
Corporate Data Archiver Software 5000.0.1.10

Release Notes and Administration Manual

Contents

Third Party Applications	2
International Trade Compliance	3
Introduction	4
Enhancements and Fixes	5
Pre-Installation Planning	7
Installation	8
Overall Workflow	11
Archive Generation from the GUI	12
Archive Generation from Command Line (ExtractXXArchive).....	21
Preparing for Tape Cutting: The Staging Process.....	23
Archiver GUI Additional Features	27
Integrating with WOW	28
Command-line Utilities	31
Integrating with EMC Centera	32
File-Level Archiving to Generic Disk	36
CDA File Transfer Utility - cdafile	37
CDA Administrator Web Interface	38
CDA User Web Interface	40
Appendix 1: Summary of Important Files and Variables.....	41
Appendix 2: The Archive Database (ADB)	43
Appendix 3: Moving Archiver Data into a new OpenWorks Project	47
Contacting Landmark Customer Support	48

Third Party Applications

Halliburton uses various third-party applications in the development of its software.

Halliburton acknowledges that certain third party code has been bundled with, or embedded in, its software. The licensors of this third party code, and the terms and conditions of their respective licenses, may be found at the following locations:

`$OWHOME/docs/third_party.pdf`

`$OWHOME/WebApps/tcltk/license.terms`

WOW is part of Halliburton's WebApps technology framework. WebApps is the shared architecture and directory name for the following products: WOW, Corporate Data Archiver (CDA), and Project Administrator (PA).

The following table lists the third party applications in WebApps 5000.0.

Product	Version	Company
Tcl/Tk	List of Tcl modules: <ul style="list-style-type: none">• Tcl 8.4.14• Tk 8.4.14• Tbcload 1.6.1• Oratcl 4.5• TkTable 2.9• TclXML 2.6• Cgi.tcl 1.10• BWidget 1.8• Tcllib 1.9• Tklib 0.4.1	Open Source (Code from SourceForge)
mod_auth_pam.so	1.1.1	Open Source
shapelib	1.2.10	Open Source (MapTools.org)
DGNLib	1.5	Open Source (MapTools.org)
Centera® SDK	3.1	EMC®
SDI cgm2png	1.0	System Development, Inc
EarthVision® EVGridIO api	1.2	Dynamic Graphics®

WebApps uses software developed by the Apache Software Foundation (<http://www.apache.org>). Apache is not distributed with WOW; the product uses the version of Apache 2.x delivered with the operating system.

Dynamic Graphics and EarthVision are registered trademarks of Dynamic Graphics, Inc. in the United States and other countries.

EMC is a registered trademark of EMC Corporation. Centera is a trademark of EMC Corporation.

Freeware source code INPOLY.C: Copyright (c) 1995-1996 Galacticomm, Inc.

International Trade Compliance

This application is manufactured or designed using U.S. origin technology and is therefore subject to the export control laws of the United States. Any use or further disposition of such items is subject to U.S. law. Exports from the United States and any re-export thereafter may require a formal export license authorization from the government. If there are doubts about the requirements of the applicable law, it is recommended that the buyer obtain qualified legal advice. These items cannot be used in the design, production, use, or storage of chemical, biological, or nuclear weapons, or missiles of any kind.

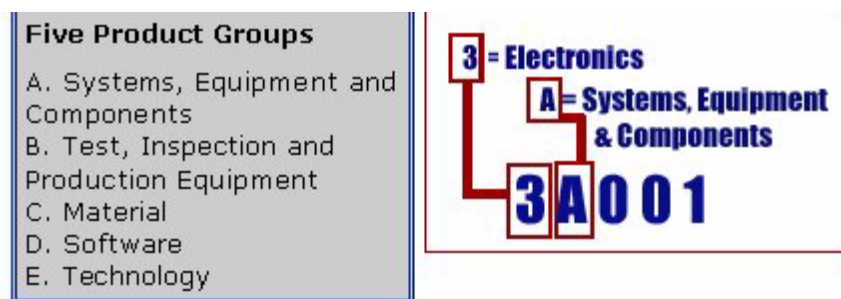
The ECCN's provided in Release Notes represent Halliburton's opinion of the correct classification for the product today (based on the original software and/or original hardware). Classifications are subject to change. If you have any questions or need assistance please contact us at: FHOUEXP@halliburton.com

Under the U.S. Export Administration Regulations (EAR), the U.S. Government assigns your organization or client, as exporter/importer of record, responsibility for determining the correct authorization for the item at the time of export/import. Restrictions may apply to shipments based on the products, the customer, or the country of destination, and an export license may be required by the Department of Commerce prior to shipment. The U.S. Bureau of Industry and Security provides a website to assist you with determining the need for a license and with information regarding where to obtain help. The URL is: <http://www.bis.doc.gov>

Definitions

CCATS (Commodity Classification Automated Tracking System) - the tracking number assigned by the U.S. Bureau of Industry and Security (BIS) to products formally reviewed and classified by the government. The CCATS provides information concerning export/re-export authorizations, available exceptions, and conditions.

ECCN (Export Control Classification Number) -The ECCN is an alpha-numeric code, e.g. 3A001, which describes a particular item or type of item, and shows the controls placed on that item. The CCL (Commerce Control List) is divided into ten broad categories, and each category is further subdivided into five product groups. The CCL is available on the EAR Website (http://www.access.gpo.gov/bis/ear/ear_data.html#ccl).



The WebApps Software, including WOW, Corporate Data Archiver (CDA) and Project Administrator (PA) are classified as follows: **ECCN number: 5D002, License type: ENC, CCATS number: G055789.** This information was last updated on 2/11/2008.

Introduction

Corporate Data Archiver (CDA) creates metadata-rich reports on subsurface technical project data, for subsequent archival or for general project documentation purposes. CDA covers OpenWorks, SeisWorks, general UNIX (including but not limited to Z-MAP Plus, SEG-Y, LAS and shapefile data), GeoProbe, ProMAX, and VIP. CDA is part of Halliburton's WebApps technology framework.

An archive is more than a project backup. Systems on the market today focus on the IT aspects of copying data from disk to tape. What makes CDA unique is the metadata-rich description of the project's contents that remains online post-archive. An innovative metadata extraction process produces an online 'stub' of the project contents. This 'stub' remains online, so that a project need never be restored 'just to see what's there'.

In addition to the static html 'stub', the Archiver also writes object-level metadata to an Oracle database, residing within an OpenWorks project schema. This means that the project can be documented post-archive, even after it has been removed from the system.

The Archiver also allows for a simple yet pragmatic approach for creating the physical archive, utilising standard UNIX utilities.

The list below summarizes some of the uses of the Corporate Data Archiver:

1. 'Google for E&P': use CDA's stub generator on all data, which allows a powerful cross-project cross-data type search for technical data
2. Project familiarization: save new interpreters months getting to know a new area by pre-generating project stubs. The stub is portable and can be copied to a laptop.
3. Project reporting: use CDA to auto-document technical project data for inclusion in an OpenJournal or other project report.
4. Knowledge capture: document a project using the stub, even after project has been taken offline.
5. Proactive disk management: disk is NOT cheap. Use CDA to flatten the rate of new disk acquisition, because disk management costs over its lifetime add up.
6. Project cleanup: use stubs to aid a project cleanup exercise pre-archive.
7. Data rooms: show acreage/data to potential partners in a simple, secure way.
8. Project snapshots for regulatory purposes: record what was submitted to the regulatory agency and when.
9. Non-operated partnership management: send snapshots to partners at critical points in the project lifecycle.
10. Version control: record the state of a project at regular intervals for audit purposes.
11. Disaster recovery: make daily snapshots of 'hot' projects for disaster recovery.
12. Archival: CDA helps you cut a tape to archive the project.

Enhancements and Fixes

See the WOW Release Notes for a list of changes to WOW.

Version R5000.0.1.10

There are no significant changes in this version.

Version R5000.0.1.9

- OpenWorks: fixed error creating neutral format archives with LAS export on wells without curves (defect 914415).

Version R5000.0.1.8

- SeisWorks: add seismic long remarks to the stub, and to stage area for neutral format archives (defect 901680).
- Geolog: restore ability to display previously-archived Geolog project stubs (Geolog is no longer supported) (defect 906115).

Version R5000.0.1.7

- Fixed issue with user-specified hyperlinks beginning with `https://` not being recognized (defect 898481).

Version R5000.0.1.6

- Changed horizon detail metadata page to display horizon data info, including external file name in storage directory (defect 892965).

Version R5000.0.1.5

- Fixed error in creating live trace outlines for very large bricked/compressed seismic volumes (defect 891791).

Version R5000.0.1.4

- SeisWorks: fixed issue with 3D horizon data files not being closed leading to 'too many open files' message - requires OpenWorks 8.1 (defect 881880).
- SeisWorks: trap case where neutral segy exports fail if the user description in `/etc/passwd` contains an apostrophe (defect 881700).
- SeisWorks: trap case where seismic file is absent during neutral format extract (defect 875343).

Version R5000.0.1.2 / 5000.0.1.3

There are no significant changes in these versions.

Version R5000.0.1.1

- Added district support, i.e. allow different Archive OpenWorks project and directory locations depending on the currently selected district using `cda_districts.dat` (defect 849544).
- Changed archives browsing page to display stub URLs directly, rather than derived from archived projects.
- Added support for GeoProbe V2 ascii/binary horizon file formats (defect 850103).
- Fixed bug in the CDA administration web page when deleting application-datatype associations.

Version R5000.0.1.0

- For OpenWorks project external ZGF directories: add `zgfThumbs` command-line utility and `ZgfThumbs` GUI utility to pre-generate ZGF picture images. If present, images will be displayed on a single page when browsing the directory in WOW (defect 849517)
- Fix bug archiving 3D surveys with spaces in the name.
- Fix style sheets for stub html; change to use a single style sheet for all applications.
- Fix core dump when archiving certain VIP VDB plot data.
- Documentation update to Appendix 3, moving archiver data into a new OpenWorks project (defect 854809).

Pre-Installation Planning

Important note: WOW is a prerequisite for CDA. Both applications use the same WebApps media. If you have installed WOW, you have installed CDA, and need only run the CDA install script as described in the [Installation](#) section. If you have not installed WOW, do so first before continuing with CDA configuration. See the WOW release notes for further details.

To avoid issues around permissions in the WebApps tree, run the CDA configuration script as the same user that installed WOW.

You must also designate an existing or create a new OpenWorks project to store Archiver metadata. The Archiver uses OpenWorks tables for leases, fields, basins, counties, stratigraphy etc. See [appendix 2](#) for more details on the Archiver data model.

Important note: To use multiple instances with CDA, create an ARCHIVES project with the same name on each instance.

Minimum Requirements

The minimum requirements for CDA are identical to WOW – see the WOW release notes for details.

CDA requires a fully configured WOW instance. Click on 'About' on the WOW top bar menu to see the current WOW version. If not currently on WOW version 5000.0.1.10, install and/or upgrade WOW to this version before proceeding.

Licensing

CDA uses standard Landmark FlexLM licensing with a feature name of CDARCHIVER. The Web interface to the product simulates a typical user session using the FlexLM 'linger' function. Whenever a user clicks on a link, the license is refreshed for a further 15 minutes. The license is checked back in after 15 minutes of inactivity, without the user needing to close the browser. Note that linger applies only to Web modules, not command-line utilities.

As WOW is a prerequisite for all WebApps modules, a WOW license feature will also be checked out. See the R5000 installation instructions and Release Notes for how to install licenses, and/or email license@lqc.com to obtain a valid license key.

Installation

Important note: WOW is a prerequisite for CDA. Both applications use the same WebApps media. If you have installed WOW, you have installed CDA, and need only run the CDA install script as described below. If you have not installed WOW, do so first before continuing with CDA configuration. See the WOW release notes for further details.

To configure CDA:

- Login to the application server as the chosen user (with write access to WebApps)
- Source the OpenWorks environment if not done by default:
prompt> source \$OWHOME/templates/.lgclogin (if C-shell)
prompt> . \$OWHOME/WebApps/templates/wow.profile (if Bourne shell)
- Run the WebApps installation script:
prompt> cd \$OWHOME/WebApps/install
prompt> ./WebAppsInstall -o \$OWHOME

```
WebApps Configuration R5000.0
=====
Install WOW (required 1st step)           1
Install Corporate Data Archiver          2
Install Project Administrator             3
EXIT .....                               0

Please enter option=> 4
```

- Choose option 2) to configure CDA as described further below.
- Alternatively, the CDA installation script can be run directly:
prompt> cd \$OWHOME/WebApps/install
prompt> ./CDAInstall -o \$OWHOME

After an environmental check, the installation script prompts for the following parameters (environmental variable name in parentheses):

- **Archive OW Project** (ARCHIVE_OW_PROJECT): the name of the designated OpenWorks project for storing Archive database records. The project must exist, and the install user must be a manager of the project with OW_ADMINISTRATOR privileges.
- **Archive Administrator Email** (ARCHIVE_ADMIN_EMAIL): email address for notification of archive project creation/modification. Defaults to the WOW email notification address.
- **Stub Directory** (ARCHIVE_STUB_DIR): the parent directory to which extracted metadata is written, e.g. /data/archive_stub.
- **Stage Directory** (ARCHIVE_STAGE_DIR): the directory from which the actual archive tape is created, e.g. /data/archive_stage.

-
- **Default Tape Device** (`ARCHIVE_TDEVICE`): the default tape device for the Archiver. Defaults to `/dev/rmt/0n`.
 - **Tape Device Capacity** (`ARCHIVE_CAPACITY`): the capacity of the default tape device in Mb. Defaults to `40000`.
 - **CDA Administration group** (`ARCHIVE_ADMIN_GROUP`): the UNIX group with permissions to generate archives in the WOW browser.
 - **EMC Centera IP address** (`CENTERA_IP_ADDRESS`): IP address of an EMC Centera system, if present (see section [Integrating with EMC Centera](#)).
 - **Review**: check the options specified and re-run if there are any errors.

During this stage the `$OWHOME/WebApps/conf/wow.env` file is updated with all the variables required by CDA. See [Appendix 1](#) for a list of all CDA configuration files and variables.

The installation script edits environmental variables, sets up required links in `$OWHOME/WebApps/bin`, and creates stub and staging directories.

Completing the Installation

To test the installation, attach `'bin/cda.cgi'` to the standard WOW URL, e.g. `http://wowservers.oilco.com/bin/cda.cgi`. Once the license is installed, CDA should be fully functional. An option for 'Archives' should also appear on the WOW left frame when next refreshed.

To test the CDA generator GUI, type the following in an OpenWorks xterm:

```
prompt> source $OWHOME/WebApps/templates/wow.login      (if C-shell)
prompt> . $OWHOME/WebApps/templates/wow.profile        (if Bourne shell)
prompt> archiver
```

The CDA archiver GUI is also available from the OpenWorks Command Menu, under Application->WOW/CDA/PA->CDA Archiver. To insert this launcher entry manually, you must launch via an xterm, e.g.:

```
"CDA Archiver" "xterm -e $OWHOME/WebApps/bin/archiver &"
```

Patching the Installation

WebApps patches (bug fixes only) and updates (bug fixes and minor enhancements) are designed as simple compatible additions or replacements of existing files in the software directories. They are provided as compressed tar files, available for download from the Landmark Software Manager.

Before installing any WebApps patch or update, please see the associated `README` for detailed installation instructions.

Current version can be obtained by clicking on 'About' on the WOW browser interface.

Optional District Configuration

From version 5000.0.1.1, CDA may be configured to independently store archive metadata based on district. To configure, edit `cda_district.dat`:

```
prompt> cd $OWHOME/WebApps/conf
prompt> cp cda_district.dat.default cda_district.dat
prompt> vi cda_district.dat
... and add district-dependent variables, e.g.:
```

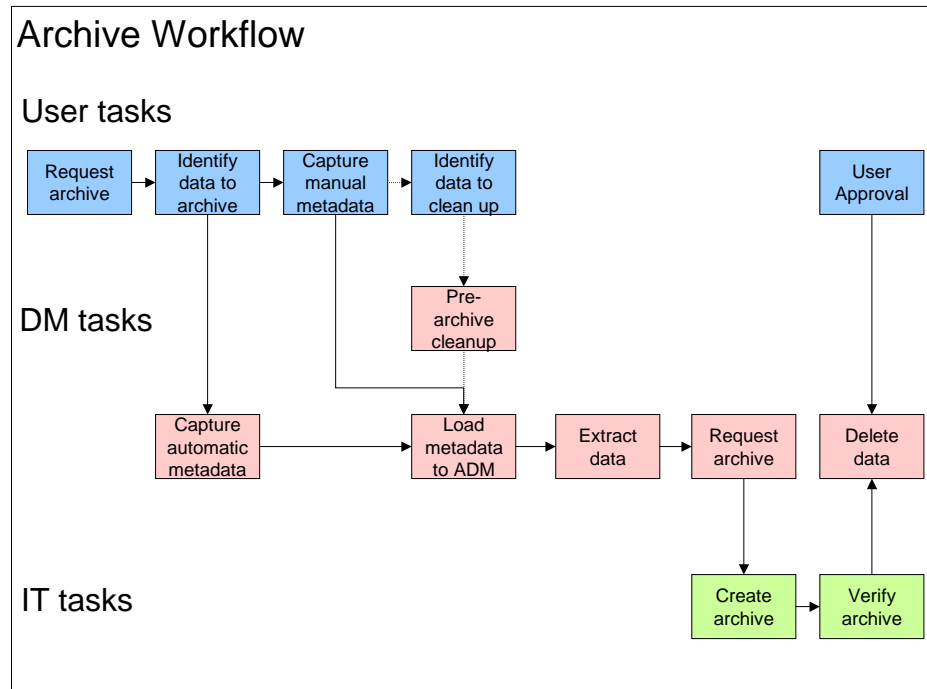
```
#####
#
# This file contains a list of district-dependent variables in CDA.
# File format:
# - DISTRICT (MUST match OpenWorks district.dat)
# - ARCHIVE_OW_PROJECT (OW project to store Archive database records)
# - ARCHIVE_STUB_DIR (Stub directory)
# - ARCHIVE_STAGE_DIR (Staging directory)
# - ARCHIVE_DISK_DIR (Generic disk archive directory)
# - ARCHIVE_STUB_URL (Stub URL - link must be created in WebApps/htdocs)
#
# If there is no entry for the current district defaults are taken
# from wow.env
#
# Comment out ARCHIVE_CTL_LOC in wow.env to use ARCHIVE_STUB_DIR as the
# control file location in each district
#
#####
ONSHORE    ARCHIVES1  /data/WebAppsData/archive_stub1
           /data/WebAppsData/archive_stage1 /data/WebAppsData/archive_loc1
           http://linux5/wow/archive_stub1
REGIONAL   ARCHIVES2  /data/WebAppsData/archive_stub2
           /data/WebAppsData/archive_stage2 /data/WebAppsData/archive_loc2
           http://linux5/wow/archive_stub2
DEEPWATER ARCHIVES3  /data/WebAppsData/archive_stub3
           /data/WebAppsData/archive_stage3 /data/WebAppsData/archive_loc3
           http://linux5/wow/archive_stub3
```

Note that links need to be created to allow the archive stub URLs to function; so for the example above, the following would be required:

```
prompt> cd $OWHOME/WebApps/htdocs
prompt> ln -s /data/WebAppsData/archive_stub1 archive_stub1
prompt> ln -s /data/WebAppsData/archive_stub2 archive_stub2
prompt> ln -s /data/WebAppsData/archive_stub3 archive_stub3
```

Overall Workflow

The Corporate Data Archiver is designed to separate end-user tasks (e.g. identifying data to archive/delete) from data administrator tasks (e.g. capturing automatic metadata) from IT tasks (cutting and verifying the tape). See illustration below.



The application is therefore modularized:

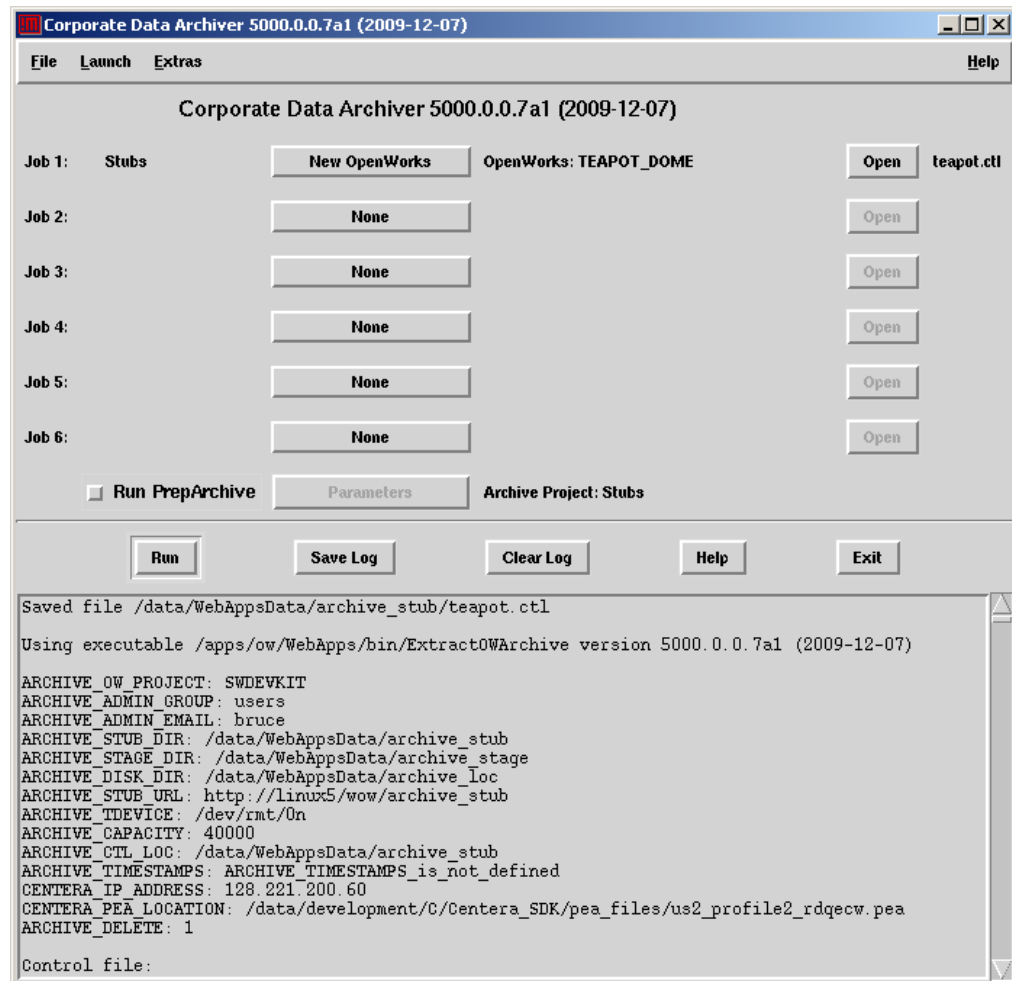
- **Extract:** the 'archiver' is a UNIX application with a graphical user interface for extracting Web-format metadata, writing object-level metadata data to Oracle and configuring the staging directory for the physical archive.
 - The Archiver can also be run in batch (command-line) mode.
 - The Archiver can also be run from within the WOW browser interface.
- **Pre-archival Cleanup:** WOW provides options for building lists of objects for subsequent deletion prior to archival
- **Prepare:** the Archiver does NOT create a tape: rather, it writes configuration information into a staging directory, which is then prepared via an additional step for writing to tape, near-line system, backup device, disk etc.
- **Administrator Interface:** a Web-based Archive Administration module is provided for maintaining the Oracle component of the archive metadata
- **User Interface:** a Web-based end-user module is provided for browsing, searching and documenting the archive.

The next sections describe these processes in more detail.

Archive Generation from the GUI

Archives can be created using the graphical user interface as described here, via a command-line method as described in the next section and in the WOW browser on OpenWorks, SeisWorks, GeoProbe, ProMAX, and VIP project summary pages.

A simple graphical user interface is provided which allows for creating/editing Archiver control files and execution of up to 6 extractions sequentially, with output written to screen and optionally saved to file. The GUI front page is illustrated below:



The archiver is invoked on the command line as follows:

```
prompt> source $OWHOME/WebApps/templates/wow.login      (if C-shell)
prompt> . $OWHOME/WebApps/templates/wow.profile        (if Bourne shell)
prompt> archiver
```

Users select a job type (e.g. OpenWorks, SeisWorks) or choose to open an existing control file.

Click on 'Open' next to any job brings up the appropriate control file editor, as illustrated overleaf for SeisWorks.

Click on 'Run' to execute the extraction using the control file as saved.

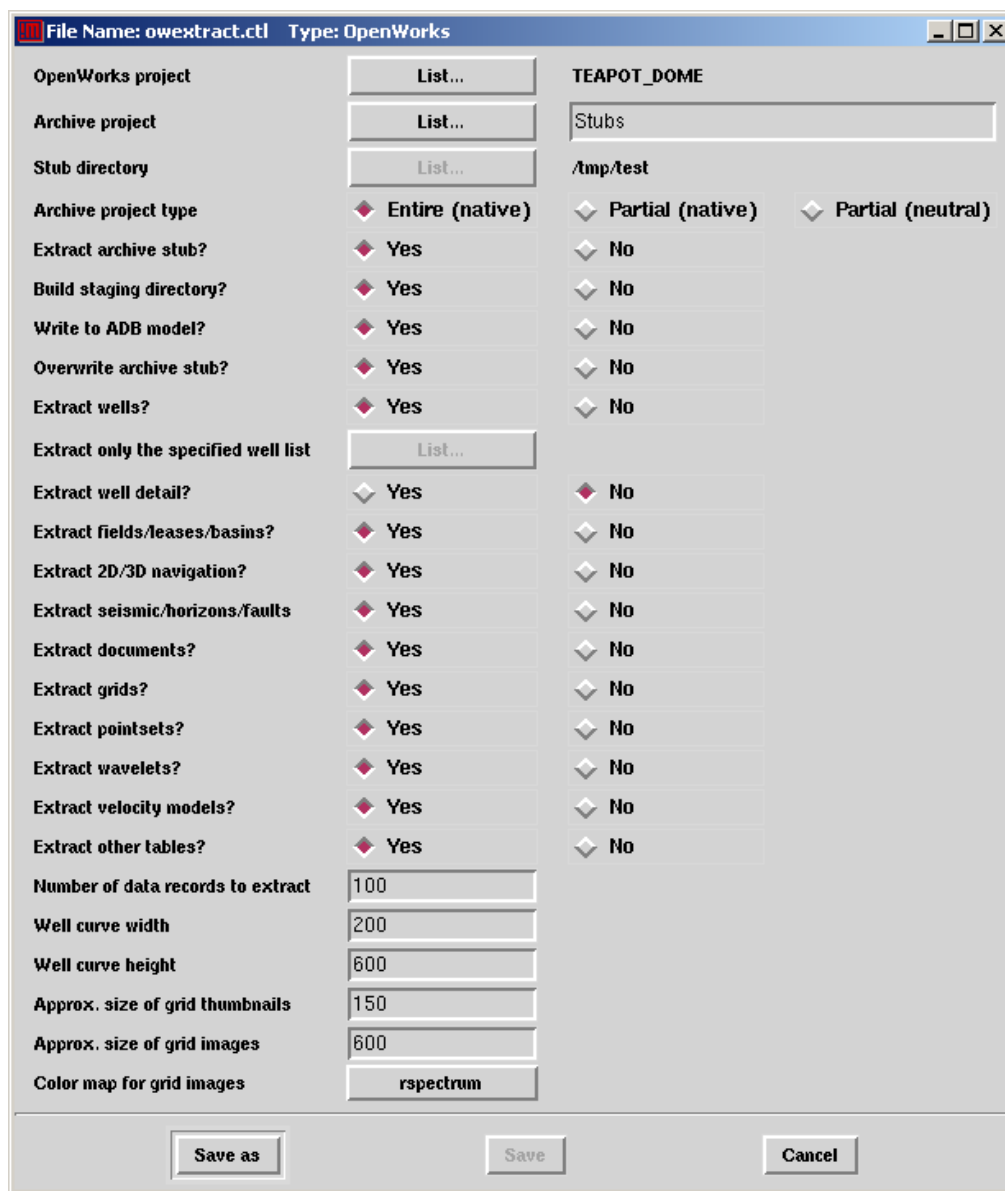
Click on 'Save Log' to save the log output from the bottom portion of the GUI to file.

Click on 'Clear Log' to clear the log window prior to subsequent runs.

Note: the 'open existing control file' and 'save as' options default to the location of ARCHIVE_STUB_DIR. To change this location, add the following line to \$OWHOME/WebApps/conf/wow.env:
ARCHIVE_CTL_LOC=/full/path/to/ctlfiles; export ARCHIVE_CTL_LOC

OpenWorks Control File Editor

The OpenWorks control file editor GUI is illustrated below:



Notes:

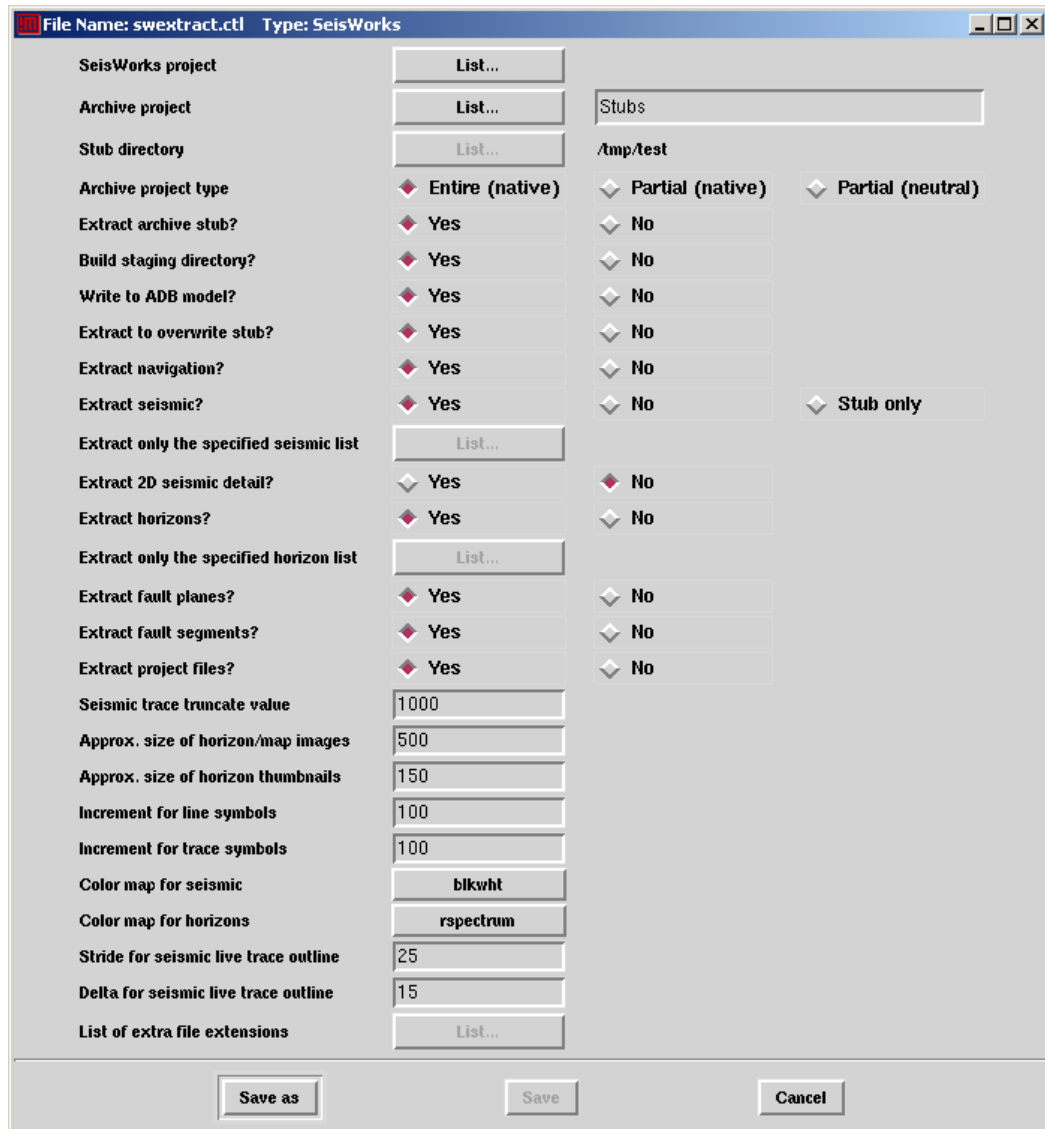
- The control file can be modified to allow either complete archival, or just stub generation to an arbitrary disk location. Changing **Write to ADB model** from 'yes' to 'no' will activate the **Stubs** dialog box.
- Change **Archive project type** to 'Partial (native)' to allow sub-selection by well list. This will enable the **Extract only specified well list** option, which restricts the extraction to the specified list only. However, with native format archival, the OpenWorks project backup will always contain all data.
- Set **Archive project type** to 'Partial (neutral)' to allow neutral-format archival of a subset of wells and data types. An OpenWorks neutral-format archive records the well headers, position logs, picks and checkshots in a self-describing ASCII format, and log curves in LAS format.
- The **Overwrite archive stub** parameter applies to the metadata extraction, and has no effect on the archive. Archive sets are always replaced in the CDA data model.
- Data type selection parameters, e.g. **Extract wells?** (yes/no) apply to both metadata extraction and archive set creation, although the project backup contains all the data.
- Metadata extraction for well data is slow. Changing **Extract well detail** to 'no' will considerably improve performance, by not drilling down to every object within every well.
- Width, height and size parameters are measured in pixels.

An example control file for OpenWorks is illustrated below.

```
# OpenWorks archiver preferences
project UTM16                # OpenWorks project (required)
arproj "Test Project"       # Archive project (required if model = yes)
artype partial_native       # Archive project type
directory /tmp/test         # Stub directory (required if model = no)
stub yes                    # Extract archive stub?
stage yes                   # Build staging directory?
model yes                   # Write to ADB model?
clob noclobber              # Overwrite archive stub? (clobber means yes)
wells yes                   # Extract wells?
welllist ""                 # Extract only the specified well list
welldetail yes              # Extract well detail?
fields yes                  # Extract fields?
leases yes                  # Extract leases?
grids yes                   # Extract grids?
pointsets yes               # Extract pointsets?
wavelets yes                # Extract wavelets?
velocity yes                # Extract velocity models?
other yes                   # Extract administration tables?
2dnav yes                   # Extract 2D navigation?
3dnav yes                   # Extract 3D navigation?
basins yes                  # Extract basins?
documents yes               # Extract documents?
maxpoints 100               # Number of data records to extract
xsize 200                   # Well curve width
ysize 600                   # Well curve height
tsize 150                   # Approx. size of grid thumbnails
gsize 600                   # Approx. size of grid images
colour rspectrum            # Color map for grid images
```

SeisWorks Control File Editor

The SeisWorks control file editor GUI is illustrated below:



Notes:

- The control file can be modified to allow either complete archival, or just stub generation to an arbitrary disk location. Changing **Write to ADB model** from 'yes' to 'no' will activate the **Stubs** dialog box.
- Change **Archive project type** to 'Partial' to allow sub-selection by seismic and horizon lists. This will enable the **Extract only specified seismic/horizon list** options, which restricts the extraction and the archive to the specified list only.
- Changing **Archive project type** to 'Partial' also enables the **List of extra file extensions** option, providing fine control over what project files are added to a partial archive. Critical project files are always archived, with extensions: .pdf

.pds .ps2 .pd2 .merge_cat .sm .line_cat .hrz_cat and .hrz. Also excluded from user control are all seismic and horizon files, with extensions: .hzd .hts .hzh_glb .hzd_glb .3dv .3dh .bri .cmp and .2v2_glb. This option allows the selective inclusion on non-critical files such as processing control files, colormaps etc. If no file extensions are specified, the default list is used as stored in \$OWHOME/WebApps/conf/sw_file_extensions.dat.

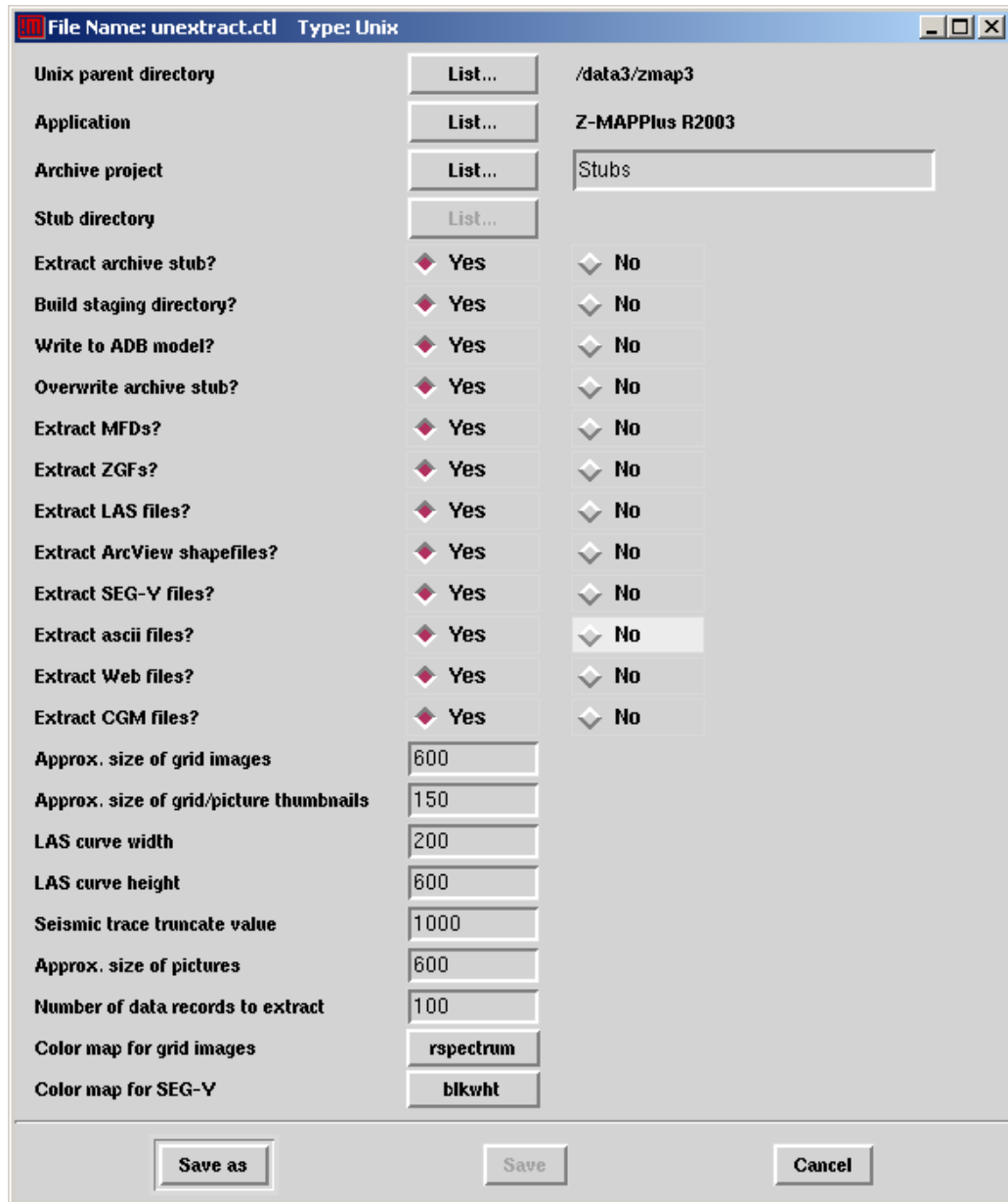
- Set **Archive project type** to 'Partial (neutral)' to allow neutral-format archival of a subset of data types. A SeisWorks neutral-format archive contains the project navigation, horizons and faults in a self-describing ASCII format, and seismic in SEG-Y format.
- The **Overwrite archive stub** parameter applies to the metadata extraction, and has no effect on the archive. Archive sets in the CDA data model and the archive staging area are always replaced.
- Data type selection parameters, e.g. **Extract horizons?** (yes/no) apply to both metadata extraction and archive set creation.
- The parameter **Extract seismic?** provides an extra option of 'stub only', which will create the stub, but not include the seismic files in the staging area. This can reduce archive size by 80-90% where seismic is mastered elsewhere, e.g. Petrobank.
- Metadata extraction for large 2D projects can be slow. Changing **Extract 2D seismic detail** to 'no' will improve performance by skipping the creation of live trace outlines.

An example control file for SeisWorks is illustrated below.

```
# SeisWorks archiver preferences
project devnor          # SeisWorks project (required)
arproj "Test Project"  # Archive project (required if model = yes)
artype partial_native  # Archive project type
directory /tmp/test    # Stub directory (required if model = no)
stub yes               # Extract archive stub?
stage yes              # Build staging directory?
model yes              # Write to ADB model?
clob noclobber        # Extract to overwrite stub? (clobber means yes)
nav yes                # Extract navigation?
seis yes               # Extract seismic?
seislist ""           # Extract only the specified seismic list
detail2d no           # Extract 2D seismic detail
hrz yes                # Extract horizons?
hrzlist ""            # Extract only the specified horizon list
flt yes                # Extract fault planes?
seg yes                # Extract fault segments?
file yes               # Extract project files?
ssize 1000             # Seismic trace truncate value (thumbs = every 4th trace)
xsize 500              # Approx. size of horizon/map images
tsize 150              # Approx. size of horizon thumbnails
lineinc 100           # Increment for line symbols
traceinc 100          # Increment for trace symbols
scolour blkwht        # Color map for seismic
hcolour rspectrum     # Color map for horizons
stride 25              # Stride for seismic live trace outline
delta 15               # Delta for seismic live trace outline
extlist ""            # List of extra file extensions
```

UNIX Control File Editor (includes Z-MAP Plus)

The UNIX control file editor GUI is illustrated below:



Notes:

- The control file can be modified to allow either complete archival or just stub generation to an arbitrary disk location. Changing **Write to ADB model** from 'yes' to 'no' will activate the **Stubs** dialog box.
- The **Overwrite archive stub** parameter applies to the metadata extraction, and has no effect on the archive. Archive sets in the CDA data model and the archive staging area are always replaced.

- Data type selection parameters, e.g. **Extract LAS files?** (yes/no) apply to both metadata extraction and archive set creation.
- The **Application** parameter is important and must match a value as set in the Archiver data model. When set to 'Z-MAPPlus' special extraction routines are run against MFD and ZGF files.
- The UNIX extraction will recursively walk through any subdirectories of the start directory specified in the control file.
- The **Extract ascii files?** parameter will create html stubs of the first **Number of data records** lines in the file. To identify a file as type ASCII, add in to `ascii_file_extensions.dat` in `$OWHOME/WebApps/conf`. If it does not exist,


```
prompt> cd $OWHOME/WebApps/conf
prompt> cp ascii_file_extensions.dat.default ascii_file_extensions.dat
```
- The **Extract web files?** parameter will copy 'web' files directly into the stub: these are 'mime types' that are universally understood by web browsers, e.g. html, gif, jpeg etc. To identify a file as a web file, add in to `web_file_extensions.dat` in `$OWHOME/WebApps/conf`. If it does not exist,


```
prompt> cd $OWHOME/WebApps/conf
prompt> cp web_file_extensions.dat.default web_file_extensions.dat
```

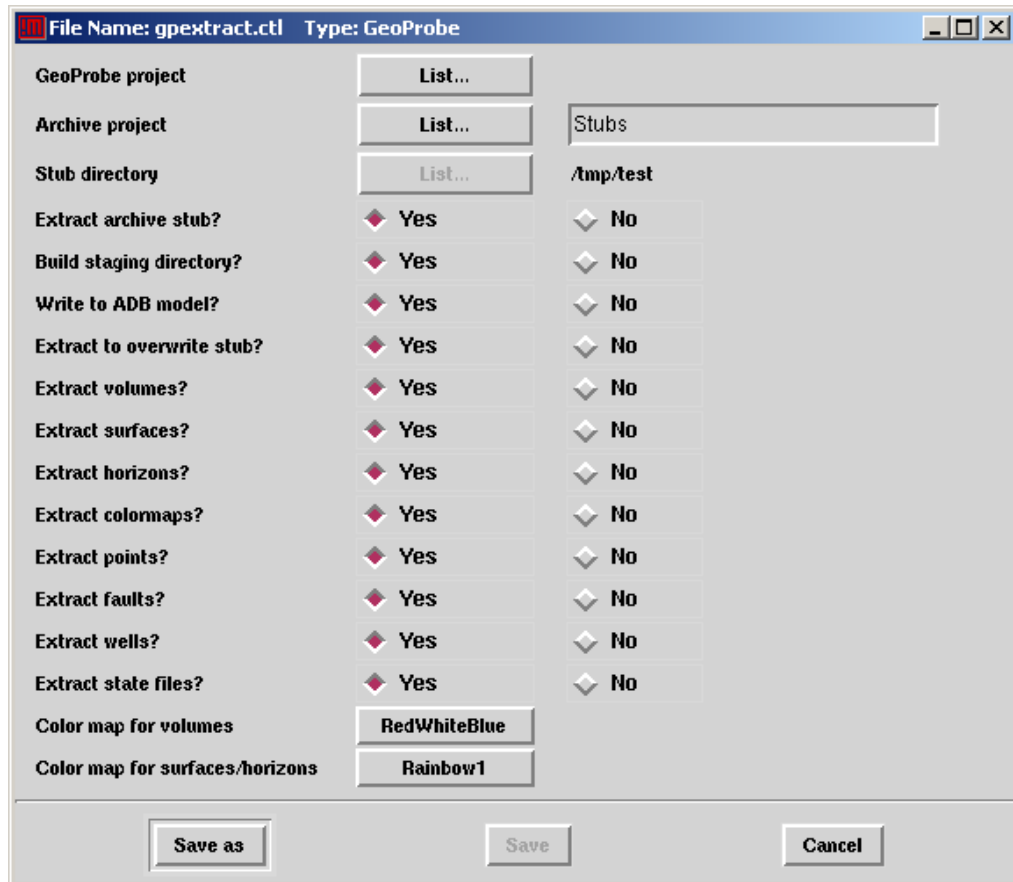
An example control file for general UNIX disks is illustrated below.

```
# UNIX archiver preferences (mostly for Z-MAPPlus)
startdirectory /data3/zmap3      # UNIX parent directory (required)
application "Z-MAPPlus"        # Application (must be registered in ADB)
arproj "Test Project"          # Archive project (required if model = yes)
arstartdirectory /tmp/test      # Stub directory (required if model = no)
stub yes                        # Extract archive stub?
stage yes                       # Build staging directory?
model yes                       # Write to ADB model?
clob noclobber                 # Overwrite archive stub? (clobber means yes)
mfd yes                         # Extract MFDs?
zgf yes                         # Extract ZGFs?
las yes                         # Extract LAS files?
shp yes                         # Extract ArcView shapefiles?
sgy yes                         # Extract SEG-Y files?
asc yes                         # Extract ascii files?
web yes                         # Extract web files?
tsize 150                       # Approx. size of grid/picture thumbnails
gsize 600                       # Approx. size of grid images
psize 600                       # Approx. size of pictures
colour rspectrum                # Color map for grid images
scolour blkwht                 # Color map for SEG-Y
ssize 1000                      # Seismic trace truncate value
xsize 200                       # LAS curve width
ysize 600                       # LAS curve height
numrecs 100                     # Number of data records to extract
```

Other Control File Editors

Similar editors exist for GeoProbe, VIP VDB data, and ProMAX. These are not visible on the `archiver` GUI unless configured.

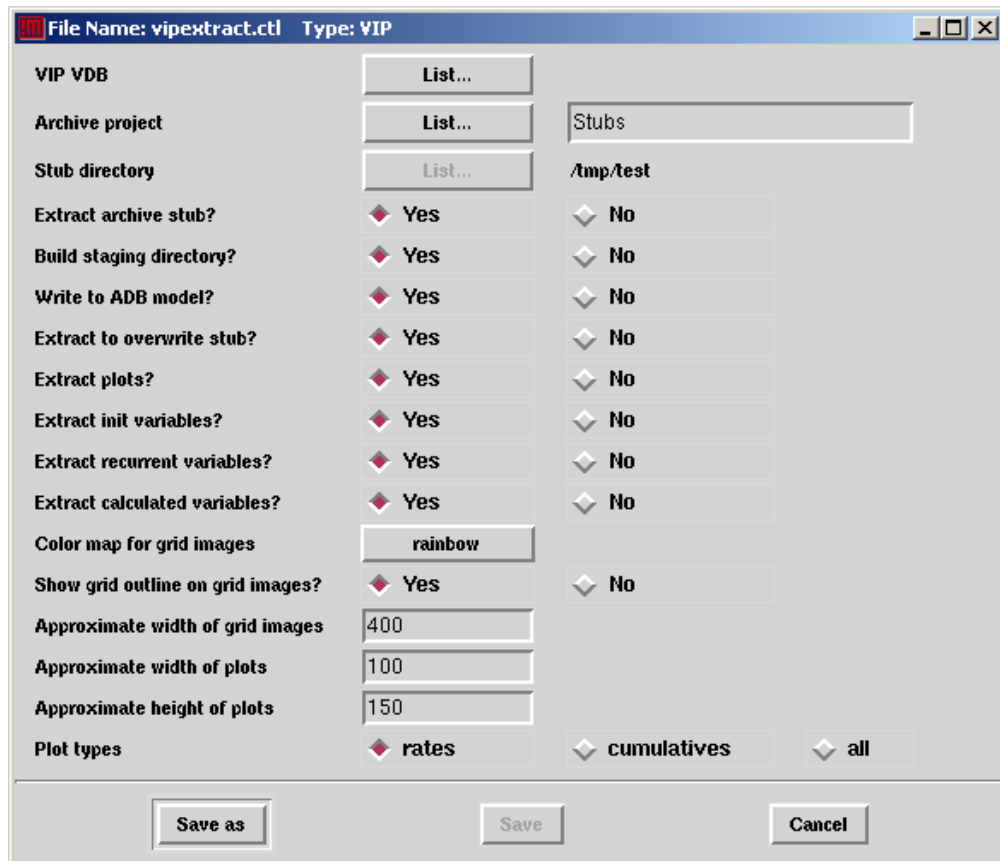
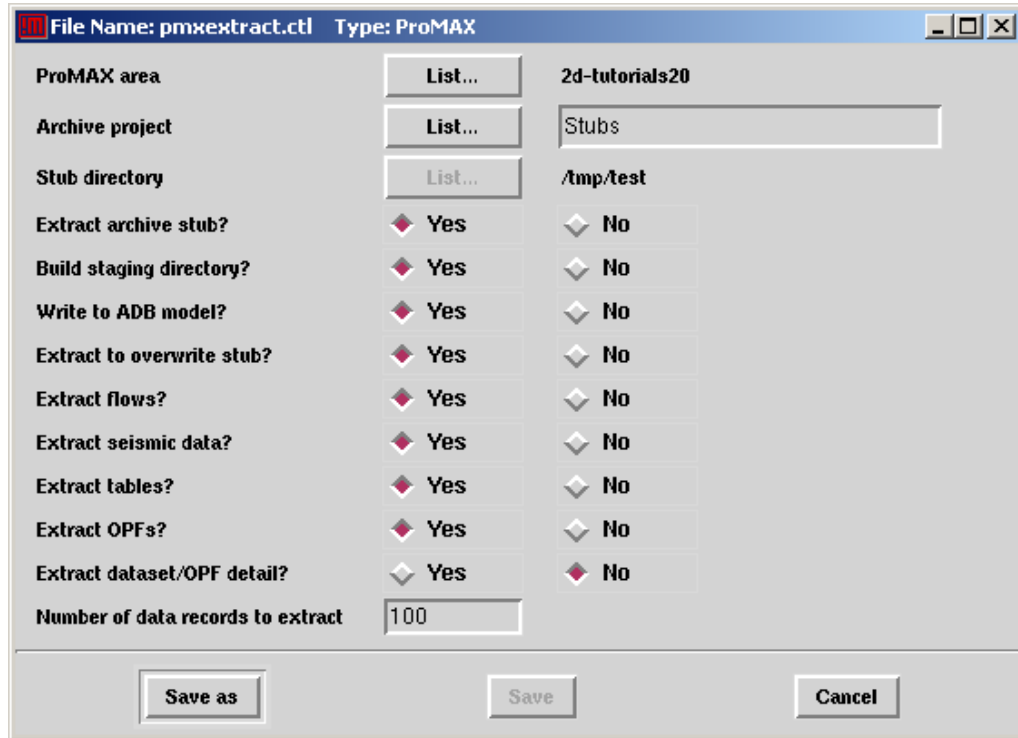
An example GeoProbe GUI editor is illustrated below:



Notes:

- The control file can be modified to allow either complete archival, or just stub generation to an arbitrary disk location. Changing **Write to ADB model** from 'yes' to 'no' will activate the **Stubs** dialog box.
- The **Overwrite archive stub** parameter applies to the metadata extraction, and has no effect on the archive. Archive sets in the CDA data model and the archive staging area are always replaced.
- Data type selection parameters, e.g. **Extract volumes?** (yes/no) apply to both metadata extraction and archive set creation.

Example ProMAX and VIP GUI editors are illustrated below:



Archive Generation from Command Line (ExtractXXArchive)

Archives can be created using the graphical user interface, or from the WOW project summary page (for most applications), or via a command-line method. This section describes the command-line method; the previous section describes the GUI interface.

Important Note: Archives cannot be created from the Archiver Web forms. Use the Archiver Web forms only for adding extra documentation and attributes to the archive.

The archive creation routines perform three functions:

1. Writes sets and objects to the archive data model (ADB). Multiple sets are created per application: up to four for a SeisWorks project, up to 10 for OpenWorks and one set per Z-MAP Plus directory/subdirectory. Sets correspond to data types, e.g. for SeisWorks there are navigation, seismic, horizon and fault sets. Objects correspond to the individual seismic volumes, horizons etc.
2. Calls the metadata extraction scripts to write metadata to
`$ARCHIVE_STUB_DIR/<archive_proj>/<app>/<app_proj>`
3. Calls the staging file creation scripts to create the staging directory with file lists and instructions for subsequent archiving in
`$ARCHIVE_STAGE_DIR/<archive_proj>/<app>/<app_proj>`.

There are separated batch archive creation scripts for OpenWorks, SeisWorks, UNIX, GeoProbe, ProMAX, and VIP projects. All the scripts take a control file as argument:

```
ExtractSWArchive <full path to control file>
ExtractOWArchive <full path to control file>
ExtractUNArchive <full path to control file>
ExtractGPArchive <full path to control file>
ExtractVIPArchive <full path to control file>
ExtractGLArchive <full path to control file>
```

For example, to create an archive project with an OpenWorks project, a SeisWorks project, and two UNIX directories containing Z-MAP Plus and SEG-Y data:

```
ExtractOWArchive /home/fred/ow_tampen.ct1
ExtractSWArchive /home/fred/sw_st97m3.ct1
ExtractUNArchive /home/fred/un_pl103.ct1
ExtractUNArchive /home/fred/un_pl197.ct1
```

The exact nature of the action carried out by these scripts depends on the parameters set in the relevant control file, described in the next section.

Remember to source the OpenWorks environment before running these scripts, which are created as links in `$OWHOME/bin` to `$OWHOME/WebApps/bin/lmksh_wrapper`.

Notes on Archiver Control Files

Whether invoked through the GUI, from WOW or from the command line, control files are utilized which contain the required parameters, such as project names, data selections, image size etc. Default versions are provided in `$OWHOME/WebApps/dat/archive`:

```
OpenWorks: owextract.ct1
SeisWorks: swextract.ct1
```

UNIX: unextract.ctl (includes Z-MAP Plus)
GeoProbe: gpextract.ctl
VIP: vipextract.ctl

Important note: we recommend not modifying the default control files. Rather, make a copy of the files, e.g.

```
prompt> cp $OWHOME/WebApps/dat/archive/swextract.ctl ~/dan3d.ctl  
prompt> vi ~/dan3d.ctl
```

Control files can be edited by hand and submitted on the command line, as described in the following section. When editing control files by hand, note that the parameter name (1st value) cannot be modified. The value (2nd parameter) is under user control. Any text after the '#' is treated as a comment and is used for labeling within the archiver GUI. Values with spaces should be quoted, e.g. "Q23 well list".

Independent Stub Creation

It is entirely possible to run the metadata extraction without reference to archival. For example, there may be a requirement to produce an html appendix to an OpenJournal project which details the data used in an analysis. Or to generate a temporary stub to assist with a data cleanup operation.

To do this, set the control file parameters as shown:

```
stub      yes   # Extract archive stub?  
stage     no    # Build staging directory?  
model     no    # Write to ADB model?
```

Independent File Staging

It is also possible to prepare for physical archive without reference to archival. For example, there may be a requirement to snapshot a project for transfer to another office or to a partner.

To do this, set the control file parameters as shown:

```
stub      no    # Extract archive stub?  
stage     yes   # Build staging directory?  
model     no    # Write to ADB model?
```

Preparing for Tape Cutting: The Staging Process

For native format archival, lists of files are prepared, rather than copying entire projects to the staging area. The objective is a simple, robust and extensible methodology that creates an archive tape that is sufficiently granular and well described, so that restoring the tape is trivial.

The process separates the 'prepare' and 'execute' stages of physical tape cutting. This allows the Corporate Data Archiver to create the archive by tar to tape, tar to file or gtar (GNU tar, which spans multiple tapes). The approach employed allows both Landmark and clients to add alternative methods over time.

Support is also provided for archival to the EMC Centera purpose-built disk-based archival system, designed to overcome the limitations of tape. See section [Integrating with EMC Centera](#).

File-level archiving (copying) to generic disk is also provided: see section [File-Level Archiving to Generic Disk](#).

The Archiver GUI or command-line execution writes all required configuration to the staging directory during the extraction process, as described below.

SeisWorks Configuration

For SeisWorks, the following files completely describe the project contents for the subsequent prepare step.

- An empty `<project>.swp` file with the name of the SeisWorks project. If neutral-format, the convention is `<project>.swn`.
- A `.dir` file per directory in the SeisWorks project, with the system directory alone annotated differently. Each of these files contains a project directory name, the name of the relative file listing for that directory (including any subdirectories), and a computed directory size in Mb.
- A `.lst` file matching each `.dir` file, containing a relative file listing for that directory (including subdirectories). This file is designed to be used with the tar command with the 'include from file' option.
- For neutral-format archives, there is one `.dir` file for each neutral-format subdirectory in the staging area. The `.lst` file contains a relative file listing for that directory as before.
- For 2D native-format archives, the process in addition also creates the `.dir` and `.lst` files for only those files in the associated master project that are used by the working project.
- A `stub.loc` file containing the location of the archive project metadata stub directory.
- Files to describe the associated OpenWorks project (see below). Archiving a SeisWorks project without the associated OpenWorks project is not

recommended. OpenWorks stores checkshots, picks and curves used in well-to-seismic ties, as well as faults and navigation.

OpenWorks Configuration

For OpenWorks, the following files completely describe the project contents for the subsequent prepare step.

- An empty `<project>.owp` file with the name of the OpenWorks project. If neutral-format, the convention is `<project>.own`. If an interpretation project, a file is also created for the associated project database.
- A file `OW_SYS_DATA.loc` containing a single line with the location of the OpenWorks system data directory.
- A `.dir` file per OpenWorks project external directory. Each of these files contains a directory name, the name of the relative file listing for that directory (including any subdirectories), and a computed directory size in Mb.
- A `.lst` file matching each `.dir` file, containing a relative file listing for that directory (including subdirectories). This file can be used with the tar command with the include (-I) flag.
- For neutral-format archives, there is one `.dir` file for each neutral-format subdirectory in the staging area. The `.lst` file contains a relative file listing for that directory as before.
- A `stub.loc` file containing the location of the archive project metadata stub directory.

UNIX/GeoProbe/VIP Configuration

- An empty `<unixdir>.und` file with the name of the UNIX directory ('/' and other special characters are replaced by '_').
- A `.dir` file for the UNIX directory. The file contains a directory name, the name of the relative file listing for that directory (including any subdirectories), and a computed directory size in Mb.
- A `.lst` file matching the `.dir` file, containing a relative file listing for that directory (including subdirectories). This file can be used with the tar command with the include (-I) flag.
- A `stub.loc` file containing the location of the archive project metadata stub directory.

The PrepArchive Script

This script acts on the specified archive project staging directory and creates a README and ARCHIVEME script. However clients may bypass this step with their own script if required.

The `PrepArchive` script has the following usage:

```
PrepArchive <stage directory> <type> <device> <capacity>
```

Tape types currently supported are: **tar** (the default), **tarfile** and **gtar** (spans multiple tapes). The recommended tape format and the default is **gtar**, since this provides a consistent open standard across multiple UNIX platforms.

Device is a no-rewind tape device or a file. Capacity is tape limit in Mb, i.e. 8000 for an 8mm Exabyte, 40000 for a DLT 7000. Tape device and capacity will default to the variables `$ARCHIVE_TDEVICE` and `$ARCHIVE_CAPACITY` is set.

You may set `ARCHIVE_CAPACITY` to a high number when using `gtar` or `tarfile`, since in both cases (multi-volume `gtar` or file) the tape length error message is not relevant.

For example, to emulate the `TARME` from the previous version of CDA (with `gtar`):

```
cd <archive stage directory>
PrepArchive (no further arguments required)
```

To archive from `/archive_stage/ProjX` to a tar file in the same directory:

```
PrepArchive /archive_stage/ProjX tarfile /archive_stage/ProjX.tar 100000
```

To archive in tar format to a new tape device `/dev/rmt/1n` with capacity 40Gb:

```
PrepArchive /archive_stage/ProjX tar /dev/rmt/1n 40000
```

To archive in `gtar` format to a tape device `/dev/rmt/2n`:

```
PrepArchive /archive_stage/ProjX gtar /dev/rmt/2n 1000000
```

The `PrepArchive` script (and any custom alternative) should perform the following:

- Creates a tar file of the archive stub directory using file `stub.loc`
- Builds lists of OpenWorks, SeisWorks and UNIX projects using files `.swp`, `.owp` and `.und` files)
- Creates a tar file of the `OW_SYS_DATA` directory if required using file `OW_SYS_DATA.loc`
- Creates OpenWorks backups of all OpenWorks projects using `owbackup`. Should you wish to skip this step, touch an empty file with name `<owproj>.dmp`. `PrepArchive` will then skip OpenWorks backup creation.
- Writes a README providing detailed information as to what exactly is on the tape
- Writes an ARCHIVEME script to create the actual physical archive.

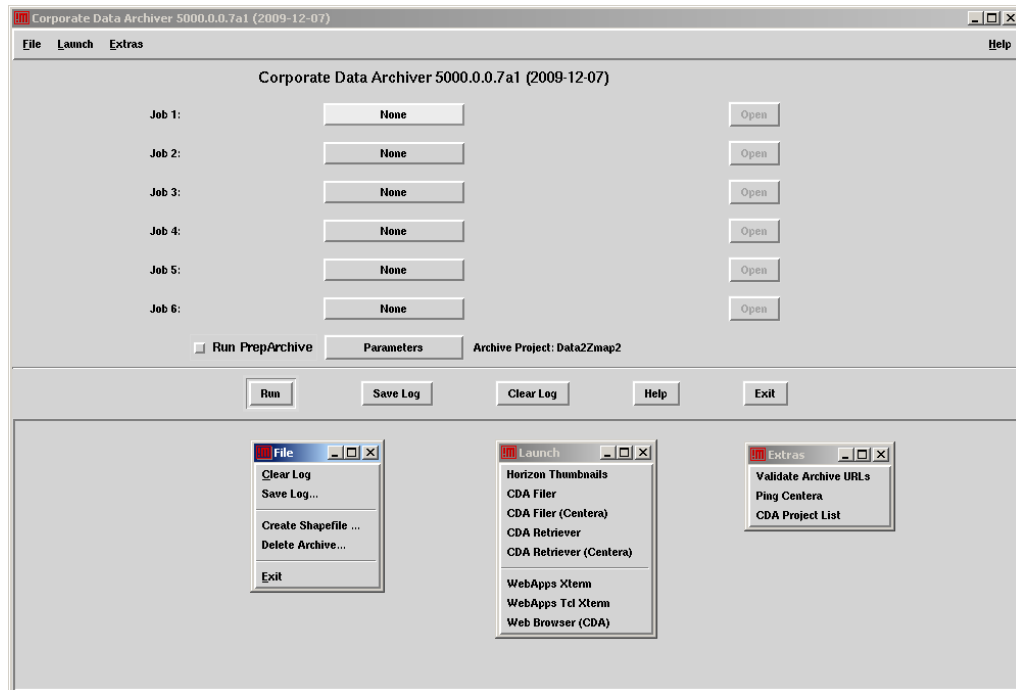
Manual inspection of the ARCHIVEME file is strongly recommended prior to execution. This should contain:

- Definition of `ARCHIVE_TDEVICE`
- Definition of `ARCHIVE_CAPACITY`
- A check for total archive size not exceeding the value of `$ARCHIVE_CAPACITY`
- Command to write the `README`, stub and other 'table of contents' info to tape
- Command to write `OW_SYS_DATA` tar file to tape
- For each OpenWorks project, write project backups to tape
- For each OpenWorks project external directory, write directories to tape
- For each SeisWorks project directory, write directories to tape
- For each UNIX directory, write directory to tape.

Note that the first file on tape contains the `README` itself and the file lists. This means that whoever restores the tape then knows exactly how many tar files to expect, and the size of each.

Archiver GUI Additional Features

The archiver GUI provides a launcher for most of the CDA command-line and GUI utilities, accessible from the File, Launch and Extras menus as illustrated below:



The archive GUI provides access to the following:

- The Create Shapefile functionality from the CDA Administrator Web interface
- The Delete Archive functionality from the CDA Administrator Web interface
- The Horizon Thumbnail GUI (`HzThumbs`)
- The ZGF Thumbnail GUI (`ZgfThumbs`)
- The CDA Filer GUI (`cdafiler`)
- The CDA Retriever GUI (`retriever`)
- An xterm containing the WebApps environment
- A WebApps Tcl xterm
- A browser running the CDA User Web interface
- The Validate Archive URLs command-line utility (`ValidateArchiveURLs`)
- The Centera ping command-line utility (`emcPing`)
- The CDA project list command-line utility (`cdalist`)

Integrating with WOW

CDA can be used to generate 'throwaway' stubs which include image thumbnails for re-use within WOW. These options are provided for OpenWorks, SeisWorks, GeoProbe, ProMAX and VIP projects.

This functionality is designed to facilitate a two-phase archival process: in the initial step, a 'throwaway' stub is created and used to clean up the project, i.e. deleting empty horizons and intermediate processing seismic volumes. Then in the 2nd step the Archiver is re-run in order to generate the final archive.

To generate these in CDA, use an archive project name of 'Stubs' to create thumbnails in a pre-determined location, for re-use in the WOW horizon list builder. Once generated, the stub can be viewed directly from a link on the SeisWorks survey summary page.

The SeisWorks horizon list builder is accessible from the 'Horizons' link on the SeisWorks project summary page. This allows viewing horizons, drilling down to view details or selecting horizons to save to a list.

SeisWorks Horizons

An alternative way to generate just SeisWorks horizon thumbnails is to use the `hrzThumbs` (command-line) or `HrzThumbs` (graphical) utilities, as illustrated below.

To execute on the command-line, first source the WebApps environment then type `hrzThumbs`:

```
prompt> source $OWHOME/WebApps/templates/wow.login      (if C-shell)
prompt> . $OWHOME/WebApps/templates/wow.profile        (if Bourne shell)
prompt> hrzThumbs
```

```
hrzThumbs owproj survey ?tsize? ?clm? ?force?
```

<owproj> OpenWorks project name

<survey> SeisWorks survey name (or All2D for 2D projects)

<tsize> approximate thumbnail width in pixels, default 100

<clm> optional SeisWorks color map (clm) file name, default is `rspectrum`

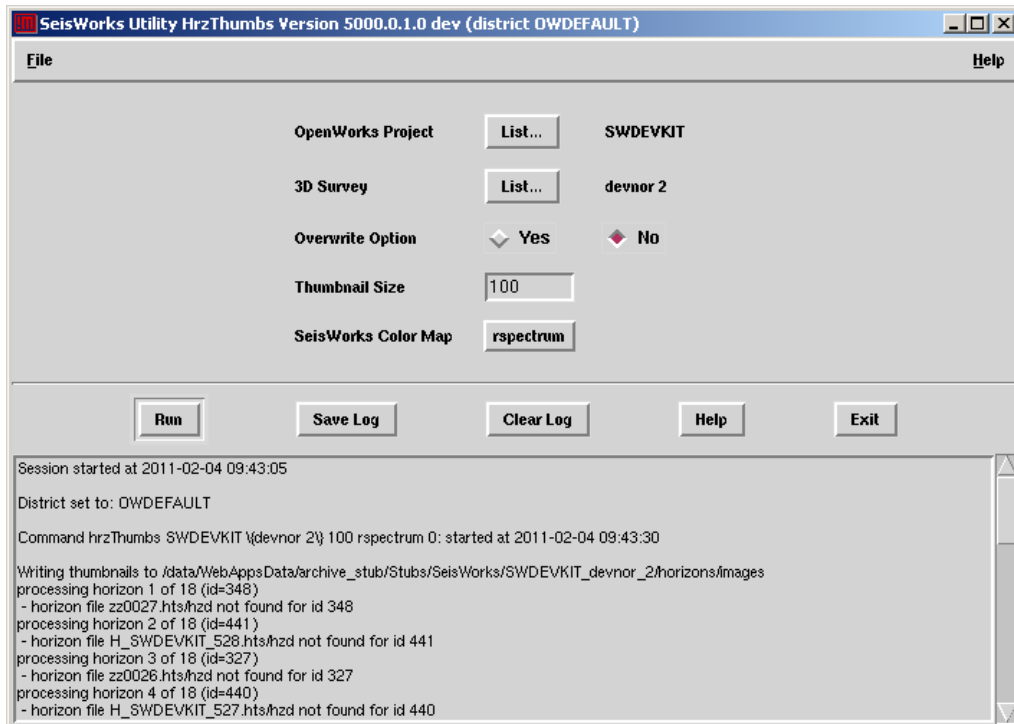
<force> optional flag to force overwriting of existing thumbnails, e.g. if changing `clm`.

The example below will create the thumbnail images for horizons in survey `devnor` using the `spectrum` color map and with a width of 150 pixels.

```
prompt> hrzThumbs SWDEVKIT devnor 150 spectrum
```

To execute from a graphical user interface, run `HrzThumbs`. The GUI is illustrated below:

```
prompt> HrzThumbs
```



OpenWorks ZGFs

An alternative way to generate just ZGF picture thumbnails for OpenWorks external project directories, is to use the `zgfThumbs` (command-line) or `ZgfThumbs` (graphical) utilities, as illustrated below.

To execute on the command-line, first source the WebApps environment then type `zgfThumbs`:

```
prompt> source $OWHOME/WebApps/templates/wow.login      (if C-shell)
prompt> . $OWHOME/WebApps/templates/wow.profile        (if Bourne shell)
prompt> zgfThumbs
```

```
zgfThumbs dir ?tsize? ?psize?"
```

<dir> Full path to OpenWorks project external ZGF directory

<tsize> approximate thumbnail width in pixels, default 150

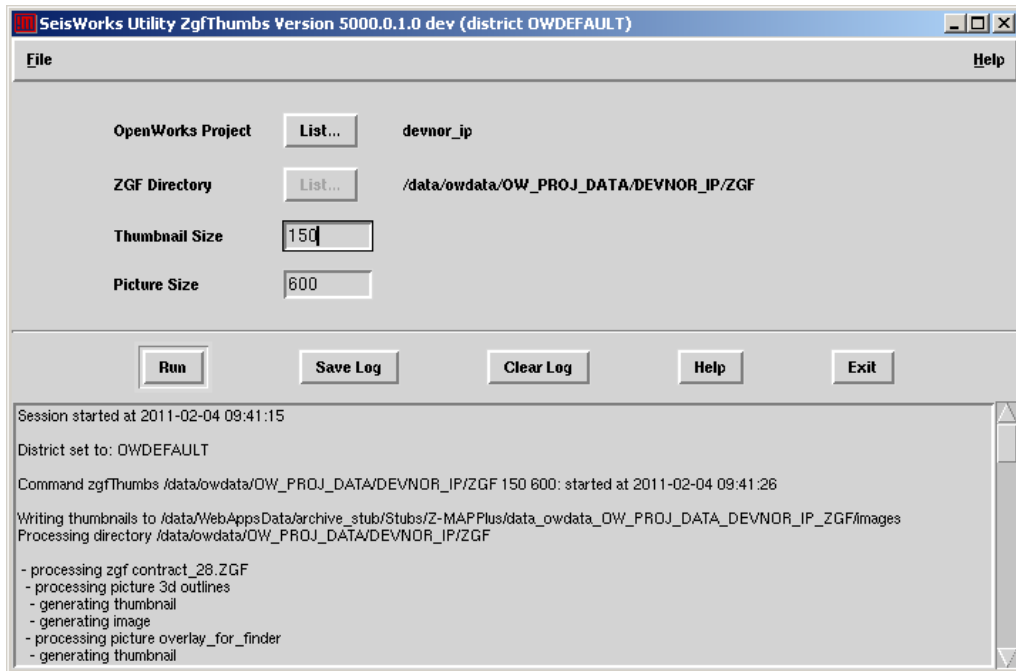
<psize> picture width in pixels, default 600

The example below will create the thumbnail images for project DEVNOR_IP with a width of 200 pixels for thumbnails and 800 pixels for the full-size images.

```
prompt> zgfThumbs /data/owdata/OW_PROJ_DATA/DEVNOR_IP/ZGF 200 800
```

To execute from a graphical user interface, run `ZgfThumbs`. The GUI is illustrated below:

```
prompt> ZgfThumbs
```



Command-line Utilities

Before running any utilities, source the WebApps environment if not already done so:

```
prompt> source $OWHOME/WebApps/templates/wow.login      (if C-shell)
prompt> . $OWHOME/WebApps/templates/wow.profile        (if Bourne shell)
```

To produce a list of archive projects:

```
prompt> cdalist
```

ID	Archive Project	Application
22	CURVEMODE	OpenWorks R2003
61	Data2Zmap2	Z-MAPPlus R2003
67	FLOUNDER	OpenWorks R5000
68	KMOC	OpenWorks R5000
43	SEGY	Z-MAPPlus R2003
80	SWDEVKIT	OpenWorks R5000
78	Salt2	OpenWorks R5000
81	TeapotDome	OpenWorks R5000
54	devnor	SeisWorks R2003
46	pwt	SeisWorks R2003

It is also possible to execute this process through the archiver GUI. menu Extras->CDA Project List.

To check the archive stub URLs as stored in the CDA database:

```
prompt> ValidateArchiveURLs
```

```
ARCHIVE_STUB_URL: http://linux5/wow/archive_stub
```

```
URLs in use for archive OW project SWDEVKIT:
```

- http://exprotech/wow/archive_stub
- http://linux3/wow/archive_stub
- http://linux5/wow/archive_stub
- http://sun10/wow/archive_stub

In most implementations, there should be a single URL matching \$ARCHIVE_STUB_URL

Possible corrections:

```
ValidateArchiveURLs http://exprotech/wow/archive_stub
http://linux5/wow/archive_stub
ValidateArchiveURLs http://sun10/wow/archive_stub
http://linux5/wow/archive_stub
```

To correct, execute the suggested corrective usage, e.g.:

```
prompt> ValidateArchiveURLs http://exprotech/wow/archive_stub
http://linux5/wow/archive_stub
36 records updated in adb_project
488 records updated in adb_set
```

Integrating with EMC Centera

Introduction

CDA can optionally be configured to write to EMC Centera, a purpose-built disk-based archival system designed to overcome the limitations of tape. For more information on the EMC Centera solution, see <http://www.emc.com/products/systems/centera.jsp>.

EMC Centera complements CDA's rich metadata extraction and reporting capability, by adding robust, file-level archival to disk. This avoids CDA's simple PrepArchive process of creating scripts that archive to tape or tar file, and also provides CDA with file-level restore capability for the first time.

CDA interacts with EMC Centera via an api. The application passes a file plus additional metadata (archive project, application, application project, data type, file location and file size) to the EMC Centera, which computes a unique content address for the object, saves the file plus metadata, and returns the address to CDA for storage. The content address then serves as a 'claim check' for the file: the new CDA retriever application can be used to present the 'claim check' to the EMC Centera for instantaneous retrieval.

EMC Centera's *content addressed storage* (CAS) has the following advantages:

- Eliminates the need to manage the physical location of files on hierarchical file systems.
- Provides a flat address space for huge numbers of archive objects stored by content address, together with customizable metadata.
- Provides fast, direct access (sub-second) to individual objects – without needing to know object location.
- Single instance storage: archive a project repeatedly over time, but there will only be a single copy of a unique file actually stored beyond a configurable threshold, resulting in considerable space savings
- The content address is more than a unique identifier; it is a digital fingerprint of the content, ensuring its authenticity.
- CAS solutions maintain at least two reliable copies of every file with content protection mirroring, guaranteeing data integrity.
- Optional, real-time mirroring of the data ensures certified data protection and provides business continuity.
- Easy installation and non-disruptive upgrades.
- Self-configuring, self-managing, self-healing.

When using CDA with EMC Centera, the online, always available nature of the archive can result in increased trust in the archival process by the business. Users can view and retrieve archive objects directly: this removes the 'retrieval gap' which typically runs from days to weeks, and encourages users to allow content to be archived. In this way the archive can itself become an attractor for more content.

Configuration

Note that EMC Centera is not required, i.e. CDA will continue to run exactly as before by default, making no assumptions about the presence of an EMC Centera system.

Configuration of CDA to recognize an available EMC Centera system requires 3 steps:

Step 1: Install and configure the EMC Centera hardware. This operation is usually carried out by an EMC engineer and is not discussed further in this document. After installation, an IP address or addresses will be provided which identify the location of the EMC Centera hardware on the network.

Step 2: Edit the `wow.env` configuration file.

Set the environmental variable `CENTERA_IP_ADDRESS` in `wow.env`:

```
prompt> cd $OWHOME/WebApps/conf
```

```
prompt> vi wow.env
```

add the following lines:

```
# EMC Centera address
CENTERA_IP_ADDRESS=128.221.200.56; export CENTERA_IP_ADDRESS

# EMC Centera GUI delete (0=off 1=on)
#CENTERA_DELETE=1; export CENTERA_DELETE
```

Environmental variable `CENTERA_DELETE` is provided to allow or disable the capability to delete EMC Centera archive objects using the CDA retriever GUI.

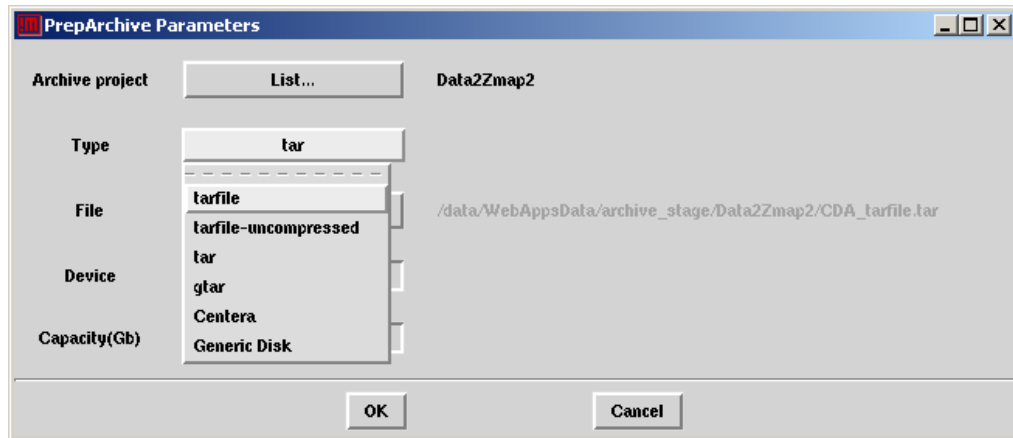
Environmental variable `CENTERA_PEA_LOCATION` can optionally be set to control EMC Centera profile/pool. This depends on how the EMC Centera system is configured, and is not discussed further here.

To check Centera configuration, use the `emcPing` utility as illustrated below:

```
prompt> emcPing
Centera IP address from environment: 128.221.200.60
Centera .pea file from environment:
/data/development/C/Centera_SDK/pea_files/us2_profile2_rdqecw.pea
Centera pool opened, pool address = 496695932009256
Centera pool closed.
```

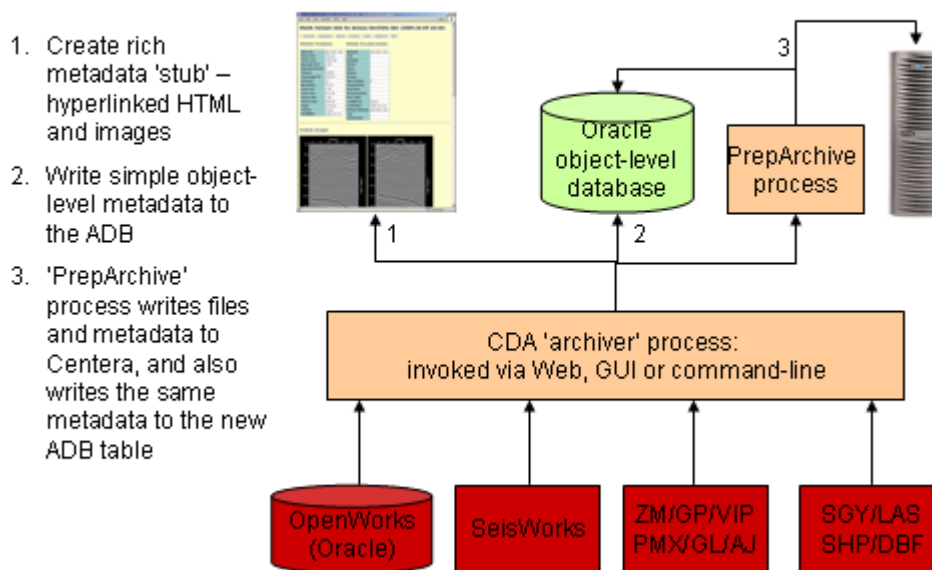
Execution

The CDA archiver GUI will look for the environmental variable `CENTERA_IP_ADDRESS`, and if defined, show 'Centera' option in the PrepArchive process, as illustrated overleaf.



When archiving to EMC Centera, always check the 'Run PrepArchive' checkbox and select archive type of 'Centera'. This will ensure that the extraction of metadata, storage of files and metadata on the EMC Centera, and storage of the same metadata plus content address in CDA, all take place in the same operation. The process is illustrated below:

CDA – Centera Integration



The new CDA retriever GUI allows browsing, filtering, validation, retrieval and deletion of archive objects stored on the EMC Centera. The GUI is illustrated below. The retriever is invoked on the command line as follows:

```
prompt> source $OWHOME/WebApps/templates/wow.login      (if C-shell)
prompt> . $OWHOME/WebApps/templates/wow.profile         (if Bourne shell)
prompt> retriever -c
```

The GUI is illustrated overleaf:

Change filter options in the top pane, then hit the 'Query' button to narrow the view.

To select files: click, hold and drag; or use <shift> and <control> to do multiple selections.

Once selected, files can be validated (check the content address exists on the EMC Centera), retrieved (to their original or a newly specified location), or deleted from the EMC Centera (provided the CENTERA_DELETE flag is set).

Click on any column heading to sort by that column; a 2nd click will reverse the sort.

Column widths can be adjusted by dragging any vertical column divider.

The screenshot shows the 'CDA File Retriever 5000.0.0.7a1 (2009-12-07)' window. At the top, there are search filters: 'Files matching query criteria: 64', 'Archive Project: All', 'Application: All', 'Application Project: All', 'Filename: ', 'Larger than (bytes): 1000', and 'Data Type: All'. A 'Query' button is on the right. Below the filters is a table with the following columns: Archive Project, Original Location, Bytes, Application, Application Project, Datatype, and Alert Addr. The table contains 20 rows of file data. At the bottom of the window, there are buttons for 'Validate', 'Delete', 'Retrieve', and 'Exit'. A status bar at the very bottom shows: 'Session started at 2009-12-08 16:45:19 Running in CENTERA mode... Searching ...done: 64 files matched criteria: project=All, application=All, application project=All, datatype=All, file like *, size > 1000'.

Archive Project	Original Location	Bytes	Application	Application Project	Datatype	Alert Addr
43	Centera /data1/zmap/Centera_test/testbar.cgm	8704	Z-MAPPlus	/data1/zmap/Centera_1	Pictures	C00MMRF
44	Centera /data1/zmap/Centera_test/tom.tar	44032	Z-MAPPlus	/data1/zmap/Centera_1	None	EPC7I6RV
45	Centera /data1/zmap/Centera_test/worldmap.cgm	13520	Z-MAPPlus	/data1/zmap/Centera_1	Pictures	70P0CRC
46	sametail /data/WebAppsData/archive_stage/sametail/sametail_stub.tar.gz	32024	CDA	sametail	Stub	3U49M83
47	sametail /data/WebAppsData/archive_stage/ADB_export.dmp	1182720	CDA	n/a	ADB Backup	DOJVJSV
48	sametail /data2/zmap/sametail/DRAKE.ZGF	390656	Z-MAPPlus	/data2/zmap/sametail	Mapping	2UFQG61.
49	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/Amplitude-Pk.gpc	11681	test		test	ALTNL6
50	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/Amplitude-Tr.gpc	11681	test		test	7ORA7TA
51	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/Combination-PK.gpc	11765	test		test	8C7D3NA
52	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/Combination-Tr.gpc	11765	test		test	8U637RFL
53	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/InstAmplitude.gpc	11765	test		test	8Q81SI11
54	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/InstFrequency.gpc	5374	test		test	50EJ41BE
55	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/InstPhase.gpc	11892	test		test	6T5NR1CI
56	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/Rainbow1.gpc	11639	test		test	EJ32E0P.
57	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/Rainbow2.gpc	11639	test		test	7L46B2E0
58	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/Semblance.gpc	11597	test		test	5KGGFAC
59	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/duo.gpc	15631	test		test	A60HCFK
60	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/grayscale.gpc	6083	test		test	64TIS8M1
61	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/semb	4922	test		test	0L7R82UF
62	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/semb2	4188	test		test	8841LDUT
63	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/sweet.gpc	11681	test		test	F0UEIVQ
64	CDA_Filer_Internal /data1/GeoProbe_data/Banshee/Colormaps/vdefault.gpc	11681	test		test	91QN0LFI

EMC Centera – Database Validation

To bulk validate that the CDA (ADB) database contains all records stored on the EMC Centera, and vice-versa, use `emcValidateClusterAdb <IP address> <days>`

E.g. to validate an EMC Centera system created within the last year against CDA:
`prompt> emcValidateClusterAdb 128.221.200.56 30`

The utility will report objects in the ADB database not on EMC Centera, and vice-versa.

File-Level Archiving to Generic Disk

Introduction

CDA can optionally be configured to write to generic disk, as an alternative to EMC Centera. Note that in this situation, the management and integrity of the archive location on disk is the responsibility of the customer. Advantages of EMC Centera such as single instance storage and self management do not apply, and the archive itself must be managed for space usage, and backed up.

The generic disk option does however allow CDA to support file-level archiving to disk, with reciprocal file-level restore.

Configuration

Configuration of CDA to write to generic disk requires 3 steps:

Step 1: Identify an appropriate archive disk location, and create the directory:

```
prompt> mkdir <directory>
```

where <directory> is the full path to the generic disk archive location, e.g. /data/archive_loc.

Step 2: Edit the `wow.env` configuration file to point to the generic disk archive location:

```
prompt> cd $OWHOME/WebApps/conf
```

```
prompt> vi wow.env
```

scroll down to the section containing `ARCHIVE_` variables and add the following:

```
ARCHIVE_DISK_DIR=<directory>; export ARCHIVE_DISK_DIR
```

where <directory> is the full path to the generic disk archive location created in step 1.

Execution

The CDA `archiver` GUI will look for the environmental variable `ARCHIVE_DISK_DIR`, and if defined, show 'Generic Disk' as an option in the `PrepArchive` process.

When archiving to generic disk, always check the 'Run `PrepArchive`' checkbox and select archive type of 'Generic Disk'. This will ensure that the extraction of metadata, storage of files on disk, and storage of the same metadata plus address in CDA, all take place in the same operation.

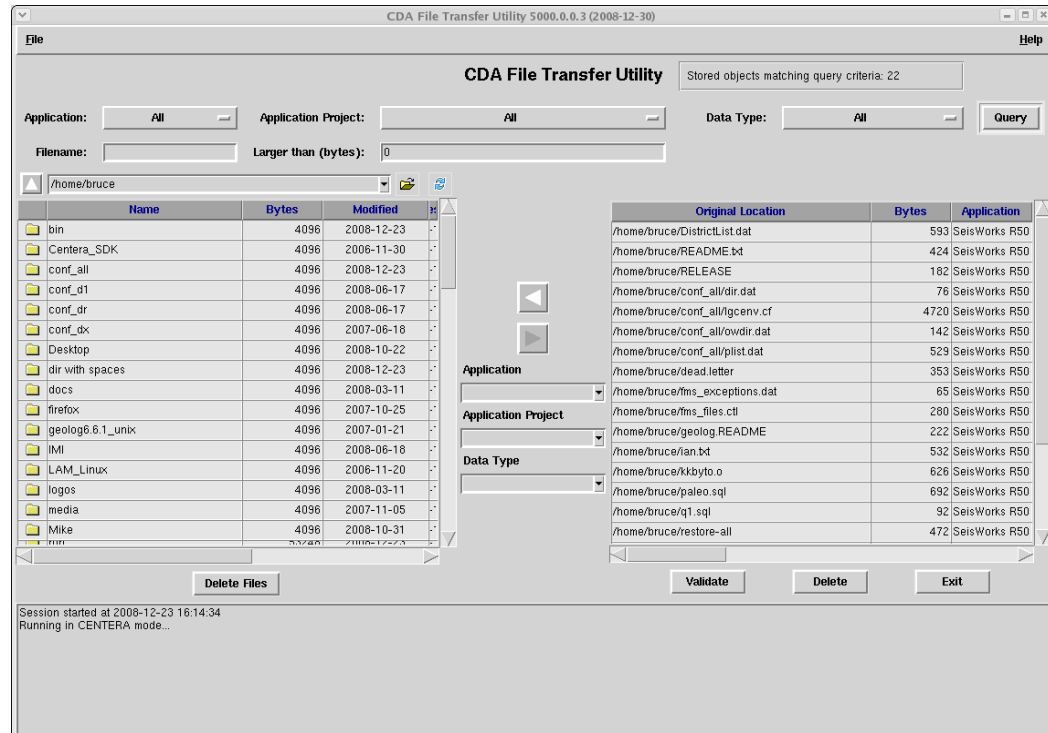
The new CDA `retriever` GUI allows browsing, filtering, validation, retrieval and deletion of archive objects stored on generic disk, in an identical manner as described for EMC Centera. The `retriever` is invoked on the command line as follows:

```
prompt> source $OWHOME/WebApps/templates/wow.login      (if C-shell)
prompt> . $OWHOME/WebApps/templates/wow.profile        (if Bourne shell)
prompt> retriever
```

CDA File Transfer Utility - cdafile

CDA is typically used to archive entire projects or directories of files. This can be somewhat restrictive in the Centera or Generic Disk situation, where the user may wish to manually select only specific files, from arbitrary disk locations, to archive.

The `cdafile` utility addresses this need, by providing a GUI to allow arbitrary files to be copied from disk to Centera or Generic Disk archive, and vice-versa. The utility is illustrated below: online disk files are tabulated on the left, and archive contents on the right.



To archive, navigate to the required directory on the left, select files, choose the archive application, project and data type tags (or type in new values) in the centre of the GUI, and hit the right-pointing arrow to copy the selected files to archive.

To retrieve, use filters at top if required to restrict the list, select files on the right, then hit the left-pointing arrow to copy from archive to disk.

Files can also be deleted, both from online disk and from the archive.

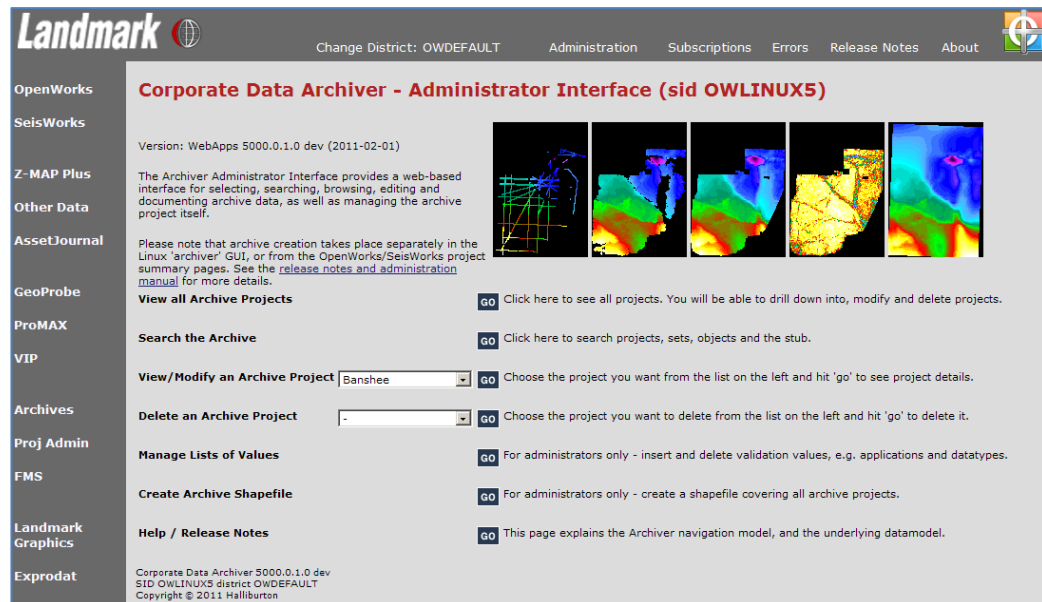
Both disk and archive tables can be sorted, by clicking on any column header. Click again to reverse the sort.

Note: the `cdafile` utility uses a special archive project named `CDA_Filer_Internal` to store metadata associated with the files archived. This should not be deleted, or used for other archival purposes.

CDA Administrator Web Interface

Creating and Deleting

A Web-based Archive Administration module is provided for maintaining the Oracle component of the archive metadata. This allows the manual creation and deletion of archive projects, sets and objects. This helps 'fix' archives where the user has made an error during creation.



Important note: changing archive project class from 'pending' to 'complete' disables addition and deletion, in order to protect the archive metadata. The class must be set to 'pending' in order to do further addition or deletion.

Manage Lists of Values

The 'manage list of values' option allows an administrator to add and delete entries in the lookup tables used in the ADB data model. The default values are:

Archive format: native, neutral

Application: EDMS, Eclipse, GeoProbe, Hampson-Russell, IrapRMS, MSOffice 2000, OpenWorks, Petrel, SeisWorks, StratiMagic, VIP, Z-MAPPlus

Datatype: 2D Nav, 2D Seismic, 3D Model, 3D Nav, 3D Seismic, AVO, Basemap, Basins, Documents, Fault Polygons, Faults, Fields, Grids, Horizons, Inversion, Leases, MS Access, MS Excel, MS PowerPoint, MS Word, Pointsets, Project Snapshot, Simulation Model, UNIX Files, Velocity, Wavelets, Wells

Media type: 3480, 3490, 3590, DLT 4000, DLT 7000, Exabyte 8200, Exabyte 8500

Project class: complete, pending

Project status: offline, online active, online inactive

Project type: bid round, data room, farm-in, regional exploration, well proposal

Notes on 'Application Datatype Associations': this maps which data types are associated with which application. It is important when adding additional general applications, to ensure that you make an association to the data type 'UNIX Files'.

Shapefile Creation

CDA provides a simple shapefile generation utility to provide rectangular project outlines from the 'project boundary' table, which is populated by clicking on the 'create from header' link on the project summary page, provided minimum and maximum latitude/longitude coordinates are stored in the project header.

The shapefile is in geographic coordinates. You can either specify a fully-pathed UNIX directory and file stem, e.g. /data/arctview/gom/archives. This will produce the required .shp .shx and .dbf files. Alternatively, if only the file stem is specified, the shapefile will be written to the \$WOW_DATA_DIR/wow_shapes/CDA directory.

CDA User Web Interface

A Web-based end-user module is provided for browsing, searching and documenting the archive. The primary difference between the 'admin' and 'user' interfaces is that archive projects, sets and objects cannot be created or deleted in the 'user' interface. They can however be modified, e.g. in order to add comments to document the archive.

The user interface is also simplified, by merging multiple data sets into an application project summary. Data sets can still be displayed by clicking on a link (if the user needs to add comments for example). Project user/boundary are also skipped in the user interface.

The user interface is illustrated below. The user either browses the archive database by drilling down on hyperlinks, or utilizes the search to find data across multiple projects. The main objective is to allow the user to find the archived project and check out the stub, in order to decide whether to request a restore.

Archive Projects (sid OWLINUX5)

[Show project headers](#)

[Show project counts](#)

[Search projects](#)

[Administer CDA](#)

[Banshee](#)
[CDA_Filer_Internal](#)
[CGMfiles](#)
[Cantera](#)
[Cantera_with_spaces](#)
[Data2Zmap2](#)
[FLOUNDER](#)
[Geplog5000](#)
[KMCic](#)
[OCEVET](#)
[OWplusSW](#)
[PmxGrowsper](#)
[ProMax_3Dtut](#)
[ProMax_3Dtut](#)
[ProMax_3Dtut2](#)
[Simple](#)
[Stubs](#)
[TESTDATA](#)
[TESTDATA2](#)
[T2_differant_CRS](#)
[Teapot](#)
[UTM16_test](#)
[VIP_amath](#)
[VIP_offshore](#)
[VIP_vfa2008](#)
[VIP_westjar](#)
[data1_zmap](#)
[data2_zmap](#)
[data3_zmap](#)
[devnor](#)
[devnor2](#)
[minkt_ip_on](#)
[partial_neutral](#)
[sametail](#)
[sw2d](#)
[sw3d](#)
[sw3d2](#)

Archive Project OCEVET

[\[project headers | project counts | search all projects | modify project header | help \]](#)

Project Header (Request Restore)

ID	91	Country	NORWAY
Project Name	OCEVET	State	UNKNOWN
Description	Created by Bruce Rodney on 2011-01-21 13:02 through archiver		
Start Date	12-JAN-10	County	UNKNOWN
End Date	18-DEC-10	Basin	Greater Falklands
Type	well proposal	Field	Potanay
Class	pending	Block	UNKNOWN
Status	offline	Min Lat	0
Source	Oilco	Max Lat	0
Proprietary?	Y	Min Lon	-7.488567
Document ID	http://linux5/wow/archive_stub/OCEVET/pri_replaceme.html	Max Lon	-7.488567
		Area Description	UTM30N BG EuropeanDatum1950 SP Intl1924
		CRS	Intl1924

Archive Header

Archive Description	Requested By	Request Date	Archived By	Archive Date	Stub Directory	Staging Directory	Media Type	Media ID
Created during project create of OCEVET					\$ARCHIVE_STUB_DIR/OCEVET	\$ARCHIVE_STAGE_DIR/OCEVET		

[Browse the archive staging area](#)

Application Project Summary

Stub	Application	Application Project
↩	OpenWorks R5000	OCEVET

[Display additional archive project details](#)

Corporate Data Archiver: 5000.0.1.0 dev
SID OWLINUX5: default ONDEFAULT
Copyright © 2011 Halliburton

Appendix 1: Summary of Important Files and Variables

The table below summarizes the main files used by CDA:

File/Directory	Description
<code>\$OWHOME/WebApps/dat/archive:</code> <code>owextract.ctl</code> <code>swextract.ctl</code> <code>unextract.ctl</code> <code>gpextracl.ctl</code> <code>vipextractl.ctl</code> <code>prj_replaceme.html</code> <code>set_replaceme.html</code> <code>*.css</code>	Directory containing CDA configuration files: Default OpenWorks control file Default SeisWorks control file Default UNIX control file Default GeoProbe control file Default VIP control file Default html file copied to project header Default html file copied to set header Default style sheets used in the 'stub'
<code>\$OWHOME/WebApps/conf:</code> <code>wow.env</code> <code>ascii_file_extensions.dat</code> <code>sw_file_extensions.dat</code> <code>web_file_extensions.dat</code>	Directory containing WebApps configuration files Main WOW/CDA configuration file (see below) List of files types interpreted as ASCII List of extra SeisWorks extensions for partial archives List of file types interpreted as Web formats, e.g. jpg
<code>\$OWHOME/WebApps/archive:</code>	Code files for the Archiver
<code>\$OWHOME/WebApps/archive/ sqlscripts:</code>	Directory containing Archiver data model SQL scripts

The list below provides an example of the environmental variables which can be set in `wow.env`. See the WOW release notes for an example of a complete `wow.env` file.

```
#=====
# Corporate Data Archiver environmentals
# Change value of all variables here
#-----

# Name of OW project to store Archive database records
ARCHIVE_OW_PROJECT=ARCHIVES; export ARCHIVE_OW_PROJECT

# Archive admin group (can generate archives in browser)
ARCHIVE_ADMIN_GROUP=${WOW_ADMIN_GROUP}; export ARCHIVE_ADMIN_GROUP

# Email address of the Archive administrator
ARCHIVE_ADMIN_EMAIL="${WOW_ADMIN_EMAIL}"; export ARCHIVE_ADMIN_EMAIL

# Stub directory
ARCHIVE_STUB_DIR=${WOW_DATA_DIR}/archive_stub; export ARCHIVE_STUB_DIR

# Staging directory
ARCHIVE_STAGE_DIR=${WOW_DATA_DIR}/archive_stage; export ARCHIVE_STAGE_DIR

# Generic disk archive directory
```

```
ARCHIVE_DISK_DIR=${WOW_DATA_DIR}/archive_loc; export ARCHIVE_DISK_DIR

# Stub URL
ARCHIVE_STUB_URL=${WOW_URL}/wow/archive_stub; export ARCHIVE_STUB_URL

# Default tape device
ARCHIVE_TDEVICE=/dev/rmt/0n; export ARCHIVE_TDEVICE

# Tape device capacity in Mb
ARCHIVE_CAPACITY=40000; export ARCHIVE_CAPACITY

# Default ctl file save location
ARCHIVE_CTL_LOC=${ARCHIVE_STUB_DIR}; export ARCHIVE_CTL_LOC

# Timestamp flag: 1=on 0=off
#ARCHIVE_TIMESTAMPS=1; export ARCHIVE_TIMESTAMPS

# Neutral format SEG Y header override (default "21 32 25 32")
#ARCHIVE_SEGY_STORED="193 32 197 32"; export ARCHIVE_SEGY_STORED

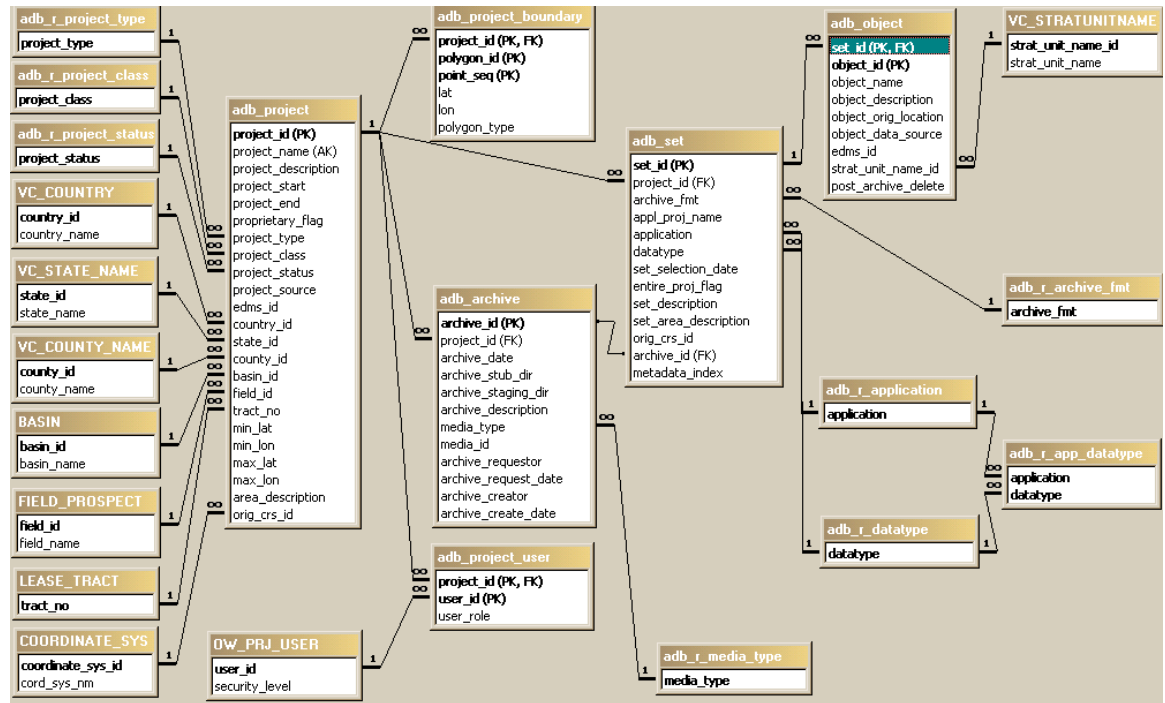
# Retriever GUI delete capability: 1=on 0=off
#ARCHIVE_DELETE=0; export ARCHIVE_DELETE

# EMC Centera IP address
CENTERA_IP_ADDRESS=128.221.200.56; export CENTERA_IP_ADDRESS

# Centera PEA file
CENTERA_PEA_LOCATION=/data/dev/C/Centera_SDK/pea_files/us1_profile2_rwqedp.pea
export CENTERA_PEA_LOCATION
```

Appendix 2: The Archive Database (ADB)

This component comprises the archive data model, created within a designated OpenWorks project. The advantage of using OpenWorks is that it provides fully serviceable cartographic, document, lease, field, company, basin and other tables as look-ups or reference tables. The data model provides coverage of project data to object level, e.g. seismic volume, horizon, fault, well etc. The data model is described below.



The model is designed for incorporation into a single OpenWorks project database. All new tables are displayed in lowercase with prefix **adb_**. New reference tables have a prefix of **adb_r_**. Existing OpenWorks tables are shown in uppercase. For simplicity only relevant attributes of OpenWorks tables are shown. For the main tables, keys are annotated as follows: PK – primary key; AK – alternate key; FK – foreign key.

Project table

A 'project' is defined in a business sense rather than in physical terms, i.e. it is a collection of data across multiple applications, databases and data types. The main **adb_project** table contains attributes pertaining to the entire project and does not presuppose any archival activity.

The first set of attributes describe the project:

project_id	internal identifier
project_name	full text name
project_description	long text description of the project
project_start	start interpretation date
project_end	end interpretation date
project_type	validated, e.g. regional exploration, bid round, well proposal

project_class	validated, e.g. public (approved), private (submitted by user)
project_status	validated, e.g. online, offline
project_source	source, e.g. 'Texaco'
proprietary_flag	Y if proprietary, N if public
edms_id	pointer to project-level documentation

The next set of attributes describe the location of the project:

country_id	validated from OW
state_id	validated from OW
county_id	validated from OW
basin_id	validated from OW
field_id	field or prospect name, validated from OW
tract_no	quad/block etc., validated from OW
min/max lat/lon	geographic limits
area_description	free text description of area
orig_crs_id	CRS id of the original project, validated from OW

The final set of attributes track audit information: create and update date, user and comments. Although not annotated on the ER diagram, these 5 attributes will be used throughout the main data model.

Set table

A project is composed of multiple data sets, stored in table **adb_set**. The key field is:

set_id	auto-created primary key
--------	--------------------------

The required fields are:

project_id	as before
datatype	validated, e.g. well, deviation, checkshot, pick, log, seismic
archive_fmt	validated, native or neutral
application	validated, e.g. SeisWorks, Hampson-Russell
appl_proj_name	project name for the specified application
set_selection_date	the date on which the set was created

Other dependent attributes are:

entire_project_flag	flag indicating if entire project (if neutral format)
set_description	free text description of the archive set
set_area_description	free text description of area
orig_crs_id	CRS id of the original project, validated from OW
archive_id	optional foreign key to adb_archive
metadata_index	pointer to the online metadata stub for that set

A special datatype of 'Undifferentiated' will be allowed for sites implementing a simple native-format project backup scheme.

Object Table

Each data set is composed of multiple objects. The purpose of the **adb_object** table is to store object-level archive metadata, e.g. every horizon in a SeisWorks project, seismic file in a Hampson-Russell project etc.

Attributes are:

set_id	part of primary key
object_id	part of primary key - unique in the originating system
object_name	additional identifier
object_description	free text description of the object
object_orig_location	original location on disk
object_data_source	free text source description
edms_id	pointer to project-level documentation
strat_unit_name_id	validated stratigraphy from OpenWorks
post_archive_delete	flag indicating if object should be deleted

Project child tables

In addition to the attributes described above, a single project must also track multiple users (interpreters) and a project boundary polygon.

The table **adb_project_user** is a simple intersection between **adb_project** and the OpenWorks **OW_PRJ_USER** table. This allows multiple users to be associated with the project. The table has a single dependent attribute: **user_role**, which allows the user's project role (e.g. geophysicist, data loader, team leader) to be tracked.

The table **adb_project_boundary** mimics the OpenWorks lease, field and basin boundary tables, except that provision is made to store only geographic coordinates. This model allows for an outline composed of multiple polygons, of type inclusive or exclusive.

Archive table

Project data sets are optionally archived using table **adb_archive**. Each project can have zero, one or many archives, although by common usage projects will have a single archive. This means that a project can be created in advance of any archive activity – the ADB could track all current and archived projects. A combination of project + description makes an archive unique. Attributes are:

archive_id	primary key to uniquely identify the archives
project_id	foreign key for project
archive_description	archive description
archive_requestor	who requested the archive be made
archive_request_date	date of request
archive_creator	who creates the archive
archive_create_date	date of archive
archive_stub_dir	location of the online metadata extraction
archive_stage_dir	location of the pre-archive staging directory

media_type validated, e.g. Exabyte, PARS, DLT etc.
media_id identifier for the archive system, e.g. PARS id

Reference Tables

The ADB uses 8 new reference tables in addition to 9 existing OpenWorks tables for validation: adb_r_archive_fmt , adb_r_application, adb_r_datatype, adb_r_app_datatype, adb_r_media_type, adb_r_project_class, adb_r_project_status, adb_r_project_type. All of these are simple single-column validation lists, with the exception of adb_r_app_datatype, which is an intersection table between adb_r_application and adb_r_datatype.

Views

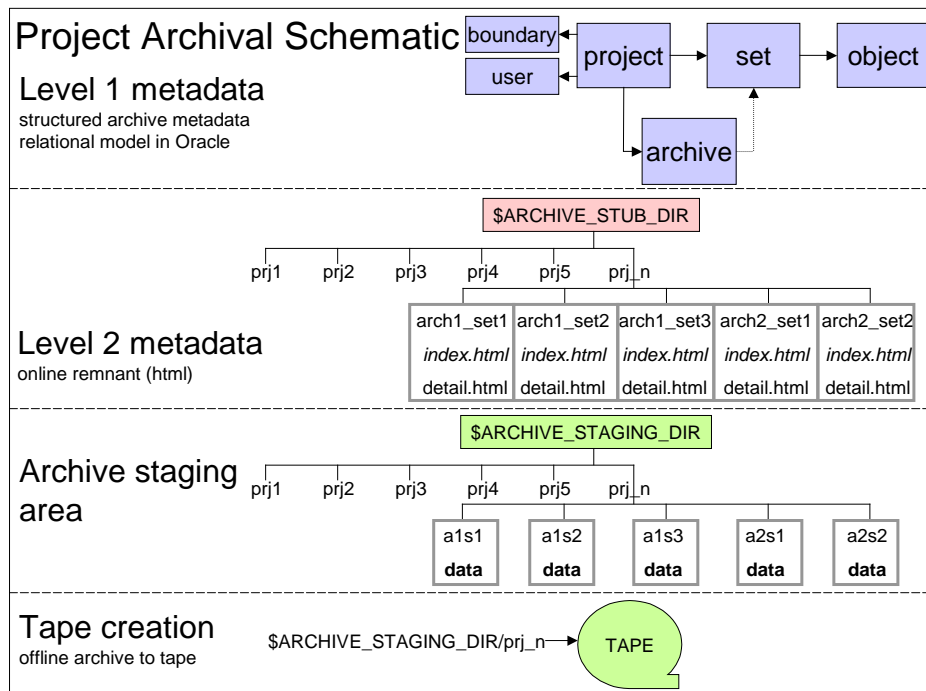
The following views will be used to simplify data access and query:

adb_v_project adb_project + vc tables
adb_v_archive adb_project + adb_archive
adb_v_set adb_project + adb_archive + adb_set
adb_v_object adb_set + adb_object

Data Access

All Oracle data access is via the Tcl/Tk 'oratcl' extension. This provides scripted access to the Oracle Call Interface (OCI). OCI provides stable and rapid select, insert, update and delete functionality to the ADB tables stored in Oracle. This mechanism is identical to the direct Oracle access mechanisms used in WOW.

This Oracle data model provides the 'level 1' metadata. See the diagram below for how this links with the 'level 2' online remnant or 'stub'.



The link between levels 1 and 2 is provided by a pointer in the Oracle data model to a location of an *index.html* file. This file in turn will reference all detailed metadata, to any level of detail, as determined by the relevant application. Locations are relative to a system environmental variable `$ARCHIVE_STUB_DIR`, to facilitate the IT aspects of managing large disk arrays.

The advantage of this approach is that it removes the need to model the universe of all applications and objects to a consistent level of detail. Application archive metadata can be at varying levels of detail and may be upgraded or improved without impacting other applications. With a smaller core relational database, data model changes and hence maintenance over time is considerably simplified. Lock-in to the developing vendor is reduced. In addition, storing metadata in Web format opens up a host of opportunities for implementing horizontal search tools, integrating with EDMS etc.

Appendix 3: Moving Archiver Data into a new OpenWorks Project

CDA utilizes an OpenWorks project to store its archive database. This project is recorded in variable `ARCHIVE_OW_PROJECT` in `$OWHOME/WebApps/conf/wow.env`. This section describes how to copy records from an existing archive project `ARCHIVES` to a new Archive OpenWorks project `NEWARCH` on the same OpenWorks Oracle instance indicated by `$ORACLE_SID`.

1. Login to OpenWorks as a user with the `MANAGE` role on both `ARCHIVES` and `NEWARCH` and open a Terminal Window off the OpenWorks launcher.
2. Copy records from old to new project:

```
cd $OWHOME/dat/project5000.0
sqlplus /@$ORACLE_SID @copyAdbDataR5000 ARCHIVES NEWARCH
```
3. Change value of `ARCHIVE_OW_PROJECT` in `wow.env`

```
cd $OWHOME/WebApps/conf
vi wow.env
ARCHIVE_OW_PROJECT=NEWARCH; export ARCHIVE_OW_PROJECT
```

Important note: The `copyAdbDataR5000.sql` script is designed primarily for a 1-off migration of archive data from an old to a new OpenWorks project. It is NOT designed as a general purpose synchronization mechanism. Records may exist in the destination OpenWorks project, but if the archive project name exists in both, it will be skipped.

Contacting Landmark Customer Support

Landmark software operates Technical Assistance Centers (TACs) in Australia, the United Kingdom, and the United States. Additional support is also provided through local support offices around the world.

Support via Web Portal

Support information is always available on the Landmark Customer Support internet page. You can also submit a support request directly to Landmark Customer Support through the Landmark Customer Support Portal:

<http://css.lgc.com/InfoCenter/index?page=home>

To request support in the Landmark Customer Support Portal:

1. In the **PIN** and **Password** text boxes in the Please Sign In area, enter your registered personal identification number and password.
2. Click the **Sign In** button.
3. In the Case & Defect Information area, click the **Create a New Case** link.
4. In the **Create Case** area, fill in the necessary information. Provide details about your technical concern, including any error messages, the workflow steps where the problem occurred, and attachments of screen shots that display the problem. To help understand the concern, you can also attach other files too, such as example data files.
5. Click the **Submit** button. A support analyst in the nearest Technical Assistance Center will respond to your request.

Technical Assistance Centers

Asia, Pacific

8:00 am - 5:00 pm Local Time
Monday-Friday, excluding holidays

61-8-9481-4488 (Perth, Australia)

Toll Free 1-800-448-488
Fax: 61-8-9481-1580
Email: apsupport@lgc.com

Europe, Africa, Middle East

9:00 am - 5:30 pm Local Time
Monday - Friday, excluding holidays

44-1372-868686 (Leatherhead, UK)

Fax: 44-1224-723260 (Aberdeen, UK)
Fax: 44-1372-868601 (Leatherhead, UK)
Email: support@lgc.com

Latin America

(Spanish, Portuguese, English)
7:00 am - 5:00 pm Local Time

713-839-3405 (Houston, TX, USA)

Fax: 713-839-3646
Email: soporte@lgc.com

North America

7:30 am - 5:30 pm Central Standard Time
Monday - Friday, excluding holidays

713-839-2200 (Houston, TX, USA)

Toll Free 1-877-435-7542
(1-877-HELP-LGC)
Fax: 713-839-2168
Email: support@lgc.com

Regional Offices

For contact information for regional offices, see the Contact Support page located at:

<http://css.lgc.com/InfoCenter/index?page=contact§ion=contact>

If problems cannot be resolved at the regional level, an escalation team is called to resolve your incidents quickly.

© 2013 Halliburton
All Rights Reserved

Information in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Halliburton. Unauthorized reproduction or distribution of this documentation, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under law.

Halliburton may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. The furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Trademarks

3D Drill View, 3D Drill View KM, 3D Surveillance, 3DFS, 3DView, Active Field Surveillance, Active Reservoir Surveillance, 3D Drill View, 3D Drill View KM, 3D Surveillance, 3DFS, 3DView, Active Field Surveillance, Active Reservoir Surveillance, Adaptive Mesh Refining, ADC, Advanced Data Transfer, Analysis Model Layering, ARIES, ARIES DecisionSuite, Asset Data Mining, Asset Decision Solutions, Asset Development Center, Asset Development Centre, Asset Journal, Asset Performance, AssetConnect, AssetConnect Enterprise, AssetConnect Enterprise Express, AssetConnect Expert, AssetDirector, AssetJournal, AssetLink, AssetLink Advisor, AssetLink Director, AssetLink Observer, AssetObserver, AssetObserver Advisor, AssetOptimizer, AssetPlanner, AssetPredictor, AssetSolver, AssetSolver Online, AssetView, AssetView 2D, AssetView 3D, BLITZPAK, CasingLife, CasingSeat, CDS Connect, Channel Trim, COMPASS, Contract Generation, Corporate Data Archiver, Corporate Data Store, Data Analyzer, DataManager, DataStar, DBPlot, Decision Management System, DecisionSpace, DecisionSpace 3D Drill View, DecisionSpace 3D Drill View KM, DecisionSpace AssetLink, DecisionSpace AssetPlanner, DecisionSpace AssetSolver, DecisionSpace Atomic Meshing, DecisionSpace Desktop, DecisionSpace Nexus, DecisionSpace Reservoir, DecisionSuite, Deeper Knowledge, Broader Understanding, Depth Team, Depth Team Explorer, Depth Team Express, Depth Team Extreme, Depth Team Interpreter, DepthTeam, DepthTeam Explorer, DepthTeam Express, DepthTeam Extreme, DepthTeam Interpreter, Desktop Navigator, DESKTOP-PVT, DESKTOP-VIP, DEX, DIMS, Discovery, Discovery 3D, Discovery Asset, Discovery Framebuilder, Discovery PowerStation, DMS, Drillability Suite, Drilling Desktop, DrillModel, Drill-to-the-Earth-Model, Drillworks, Drillworks ConnectML, DSS, Dynamic GeoModeling, Dynamic Reservoir Management, Dynamic Surveillance System, EarthCube, EDM, EDM AutoSync, EDT, eLandmark, Engineer's Data Model, Engineer's Desktop, Engineer's Link, ESP, Event Similarity Prediction, ezFault, ezModel, ezSurface, ezTracker, ezTracker2D, FastTrack, Field Scenario Planner, FieldPlan, For Production, FrameBuilder, FZAP!, GeoAtlas, GeoDataLoad, GeoGraphix, GeoGraphix Exploration System, GeoLink, Geometric Kernel, GeoProbe, GeoProbe GF DataServer, GeoSmith, GES, GES97, GESXplorer, GMAPlus, GMI Imager, Grid3D, GRIDGENR, H. Clean, Handheld Field Operator, HHFO, High Science Simplified, Horizon Generation, I² Enterprise, iDIMS, Infrastructure, Iso Core, IsoMap, iWellFile, KnowledgeSource, Landmark (*as a service*), Landmark (*as software*), Landmark Decision Center, Landmark Logo and Design, Landscape, Large Model, Lattix, LeaseMap, LithoTect, LogEdit, LogM, LogPrep, Magic Earth, Make Great Decisions, MathPack, MDS Connect, MicroTopology, MIMIC, MIMIC+, Model Builder, NETool, Nexus (*as a service*), Nexus (*as software*), Nexus View, Object MP, OpenBooks, OpenJournal, OpenSGM, OpenVision, OpenWells, OpenWire, OpenWire Client, OpenWire Server, OpenWorks, OpenWorks Development Kit, OpenWorks Production, OpenWorks Well File, PAL, Parallel-VIP, Parametric Modeling, Permedia, PetroBank, PetroBank Explorer, PetroBank Master Data Store, PetroStor, PetroWorks, PetroWorks Asset, PetroWorks Pro, PetroWorks ULTRA, PlotView, Point Gridding Plus, Pointing Dispatcher, PostStack, PostStack ESP, PostStack Family, Power Interpretation, PowerCalculator, PowerExplorer, PowerExplorer Connect, PowerGrid, PowerHub, PowerModel, PowerView, PrecisionTarget, Presgraf, PressWorks, PRIZM, Production, Production Asset Manager, PROFILE, Project Administrator, ProMAGIC, ProMAGIC Connect, ProMAGIC Server, ProMAX, ProMAX 2D, ProMax 3D, ProMAX 3DPSDM, ProMAX 4D, ProMAX Family, ProMAX MVA, ProMAX VSP, pSTax, Query Builder, Quick, Quick+, QUICKDIF, Quickwell, Quickwell+, Quiklog, QUIKRAY, QUIKSHOT, QUIKVSP, RAVE, RAYMAP, RAYMAP+, Real Freedom, Real Time Asset Management Center, Real Time Decision Center, Real Time Operations Center, Real Time Production Surveillance, Real Time Surveillance, Real-time View, Reference Data Manager, Reservoir, Reservoir Framework Builder, RESev, ResMap, RTOC, SCAN, SeisCube, SeisMap, SeisModel, SeisSpace, SeisVision, SeisWell, SeisWorks, SeisWorks 2D, SeisWorks 3D, SeisWorks PowerCalculator, SeisWorks PowerJournal, SeisWorks PowerSection, SeisWorks PowerView, SeisXchange, Semblance Computation and Analysis, Sierra Family, SigmaView, SimConnect, SimConvert, SimDataStudio, SimResults, SimResults+, SimResults+3D, SIVA+, SLAM, SmartFlow, smartSECTION, smartSTRAT, Spatializer, SpecDecomp, StrataAmp, StrataMap, StrataModel, StrataSim, StratWorks, StratWorks 3D, StreamCalc, StressCheck, STRUCT, Structure Cube, Surf & Connect, SurfNet, SynTool, System Start for Servers, SystemStart, SystemStart for Clients, SystemStart for Servers, SystemStart for Storage, Tanks & Tubes, TDQ, Team Workspace, TERAS, T-Grid, The Engineer's DeskTop, Total Drilling Performance, TOW/cs, TOW/cs Revenue Interface, TracPlanner, TracPlanner Xpress, Trend Form Gridding, Trimmed Grid, Turbo Synthetics, Unconventional Essentials, VESPA, VESPA+, VIP, VIP-COMP, VIP-CORE, VIPDataStudio, VIP-DUAL, VIP-ENCORE, VIP-EXECUTIVE, VIP-Local Grid Refinement, VIP-THERM, WavX, Web Editor, Well Cost, Well H. Clean, Well Seismic Fusion, Wellbase, Wellbore Planner, Wellbore Planner Connect, WELLCAT, WellDirect, WELLPLAN, WellSolver, WellXchange, WOW, Xsection, You're in Control. Experience the difference, ZAP!, and Z-MAP Plus are trademarks, registered trademarks, or service marks of Halliburton.

All other trademarks, service marks and product or service names are the trademarks or names of their respective owners.

Note

The information contained in this document is subject to change without notice and should not be construed as a commitment by Halliburton. Halliburton assumes no responsibility for any error that may appear in this manual. Some states or jurisdictions do not allow disclaimer of expressed or implied warranties in certain transactions; therefore, this statement may not apply to you.