test suites, which do not have any previous recordings, were run in record mode to generate golden data. The recordings were added to this release. These new automated test suites are:

```
tests/Toolkit/ComboBox
tests/Toolkit/SpinBox
tests/Toolkit/Notebook
tests/Toolkit/IconVC
```

Test Results

The results were compared with the recordings made against Motif 2.0. The results are mostly consistent with the previous runs carried out at OSF. There are failures due to both QATS problems and Motif defects. All the errors/failures reported during the run are due to known problems in both QATS and Motif code and have defects already opened against them. Also, the results were consistent across all the platforms on which the tests were executed.

Compatibility

OSF/Motif 2.0 Compatibility

Compatibility with CDE/Motif 1.2 was given great emphasis in this release, even at the expense of compatibility with OSF/Motif 2.0. Some OSF/Motif 2.0 applications may experience problems because of the following changes:

- The **XmCSText** widget has been withdrawn, as have those APIs added to OSF/Motif 2.0 solely to support it.
- Xrm support for word−size independent `.uid` files has been removed. Existing `.uid` files compiled with Motif 2.0 **uil** may not be readable. As in OSF/Motif 1.2, `.uid` files are portable only between machines with the same word size.
- New **XmComboBox XmNpositionMode** and **XmSpinBox XmNpositionType** resources default to incompatible index values, and should be forced by all applications using these widgets. **XmONE_BASED** is recommended for **XmComboBox** widgets because it lets applications distinguish between new values entered in the text field and the first item in the list.
- The **_XmStrings** array has, on some machines, been split into multiple sub−arrays with the same techniques used by libXt. This preserves compatibility with Motif 1.2 and permits future expansion.
- In Motif 2.0 there were two distinct **XmREPLACE** constants, with different values. The **XmMergeMode** constant has been renamed **XmMERGE_REPLACE**. This is a source compatibility issue; binary compatibility is unaffected.
- A variety of changes to visual appearance may be noticed. See the Motif 2.1.0 Release Notes for details.

CDE Binary Compatability

Binary Compatibility testing was carried out using products provided by SunSoft and IBM. In addition, binary executables built locally for the CDE 1.0.10 release were tested. Some incompatibilities were found, as described below. Some problems were found that were specific to the tested application and which we do not believe indicate a compatibility problem with the reference system.

- As with previous major releases of Motif, binary compatibility is not guaranteed for application widgets that subclass Motif widgets without using **XmResolvePartOffsets** to access instance fields. Subclasses that use **XmResolvePartOffsets** (or **XmResolveAllPartOffsets**) should be unaffected.
- On AIX, the CDE shared libraries are linked with the reentrant C library, `libc_r.a`. Legacy applications that reference the non−reentrant C library, `libc.a`, will experience difficulties because two different copies of data are being referenced. The solution recommended by IBM is to build and install a complete set of CDE libraries with the non−reentrant compiler, giving them the standard names, and another complete reentrant set, giving them names with the `_r` suffix (for example, `libXm_r.a`). The default configuration files build only the reentrant set.

- On AIX, the sample application **llenhcapture** did not execute. We believe this is due to an incompatibility in standard AIX libraries.
- On AIX, the sample application **SDE** did not execute, as it calls the undocumented and withdrawn function `XmStringLoadQueryFont`.
- The DtHelp library has a dependency on the math library, `libm.a`. On SVR4 systems, any legacy application that does not already reference the math library must be re−linked.

We did not use Digital or Fujitsu platforms in the BC testing since they were not reference platforms for the CDE1.0.10 release.

Pluggable Authentication Module (PAM)

Pluggable Authentication Module (PAM) code supplied by SunSoft is included with the CDE source. The source is located in the `$CDESOURCEDIR/cde/lib/pam` subdirectory. Documentation, manual pages, examples, and a selection of modules (back−end code for some common authentication technologies) are provided in addition to the front−end pam library.

PAM is built and installed automatically on Sun platforms. The **dtlogin** source contains conditionally compiled code to invoke the PAM front−end API.

New Functionality

The following paragraphs summarize the new functionality of this release relative to CDE1.0. Please refer to the specific on−line documentation for more detailed information.

Printing

Starting with this release **dtpad**, **dtmail** and **dtcm** are using the X protocol based print server. This print server produces output in three formats: PCL, Postscript and Raster.

Thread−Safe Libraries

The following libraries are thread−safe enabled: Xm, tt, DtHelp, DtSvc, DtTerm, DtWidget, and Mrm. This means that the libraries themselves are thread safe and a multithreaded application need not do explicit locking when accessing these libraries. This is supported only for platforms that support multithreaded programming.

X11R6.2

This release is based on X11R6.2, which includes additional support for Vertical Writing, User −Defined Characters (UDCs) and the X Print Service.

In addition, session management using the X11R6 XSMP is provided and supported by **dtsession** and **dtwm**.

Internationalization

There are several new features that support the internationalization of CDE and developing applications for international markets. CDE now supports the On−The−Spot input method for **Xm** and **dtterm**. Based on the session start−up locale, users can select their preferred Input method server either at startup time (using the new Input method server selection and launch) or during the session (via **dtstyle**). They can also select their preferred input method style (Off −The−Spot, On−The−Spot, etc.) via **dtstyle**. Users may use and define their own fonts with the User Definable Character (UDC) feature. Also, Vertical Writing is now supported.

On−Line Information Access

This release provides a standard set of online documentation that covers the various CDE and Motif components. The CDE desktop includes the Information Manager which can be used to access, read, and search through this documentation.

The Information Manager replaces the the Help Manager on the Front Panel. Additional

information about using Information Manager is available on−line in the CDE: Information Manager User's Guide and in the **dtinfo** man page. Information about authoring on−line documentation for use with the Information Manager is also available on−line in the CDE: Information System Author's and Programmer's Guide and in the **dtinfogen** man page.

The Information Manager includes a general purpose text search and retrieval system. This functionality is provided by the DtSearch library and by the dtsr* programs. Applications can also directly access on−line information using the DtInfo Database Engine API provided by the DtMmdb library.

Tests

Several new test suites have been added for this release.

- Threadsafe Motif, Tooltalk and Dt libraries
- Postscript and PCL ddx
- Printing: API, PDM, XmPrinting, C/RE, and Dt clients
- Dtinfo Desktop and Printing (QC/Replay based)
- DtSearch
- Error Message API
- X11R6 session management
- Standalone Window Manager (QC/Replay based)
- Motif 2.0 changes (Motif QATS)
- dtimsstart
- I18N features (Vertical Writing, UDCs, on−the−spot IM)
- SDL Converter

Reference Platforms

This chapter defines the reference platforms for this release. A reference platform is a system configuration on which the shipped CDE source has been verified to build and execute.

Digital

Component

Model	AlphaStation 200
RAM	64 MB
Graphics	1280x1024 8 plane

Software

OS Version	Digital UNIX V4.0 (Rev. 386)
Patches to OS	none
Network Communications	NIS, NFS
Compiler(s)	DEC C V5.2-023 DEC C++ V5.4-006

Fujitsu

Component

Model	DS/90 7000
RAM	64 MB
Graphics	1280x1024 8 plane (CF1 or CF3 display adapter required)

Software

OS Version	UXP/DS V20L10
Patches to OS	PTF Y96021 and see below
Network Communications	NIS, NFS
Compiler(s)	C Development Set (CDS): CC Driver UXP/DS R2.6 C++ V12L20
X Window System Server	UXP/DS X11R6 XServer V10L10

Comments

The following patches are currently available directly from Fujitsu:

OS patch PEF TY12846 is required to repair some RPC problems. This fix will be included in V20L10 Y96111.

OS patch TY13126 may be required to fix some problems in building the release using

NFS.

The listen daemon needs a non−public patch to prevent a core dump when **rpc.ttdbserverd** is started.

A non−public patch is needed to solve an NFS locking problem seen at the X Consortium while attempting to perform CDE builds.

A non−public patch must be applied to the C++ compilation utilities (package name uxpcplus).

A non−public patch is needed to enter Japanese text; this lets the input server **xfeoak** run using X11R6.

Hewlett Packard

Component

```
Model                   HP9000/7xx
RAM                     64 MB
Graphics                Artist (HP208LC1280) 1280x1024 8 plane
```

Software

```
OS Version              HP-UX 10.01
Patches to OS           see below
Network Communications  NIS, NFS

Compiler(s)             HP C/ANSI C Developer's Bundle for HP-UX 10.0
                        HP C++ Compiler Bundle
                        Patches PHSS_9138 (compiler) and PHSS_8831 (ld)
Additional components   see below
```

Comments

The following additional patches should be applied. Patches listed are those available as of 11 December, 1996. All HP patches are subject to being superseded from time to time; all new patches list the old ones that they replace. Current replacement patches may be substituted for those listed below.

The platforms at the X Consortium have some patches that have been superseded by those listed.

Some of these patches are needed for correct operation of ClearCase release 2.0.3. They may not be necessary to build and run CDE from a locally installed tree.

```
PHCO_5803       fsck_hfs(1M) patch for possible corruption
PHKL_5812       Fix panic when opening some fix partition disks
PHKL_9114       fix for process hangs in semop(2)
PHKL_9404       JFS, LVM, UFS, pstat cumulative patch
PHNE_6419       cumulative STREAMS patch
PHNE_6527       LAN products Patch
PHNE_7514       cumulative telnet(1) patch
PHNE_7553       telnetd(1M) cumulative patch
PHNE_8114       NFS/NIS Runtime cumulative patch
```

```
PHNE_8168       ARPA Transport cumulative patch
PHNE_9089       Cumulative Mux and Pty Patch
```

The kernel parameter *maxuprc* (maximum processes per user) should be increased from its default value of 50 to 100.

IBM

Component

```
Model                   RS/6000
RAM                     64 MB
graphics                1280x1024 8 plane
```

Software

```
OS Version              AIX 4.2
Patches to OS           PTF 446072
Network Communications  NIS, NFS

Compiler(s)             xlC.C     3.1.4.0
                        xlC.C++.* 3.1.4.0 (several filesets)
                        xlC.cpp   3.1.4.0
                        xlC.rts   3.1.4.0
Additional components   See below
```

Comments

Version 3.1.4.0 of the C Programming Information filesets should also be installed.

All platforms require the standard X11 filesets. At the X Consortium, we have found that the original standard installs did not have the following required fileset:

```
X11.samples.lib.Core
```

Performance testing using a software technique of removing memory requires the fileset perfagent.tools (Performance Analysis and Control Commands), version 2.1.3.0, to enable artificial memory degradation.

Novell

Component

```
Model                   Intel 486/Pentium
RAM                     64 MB
Graphics                1280x1024 8 plane
```

Software

```
OS Version              UnixWare 2.02
Patches to OS           PTF2094 - TCP/IP Recovery
Network Communications  NIS, NFS

Compiler(s)             CCS Optimizing C Compilation System 3.0
                        C++ Compilation System 2.0
```

```
Additional components   See below
```

Comments

It is necessary to increase some kernel limits, as follows:

```
SDATLIM 0x8000000
HDATLIM 0x8000000
SSTKLIM 0x8000000
HSTKLIM 0x8000000
SVMMLIM 0x10000000
HVMMLIM 0x10000000
```

The following optional packages should be installed (this is pkginfo format):

```
application SDKdocs        Software Development Kit Documentation
graphics    desksup        Desktop Manager Development
application ihvkit         IHV Development Kit
system      nmsdk          Network Management SDK
system      softint        Software Packaging Tools
system      stdcomps       Standard Components
```

Silicon Graphics

This is a Motif–only platform.

Component

```
Model                   Indigo2
RAM                     64 MB
Graphics                1280x1024 8 plane
```

Software

```
OS Version              SC4-W4DI2-6.2: 6.2 with Indigo2 R10000 Impact
Patches to OS           None
Network Communications  SC4-NFS-6.2.1: NFS software

Compiler(s)             SC4-IDO-7.0: IRIX Development Option
                        SC4-C++-7.0: C++ compiler
```

Comments

In addition, there are three patches that apply to the compilers: patches 1451, 1453 and 1463.

Sun

Component

```
Model                   SPARCstation
RAM                     64 MB
Graphics                1152x900 8 plane
```

Software

```
OS Version              Solaris 2.4
Patches to OS           see below
```

```
Network Communications  NIS, NFS

Compiler(s)             SPARCompiler C 2.0.1
                        SPARCompiler C++ 3.0.1

Additional components   See below
```

Comments

The Solaris Common Build Environment is required.

The following patches are required. This list reflects patch numbers available as of 30 October, 1995. From time to time, replacement patches may be issued with changed suffix values; the current replacement patches may be substituted for those listed below.

```
100962-01       C++ patch for USL Cfront, release 3.0.2.
100974-02       dbx/ld scoped libs
101880-08       Mime message conversion field error
101907-05       volume management fixes
101925-02       Header file fixes
101933-01       /dev/mouse open causes cursor to freeze
101945-32       Kernel jumbo patch
102057-20       Xserver jumbo patch
102303-05       link editors require to provide for dynamic interfaces
102330-03       WABI fixes
102450-02       Docviewer will not start with full colormap
102650-02       New Tooltalk
103250-01       fd recalculate fix
```

Open/Closed Defect Lists

The following files listing defect reports are provided in the directories `$CDESOURCEDIR/cde` and `$CDESOURCEDIR/motif`:

`OPENBUGS`

Lists the currently known unresolved defects reported against CDE or Motif components respectively.

`CLOSEDBUGS`

Lists the defects that have been resolved since CDE 1.0.10 or Motif 2.0.

Additional Note

The **mapconv** utility function is available in the `cde-contrib` directory. It provides a method for converting a string of characters from one codeset type to another type.