

Coraid EtherDrive[®]

VS

Command Reference Manual

Update 5/18/09

*Coraid, Inc.
220 College Ave, STE 7S
Athens, GA 30601
<http://www.coraid.com>*

NAME

VS – VirtualStorage Appliance

DESCRIPTION

This introduction documents basic setup and quick start examples for the VS. Please see Appendix A for a description of the VS firmware update procedure. Console connectivity to the VS is available by keyboard+video, CEC, or serial connection. CEC is enabled on all available VS interfaces. A serial connection can be established at 9600 baud, 8-N-1.

VS firmware releases and documentation are available at the VS support page at coraid.com. A low traffic mailing list for VS related firmware and documentation release announcements exists. E-mail support@coraid.com and ask to be placed on the VS Announcement mailing list.

The VS is a storage virtualization appliance. It allows the storage administrator to flexibly allocate storage capacity as needed, with full redundancy as needed. The VS uses AoE storage as its raw materials, constructing new AoE targets that have the desired size and redundancy characteristics.

The raw storage materials of the VS are 4MB chunks called extents. The extents reside on physical volumes, storage targets prepared for use with the VS. A pool of extents is called a volume group. The VS creates usable storage targets called *LUNs* by allocating extents from a volume group.

A LUN can be mirrored to provide redundancy across multiple physical volumes. For example, an SR has internal redundancy by virtue of its RAID configuration, but it only has one processor. Two SRs are fully redundant, and when the VS mirrors across them, the resulting LUN is resistant not only to a hard disk failure but even to a processor failure in one of the SRs.

This manual documents the commands used to initialize and examine physical volumes, volume groups, LUNs, and mirrors. The most commonly used commands are listed below.

<i>lsoe</i>	display information about AoE targets
<i>lslun</i>	display information about existing LUNs
<i>lspv</i>	display information about physical volumes
<i>lsvg</i>	display information about volume groups
<i>mirror</i>	establish mirroring between two physical volumes
<i>mklegacy</i>	add target to a volume group so that existing data remains accessible
<i>mkln</i>	create a storage target
<i>mkpv</i>	initialize an AoE target for use as a physical volume
<i>mkvg</i>	create a volume group
<i>offline</i>	deactivate a LUN
<i>online</i>	activate a LUN
<i>rmlun</i>	remove a LUN
<i>rmpv</i>	stop using a physical volume
<i>rmvg</i>	remove a volume group
<i>setboot</i>	choose which flash disk area to boot from, A or B
<i>shelf</i>	show or set the VS shelf address
<i>srvshelf</i>	show or set the VS service shelf address
<i>unmirror</i>	break the mirroring relationship between two physical volumes

EXAMPLE**CREATING LUNS**

The following quickstart example sets up the VS with SRs that do not already contain user data.

Set shelf addresses:

```
VS-1:-1> shelf 16
VS16:-1> srvshelf 18
VS16:18>
```

Create volume group:

```
VS16:18> mkvg storage
```

Create physical volumes, which assumes there is a SR on the network with shelf number 9:

```
VS16:18> mkpv 9.6 storage
```

Create the LUN:

```
VS16:18> mklun 0 10g storage
```

Put LUN online:

```
VS16:18> online 0
```

Now all AoE clients will be able to access the storage located at shelf address 18 slot 0, where 18 is the *srvshelf* number and 0 is the LUN number.

CREATING LUNS WITH A LEGACY SR

In order to use existing *legacy* SR storage on the VS, there are a couple of special steps. These steps must be done to protect data stored on the SR.

Before adding the legacy storage, a new physical volume must be available. The reason for the new physical volume is a location for the *metadata* associated with the LUN. The metadata is the information the VS uses to maintain state about the storage it is using. Since the legacy SRs do not have a place to store that locally, a new physical volume must be added. The new physical volume could be a new *LUN* on an existing SR.

Follow these steps to bring the current SR shelf 9 with new SR shelf number 7. First, set the shelf addresses for the VS:

```
VS-1:-1> shelf 16
```

```
VS16:-1> srvshelf 18
```

```
VS16:18>
```

Create the volume group:

```
VS16:18> mkvg existing
```

Create the physical volume for where *metadata* will be stored:

```
VS16:18> mkpv 7.2 existing
```

Create the LUN:

```
VS16:18> mklegacy 10 9.15 existing
```

Put LUN online:

```
VS16:18> online 10
```

Now all AoE clients will be able to access the storage located at AoE address 18.10, where 18 is the *srvshelf* number and 10 is the LUN number. For additional relevant information on preparing a legacy volume for use prior to using *mklegacy*, please see the command man page for *mklegacy*.

SEE ALSO

For more information on the basic concepts of network storage virtualization, see http://en.wikipedia.org/wiki/Storage_virtualization

For a discussion of EtherDrive VS concepts see the paper *VS Principles of Operation* located at the VS support page at coraid.com.

NAME

aoediscover, aoeflush – manage local AoE target list

SYNOPSIS

```
aoediscover
aoeflush [ -a ] [ shelf.slot ... ]
```

DESCRIPTION

Aoediscover sends a broadcast AoE query configuration message onto the network to discover new AoE devices on the network.

Aoeflush removes AoE targets from the list of known targets. When supplied with the `-a` flag, all targets are removed from the list.

The list of discovered AoE devices can be viewed with the *lsaoe* command.

EXAMPLE

```
VS66:63> lsaoe
TARGET          LENGTH    STATE  PORT      ADDR
61.0            5001.078GB  V      0    00304833f6aa
61.0            5001.078GB  V      1    00304833f6ab
61.1            5001.078GB  V      0    00304833f6aa
61.1            5001.078GB  V      1    00304833f6ab
61.2             0.040GB    M      0    00304833f6aa
61.2             0.040GB    M      1    00304833f6ab
VS66:63> aoeflush 61.2
VS66:63> lsaoe
TARGET          LENGTH    STATE  PORT      ADDR
61.0            5001.078GB  V      0    00304833f6aa
61.0            5001.078GB  V      1    00304833f6ab
61.1            5001.078GB  V      0    00304833f6aa
61.1            5001.078GB  V      1    00304833f6ab
VS66:63>
```

DIAGNOSTICS

A periodic broadcast query config message is sent to automatically discover targets on the network. It is possible to flush a target and have it reappear seconds later.

SEE ALSO

lsaoe(1)

NAME

clrshelves - clear shelf addresses and reboot

SYNOPSIS

clrshelves

DESCRIPTION

Clrshelves will clear both the shelf and srvshelf addresses. When executed, *clrshelves* asks for a confirmation from the user and when received, it automatically reboots the unit. After reboot, the shelf and srvshelf are both unset and the prompt displays "-1" for both, as when the unit was shipped.

SEE ALSO

shelf(1)

NAME

clstat - display cluster node status

SYNOPSIS

clstat

DESCRIPTION

Clstat displays the master/slave status of each node in the cluster along with the VS release the cluster node is running. Each VS's admin shelf address is reported along with its role, M for master and S for slave, and the firmware release.

EXAMPLE

```
VS66:63> clstat
SHELF  ROLE   RELEASE
66      M    20081028
65      S    20081027
VS66:63>
```

NAME

copy - copy one logical-volume to another logical-volume

SYNOPSIS

copy luna lunb

DESCRIPTION

Copy initiates a background job to copy the entire data contents from *luna* to *lunb*. The status of the background LUN copy can be viewed with the *wstat* command. A log message is generated when the background copy is initiated.

DIAGNOSTICS

Both LUNs must already exist and use the same extent size. The LUN extent size is defined by the volume group from which the LUNs are defined and is viewed with the *lsvg* command. Additionally, *lunb* should not be smaller than *luna* or the copy will ultimately fail. This behavior is not restricted, however, in the event that it proves useful.

LUN copying is very simple and does not ensure two LUNs remain in sync. The copy process merely reads every extent from the source LUN and writes it to the destination LUN. If either LUN is written during the copy, the LUNs may not be identical when the copy completes. It is recommended that users take both source and destination LUNs offline when using this feature.

SEE ALSO

lsvg(1), *pause(1)*, *resume(1)*, *wstat(1)*

NAME

counters – display vs counters

SYNOPSIS

counters

DESCRIPTION

Counters displays VS counters for various system statistics.

The statistics are presented as running totals and are defined as follows.

msgs – AoE commands received from clients.

wronglun – received AoE commands for LUNs that do not exist or are offline.

toomanysectors – received AoE commands requesting more data than can fit in a response, or requesting more data than exists on a LUN (I/O off the end of the device).

badflash – received AoE commands that were dropped due to an invalid flash signature.

intrxlate – received AoE commands that were handled with the fast translation path.

noaarp – received AoE commands that could not be translated to a backing target due to a missing target.

queued – received AoE commands that were queued for processing and could not be handled by the fast translation path.

bcast – received AoE commands that were sent to a broadcast shelf or slot address.

lunoffline – received AoE commands to an offline LUN that were dropped.

xprocbadlun – received AoE commands that were queued for processing, but sent to a nonexistent LUN.

wrongdiskcmd – received AoE commands that were queued for processing, but sent with an unrecognized AoE ATA command.

wrongaoecmd – received AoE commands that were queued for processing, but sent with an unrecognized AoE command.

opfailed – received AoE commands that were queued for processing, but could not complete due to I/O error.

procxlate – received AoE commands that were queued for processing and successfully processed.

nsplits – received AoE ATA commands that spanned two extents and had to be split into two requests to complete.

resend – AoE commands initiated from the VS that had to be retransmitted.

badtag – received AoE responses that could not be resolved to an outstanding command tag.

cmdfailed – VS initiated AoE commands that failed due to retransmission timeout.

noleader – AoE commands dropped during an election due to no established leader.

masked – AoE commands dropped due to failed mask list validation.

DATE(1)

DATE(1)

NAME

date – show or set date and time

SYNOPSIS

date [*YYYYMMDDhhmm*]

DESCRIPTION

Date is used to print or set the date. If no argument is given, the date will be printed in the following format:

Sat Mar 22 12:13:42 2008

To set the date, specify the Year, Month, Day, Hour, and Minute of the time using two digits for each field. At this time, time zone is not used and time is assumed to be set to local time.

EXAMPLE

```
VS66:63> date
Mon Oct 27 16:46:48 2008
VS66:63> date 200803221215
Sat Mar 22 12:15:00 2008
VS66:63> date
Sat Mar 22 12:15:02 2008
VS66:63>
```

NAME

elstat - display EL connection status

SYNOPSIS

elstat

DESCRIPTION

Elstat displays the EL (Ethernet Link) connections on the local VS. EL connections are only established from slave to master. Each EL connection is assigned an ECB (EL Control Block) number and is distinguished by a foreign and local address and port. The address is the admin shelf address.

In the example below the master displays an ECB 0 used to listen for inbound slave connections, and an ECB 1 for slave 65.

EXAMPLE

```
VS66:63> elstat
ECB  FADDR  FPORT  LADDR  LPORT  STATE
0      0      0      66     220  listen, Closed
1     65    1027    66     220  open, Estab
VS66:63>
```

EXIT(1)

EXIT(1)

NAME

exit - console logout

SYNOPSIS

Exit

DESCRIPTION

Exit logs out the console connection and returns to the login prompt.

NAME

grow, shrink – resize a logical-volume

SYNOPSIS

grow lun newsize [pv ...]

shrink lun newsize

DESCRIPTION

Grow and *shrink* change the size of the specified *lun*. Both accept a *newsize* argument specifying the total size of the LUN after grow/shrink.

Grow accepts one or more optional physical volume *pv* parameters. When provided, the LUN will be grown using the specified physical volume(s). If multiple physical volumes are specified, extent allocation is round robin across the specified PVs as in the *mklun* command. The physical volume(s) must reside in the volume group from which the LUN is already allocated.

Shrinking the LUN reduces the size of the LUN by deallocating extents at the end of the LUN, freeing them back to the PV(s) they belong to. The VS is not able to know whether this storage is used or needed by client systems. Administrators are advised to explore their filesystem/database utilities to properly shrink usage of the LUN before using the shrink command. Once a LUN has been shrunk, the freed extents are lost.

EXAMPLE

```
VS66:63> mklun 0 500m vg0
updating 61.1
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            0.503GB   OFF         63.0
VS66:63> grow 0 1g
updating 61.1
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            1.002GB   OFF         63.0
VS66:63> shrink 0 250m
Shrink LUN 0 to 250,000,000 bytes? [n]: y
updating 61.1
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            0.251GB   OFF         63.0
VS66:63>
```

SEE ALSO

lsvg(1), *mkvg(1)*

NAME

halt, reboot - shutdown and optionally wait for key entry to reboot

SYNOPSIS

halt

reboot

DESCRIPTION

Halt performs a system shutdown and waits for a key to be pressed. Upon keypress, the VS performs a hardware reboot.

Reboot behaves similarly, but immediately reboots after system shutdown.

BUGS

Halt currently takes down the network as part of its shutdown procedure. As a result, users that run halt via CEC will not be able to send another keystroke to cause the VS to reboot. CEC users should not use the halt command until this is resolved.

NAME

help - print command usage

SYNOPSIS

help [*cmd*]

DESCRIPTION

Help prints the command usage and a short description for a specified *cmd*. Without any arguments, *help* displays all available commands.

NAME

ifstat - display network interface status

SYNOPSIS

ifstat [-a] [port ...]

DESCRIPTION

Ifstat displays information about the network interfaces. If no network *port* is specified, all available ports are displayed.

If the *-a* flag is supplied, detailed statistics about the port(s) are displayed.

EXAMPLE

```
VS66:63> ifstat
PORT          ADDR          LINK  TYPE
0      0030488f35d6    1000/1000  i563
1      0030488f35d7    1000/1000  i563
VS66:63> ifstat -a 0
PORT          ADDR          LINK  TYPE
0      0030488f35d6    1000/1000  i563
Packets Received (64 Bytes): 14403029 72358
Packets Received (65-127 Bytes): 371 0
Packets Received (512-1023 Bytes): 364 0
Packets Received (1024-1522 Bytes): 14401211 72339
Good Packets Received: 28804977 144699
Broadcast Packets Received: 534 4
Good Packets Transmitted: 28804361 144694
Good Octets Received: 18446744072923265341 27
Good Octets Transmitted: 18446744072918815054 27
Total Octets Received: 18446744072923265341 27
Total Octets Transmitted: 18446744072918815054 27
Total Packets Received: 28804977 144699
Total Packets Transmitted: 28804361 144694
Packets Transmitted (64 Bytes): 14403666 72350
Packets Transmitted (65-127 Bytes): 1 0
Packets Transmitted (1024-1522 Bytes): 14400694 72344
Broadcast Packets Transmitted: 286 1
rdfree=405 (405)
tdfree=1024
nrd=512
more...
VS66:63>
```

EXAMPLE CURTAILED FOR BREVITY

NAME

kill - kill background job

SYNOPSIS

kill *workid*

DESCRIPTION

Kill stops the background work specified by *workid*. Work ids are assigned when background work is initiated and are displayed by the *wstat* command.

If a LUN *copy* is terminated, there is no way to restart it at the point it was terminated.

If a physical volume mirror rebuild is terminated, the mirror is broken. This is equivalent to using *unmirror* to break the mirror.

EXAMPLE

```
VS66:63> wstat
ID          ACTION      DONE      RATE      TTC      STATE      NP
0          mirror 61.0->61.1  5.81%    347.30MB/s  7:11:29      R      64
VS66:63> kill 0
2008.03.22 15:41:13 work 0 killed by user
VS66:63>
2008.03.22 15:41:13 mirror rebuild killed: 61.0 -> 61.1
updating 61.0

VS66:63> wstat
VS66:63>
```

SEE ALSO

pause(1), *resume(1)*, *wstat(1)*

NAME

label - set label for LUN

SYNOPSIS

label *lun label*

DESCRIPTION

Label associates a textual label with a specified *lun*. The specified *label* is a text string without spaces. The LUN label is displayed in the output of the *lslun* command.

EXAMPLE

```
VS66:63> mklun 0 5g vg0
updating 61.1
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            5.003GB  OFF         63.0
VS66:63> label 0 accounting
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            5.003GB  OFF         63.0  accounting
VS66:63>
```

SEE ALSO

lslun(1)

NAME

lsaoe - show the status of AoE targets

SYNOPSIS

lsaoe [-c]

DESCRIPTION

lsaoe displays information about the AoE targets on the network. If no arguments are used, *lsaoe* displays the following information for all of the visible AoE targets:

TARGET - the AoE shelf.slot target address

LENGTH - the size of the AoE target in GigaBytes

STATE - the VS communication status with the AoE target

The possible values for this field are:

G - the AoE target responded to an AoE query config and the VS is retrieving the size information from the target

V - the target has completed size and status initialization and is online for use

M - the target was previously visible and valid, but has stopped responding to the VS

PORT - the Ethernet port on which the AoE target is visible

ADDR - the Ethernet address of the AoE target

If a *-c* flag is provided, *lsaoe* displays the AoE config string information stored on the target. The output displays the following information for each AoE target:

TARGET - the AoE shelf.slot target address

LEN - the length of the stored configuration string

CONFIG - the configuration string stored on the target

Note in the example below that unlike the default output, *lsaoe -c* only displays one instance of each target regardless of the PORT it is on.

EXAMPLE

```
VS66:63> lsaoe
TARGET          LENGTH  STATE  PORT          ADDR
61.0            5001.078GB  V      0  00304833f6aa
61.0            5001.078GB  V      1  00304833f6aa
61.0            5001.078GB  V      0  00304833f6ab
61.0            5001.078GB  V      1  00304833f6ab
61.1            5001.078GB  V      0  00304833f6aa
61.1            5001.078GB  V      1  00304833f6aa
61.1            5001.078GB  V      0  00304833f6ab
61.1            5001.078GB  V      1  00304833f6ab
VS66:63> lsaoe -c
TARGET          LEN  CONFIG
61.0            0
61.1            58  'CoraidPV 63 61.1 -1.-1 1 2 1192349 vg0 1 64 61.1.1 61.1.2 '
```

NAME

lslun - list logical volumes

SYNOPSIS

lslun [-a] [lun ...]

DESCRIPTION

lslun displays the LUN(s) defined by the VS. One or more LUNs may be specified on the command line. The following information is displayed for each LUN:

LUN - the LUN ID

LENGTH - the size of the LUN in GigaBytes

ONLINE - the online/offline state of the LUN

TARGET - the LUN's target address as exported by the VS.

LABEL - the label assigned to the LUN (if any)

The *-a* flag lists extended information about the LUN including the physical volume(s) the LUN uses and the volume group from which the LUN is defined. Additional debugging information is listed that is useful to Coraid support; undocumented information can be ignored.

EXAMPLE

```
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            5.003GB  OFF         63.0   accounting
1            5.003GB  ON          63.1   accounting
15           0.008GB  ON          66.15  flash area B
VS66:63> lslun -a 0 1
LUN          LENGTH  ONLINE      TARGET  LABEL
0            5.003GB  OFF         63.0   accounting
           volume group: vg0
           pvs: 61.1

           flag: AoE_Target
           serial: 755357c5_1206189825
           levec=794c8fa0 nle=1194 g=79ce9d30 base=0 limit=0
           cfglen=0 snapshot=0 sslun=0
1            5.003GB  ON          63.1   accounting
           volume group: vg0
           pvs: 61.1

           flag: AoE_Target online wants_online
           serial: f53f17f8_1206190241
           levec=794ce3a0 nle=1194 g=79ce9d30 base=0 limit=0
           cfglen=0 snapshot=0 sslun=0
VS66:63>
```

SEE ALSO

label(1)

NAME

`lspv` - list physical volume(s)

SYNOPSIS

`lspv` [`-a`] [`pv ...`]

DESCRIPTION

`Lspv` displays the physical volumes in all volume groups managed by the VS. One or more PVs may be specified on the command line. The following information is displayed for each PV:

TARGET - the AoE shelf.slot target address

VOLUME GROUP - the volume group to which the PV belongs

LENGTH - the size of the PV in GigaBytes

AVAILABLE - the space available on the PV for VS allocation

NPE - the count of physical extents on the PV

MIRROR - the mirror of the PV, if any

The `-a` flag lists extended information about the PV including LUNs that use the PV. Additional debugging information is listed that is useful to Coraid support; undocumented information can be ignored.

EXAMPLE

```
VS66:63> lspv
TARGET          VOLUME GROUP      LENGTH      AVAILABLE      NPE  MIRROR
61.0             vg0                5001.074GB  5001.065GB     1192349
61.1             vg0                5001.074GB  4991.049GB     1192349
VS66:63> lspv -a 61.0
TARGET          VOLUME GROUP      LENGTH      AVAILABLE      NPE  MIRROR
61.0             vg0                5001.074GB  5001.065GB     1192349
    LUNs:
    metadata: 61.0.1 61.0.2
    nvec=1192349 data=0 written=0 dirty=0
    cow=0 changing=0 metadata=2 scanned=1
VS66:63>
```

NAME

`lsvg` - list volume group(s)

SYNOPSIS

`lsvg` [`-a`] [`vg ...`]

DESCRIPTION

`Lsvg` displays the volume groups defined by the VS. One or more volume group names may be specified on the command line. If no volume groups are specified, all volume groups are displayed. The following information is displayed for each VG:

NAME - the assigned name of the VG

LENGTH - the size of the VG in MegaBytes

AVAILABLE - the space available on the VG for VS allocation

EXTSZ - the extent size of the VG

PVS - the count of PVs currently available / the total count of PVs in the volume group

The `-a` flag lists extended information about the VG including currently available constituent PVs. Additional debugging information is listed that is useful to Coraid support; undocumented information can be ignored.

EXAMPLE

```
VS66:63> lsvg
```

NAME	LENGTH	AVAILABLE	EXTSZ	PVS
vg1	5001.074GB	5001.065GB	4MB	01/01
vg0	5001.074GB	4991.049GB	4MB	01/02

```
VS66:63> lsvg -a vg1
```

NAME	LENGTH	AVAILABLE	EXTSZ	PVS
vg1	5001.074GB	5001.065GB	4MB	01/01

PVS: 61.0

LUNS:

gen=1 eshift=13 emask=8191 emult=1

```
VS66:63>
```

SEE ALSO

`mkvg(1)`

NAME

mask - manage lun masks

SYNOPSIS

```
mask [ lun [ +mac ... ] [ -mac ... ] ]
```

DESCRIPTION

Administrators may limit client access to LUNs by assigning a LUN a mac mask list. By default, a LUN's mask list is empty permitting access to anyone on the network. If a LUN's mask list is not empty, only those mac addresses in the mask list will be permitted access.

Mask manages the mac mask list for a LUN. Without arguments, *mask* displays all LUNs and their mac mask lists. Given only a *lun* argument, *mask* displays the mac mask list for the specified *lun*. Further arguments are directives to add (+) or remove (-) mac addresses from the mask list.

EXAMPLE

```
VS66:63> mask
 0 0123456789ab
 1
VS66:63> mask 0 +cdef01234567
VS66:63> mask
 0 0123456789ab cdef01234567
 1
VS66:63> mask 0 -0123456789ab
VS66:63> mask
 0 cdef01234567
 1
VS66:63>
```

NAME

mirror, unmirror - manage physical-volume redundancy

SYNOPSIS

mirror *pva pvb*

unmirror *pv*

DESCRIPTION

Mirror creates a mirror from *pva* to *pvb* causing the two physical volumes to stay in sync. *Pva* is considered the primary and *pvb* is considered the mirror. If the primary of a mirrored PV becomes unavailable, the mirror will be promoted as the primary and the mirror will be broken.

Unmirror breaks a mirror, releasing the configured mirror volume for other use.

Upon creation, a rebuild of the mirror will be initiated to copy any dirty extents from *pva* to *pvb*. A log message is generated to indicate the rebuild has started. The status of the rebuild can be viewed with the *wstat* command. The *lspv* command displays the defined mirrors.

EXAMPLE

```
VS66:63> lsaoe
TARGET          LENGTH  STATE  PORT      ADDR
61.0            5001.078GB  V      0  00304833f6aa
61.0            5001.078GB  V      1  00304833f6ab
61.1            5001.078GB  V      0  00304833f6aa
61.1            5001.078GB  V      1  00304833f6ab
VS66:63> mkpv 61.0 vg0
updating 61.0
VS66:63> mirror 61.0 61.1
updating 61.0
2008.03.22 13:07:11 started rebuilding: 61.0 -> 61.1
VS66:63>
2008.03.22 13:07:12 mirror rebuild complete: 61.0 -> 61.1
updating 61.0
VS66:63> lspv
TARGET          VOLUME GROUP      LENGTH      AVAILABLE      NPE  MIRROR
61.0            vg0              5001.074GB   5001.065GB     1192349  61.1
VS66:63>
```

SEE ALSO

mkpv(1), *pause(1)*, *resume(1)*, *wstat(1)* *lspv(1)*

NAME

mklegacy – legacy create a volume group

SYNOPSIS

mklegacy *lun target vg*

DESCRIPTION

Mklegacy creates LUNs comprised of SR storage that already contains user data, permitting the use of legacy SR storage with the VS.

Lun is the LUN number to export the legacy volume. *Target* is the AoE shelf.slot target of the legacy physical volume. The physical volume will be imported into the specified volume group *vg*.

EXAMPLE

```
VS66:63> lspv
TARGET          VOLUME GROUP          LENGTH      AVAILABLE      NPE  MIRROR
61.0              vg0          5001.074GB  5001.065GB    1192349
VS66:63> mklegacy 0 61.1 vg0
updating 61.1
updating 61.0
VS66:63> lslun -a 0
LUN             LENGTH  ONLINE      TARGET  LABEL
0               5001.074GB  OFF         63.0
      volume group: vg0
      pvs: 61.0 61.1

      flag: AoE_Target
      serial: 5150f4e9_1206191462
      levec=773cdfa0 nle=1192350 g=794cdeb0 base=0 limit=0
      cfglen=0 snapshot=0 sslun=0
VS66:63> lspv
TARGET          VOLUME GROUP          LENGTH      AVAILABLE      NPE  MIRROR
61.0              vg0          5001.074GB  5001.053GB    1192349
61.1              vg0          5001.074GB   0.000GB      1192349
VS66:63>
```

In this example, the legacy LUN is being created for physical volume 61.1. The metadata for the legacy LUN is stored on physical volume 61.0.

DIAGNOSTICS

The *vg* must already exist and contain at least one physical volume with an allocatable extent for the legacy physical volume's metadata. Normally the metadata for a physical volume is stored on the physical volume itself, but this is not possible with legacy physical volumes. For a full explanation of the legacy physical volume support, please see the document *VS Principles of Operation*.

The following example sets up a legacy LUN.

BUGS

Due to the current implementation of the VS legacy feature some data at the end of the legacy device may be ignored. The VS addresses physical volumes in units of extents. The extent size is defined by the extent size of the volume group the physical volume resides in. When importing a pv into a volume group, the size of the pv is divided into extentsize units; up to extentsize – 1 remainder bytes at the end of the device are ignored. This is a problem for legacy physical volumes as these ignored bytes may contain user data.

To workaroud this problem users should ensure they are not using this extentsize – 1 remainder bytes on a legacy AoE target before making it a legacy physical volume. The simplest method for doing this is to shrink the use of the AoE target by extentsize bytes. The method for doing this will vary based on the use of the AoE target. Consult your filesystem/database documentation to see if a shrink utility is available.

SEE ALSO

lsaoe(1), *lsvg(1)*, *mkpv(1)*, *mkvg(1)*

NAME

`mkln` - create a logical volume

SYNOPSIS

`mkln lun size vg [pv ...]`

`mkln lun flash [A | B]`

DESCRIPTION

Mkln creates a logical volume backed by AoE storage or by flash boot images for updating the appliance.

Lun is the LUN to use in exporting the data. The valid LUN range is 0 - 254.

Size is the size of the logical volume being created. In specifying the size, the following suffixes can be used to denote the size.

m, M - Megabyte

g, G - Gigabyte

t, T - Terabyte

If no suffix is specified the size is assumed to be in Megabytes. *Vg* is the volume group to allocate the LUN from. The specified VG must have already been created using the *mkvg* command.

By default, LUN data is allocated from all physical volumes in *vg* in a round-robin fashion. The PVs used for allocation can be restricted by specifying the PV(s) to allocate from on the command line. Any specified physical volumes must reside in the specified *vg*.

Mkln can also be used to create a flash based LUN to update the appliance firmware. There are two firmware image areas on the flash, *A* and *B*. For more information on updating the VS appliance, see Appendix A.

EXAMPLE

Create LUN 0 of 100 Megabytes using volume group *vg1*:

```
mkln 0 100M vg1
```

Create LUN 10 of 20 Terabytes using volume group *vg7*:

```
mkln 10 20t vg7
```

Create LUN 6 backed by flash boot image *A*:

```
mkln 6 flash A
```

DIAGNOSTICS

Information about the LUN is stored in the Per LUN Data (PLD). If the PLD is not able to be written to its assigned physical volume, the following log message is displayed:

```
pldsave failed: shelf.slot error
```

Shelf.slot defines the physical volume target and *error* defines the error that occurred.

The LUN will be created, but will not survive reboot.

SEE ALSO

lslun(1), *mklegacy(1)*, *mkpv(1)*, *offline(1)*, *shelf(1)*

NAME

mkpv – add a physical volume to a volume group

SYNOPSIS

mkpv *target* *vg*

DESCRIPTION

Mkpv makes an AoE target a physical volume, adding it to a volume group. The *target* must not already be in use and must be listed as available in the output of *lsaoe*. The specified *vg* must already exist, having been created with the *mkvg* command.

EXAMPLE

```
VS66:63> lspv
TARGET          VOLUME GROUP      LENGTH      AVAILABLE      NPE  MIRROR
61.0              vg0      5001.074GB   5001.057GB     1192349
VS66:63> lsvg
NAME              LENGTH      AVAILABLE  EXTSZ    PVS
vg0              5001.074GB   5001.057GB   4MB    01/01
VS66:63> lsaoe
TARGET          LENGTH  STATE  PORT      ADDR
61.0            5001.078GB   V     0    00304833f6aa
61.0            5001.078GB   V     1    00304833f6ab
61.1            5001.078GB   V     0    00304833f6aa
61.1            5001.078GB   V     1    00304833f6ab
VS66:63> mkpv 61.1 vg0
updating 61.1
updating 61.0
VS66:63> lspv
TARGET          VOLUME GROUP      LENGTH      AVAILABLE      NPE  MIRROR
61.0              vg0      5001.074GB   5001.057GB     1192349
61.1              vg0      5001.074GB   5001.065GB     1192349
VS66:63> lsvg
NAME              LENGTH      AVAILABLE  EXTSZ    PVS
vg0             10002.148GB   10002.123GB   4MB    02/02
VS66:63>
```

SEE ALSO

lspv(1), *lsaoe*(1), *mkvg*(1)

NAME

mkvg - make a volume group

SYNOPSIS

mkvg *vg*

DESCRIPTION

Mkvg creates an empty volume group named by the parameter *vg*. The name of the volume group is limited to 32 characters.

EXAMPLE

Create the volume group *vg-test*:

```
mkvg vg-test
```

DIAGNOSTICS

If a serve shelf has not been set with the *srvshelf* command, the *mkvg* command will fail.

Only 4MB extents are currently supported.

SEE ALSO

shelf(1), *lsvg*(1)

NAME

model - display VS hardware model

SYNOPSIS

model

DESCRIPTION

Model displays the VS hardware model.

EXAMPLE

```
VS66:63> model
VS21
VS66:63>
```

NAME

more - control more... paging

SYNOPSIS

more [*on* | *off*]

DESCRIPTION

More controls the the "more..." paging in the VS. By default, paging is enabled and assumes a 24 line display. When run without any arguments, *more* displays the current paging state. When provided with a *on* or *off* parameter, paging is enabled or disabled, respectively.

EXAMPLE

```
VS66:63> more
more on
VS66:63> more off
VS66:63> more
more off
VS66:63>
VS66:63>
```

DIAGNOSTICS

The state of more... paging is lost on reboot.

NAME

mvlun - rename LUN id

SYNOPSIS

mvlun *lun newlun*

DESCRIPTION

Mvlun renames a LUN id *lun* to the new LUN id *newlun*. When merging VS installations into a single installation it may be necessary to rename LUN ids before merging to guarantee each LUN is unique.

EXAMPLE

```
VS65:63> lslun
LUN          LENGTH  ONLINE  TARGET  LABEL
0            1.002GB   ON      63.0
1            5.003GB   ON      63.1
VS65:63> mvlun 1 5
updating 61.1
VS65:63> lslun
LUN          LENGTH  ONLINE  TARGET  LABEL
0            1.002GB   ON      63.0
5            5.003GB   ON      63.5
VS65:63>
```

NAME

offline, online – change status of a LUN

SYNOPSIS

offline *lun*

online *lun*

DESCRIPTION

Offline makes the specified *lun* unavailable for use.

Online makes the specified *lun* available for use.

EXAMPLE

```
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            5.003GB  OFF         63.0
VS66:63> online 0
2008.03.23 10:46:30 Putting LUN 0 online
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            5.003GB  ON          63.0
VS66:63> offline 0
2008.03.23 10:46:34 Taking LUN 0 offline
VS66:63> lslun
LUN          LENGTH  ONLINE      TARGET  LABEL
0            5.003GB  OFF         63.0
VS66:63>
```

NAME

passwd - change the password

SYNOPSIS

passwd

DESCRIPTION

The *passwd* command is used to set the VS login password.

If necessary, *passwd* first prompts for the current password. Upon a correct match of the current password, it prompts for the new password twice. Upon a match in the new password, the VS password is successfully changed.

EXAMPLE

```
VS66:63> passwd
  new password: ****
again to verify: ****
password changed
VS66:63> exit
Login[66:63]: ****
VS66:63> passwd
old password: ****
  new password:
again to verify:
password changed
VS66:63> exit
Console password unset.  Access granted.
VS66:63>
```

DIAGNOSTICS

The password is limited to 32 characters. Setting an empty password clears the password.

If you forget your password, enter the password *ivelostit*, and email support@coraid.com the challenge output. The support team will supply the response to the challenge. Entering a successful response to the password reset challenge clears the password. The password *ivelostit* is reserved and may not be used.

NAME

pause – pause background work in progress

SYNOPSIS

pause *workid*

DESCRIPTION

Pause temporarily stops background work in progress. The background work can be restarted using the *resume* command. The status of all background work is visible via the *wstat* command.

EXAMPLE

```
VS66:63> mirror 61.0 61.1
updating 61.0
2008.03.22 14:02:35 started rebuilding: 61.0 -> 61.1
VS66:63> wstat
ID          ACTION      DONE      RATE      TTC      STATE      NP
0          mirror 61.0->61.1  0.01%    350.47MB/s  7:34:10      R      64
VS66:63> pause 0
2008.03.22 14:02:42 work id 0 paused
VS66:63> wstat
ID          ACTION      DONE      RATE      TTC      STATE      NP
0          mirror 61.0->61.1  0.02%    265.39MB/s  9:59:46      R (P)    0
VS66:63>
```

DIAGNOSTICS

Pause simply directs the background work to pause itself when it can. As an example, mirror copying will pause only when all currently processing extents are completed. To ensure work is fully paused, verify the *wstat* output displays the work in the (P) state.

SEE ALSO

kill(1), *mirror(1)*, *resume(1)*, *wstat(1)*

NAME

ps - show processes

SYNOPSIS

ps [-m]

DESCRIPTION

Ps displays information about the processes running in the system. Without any arguments, the output is formatted as follows:

TIME	STATE	NAME
------	-------	------

TIME is the amount of run time that process has had since the last reboot.

STATE is the current state of the process, which can be *ready*, *running*, or *sleeping*.

NAME is the name of the process.

If the optional *-m* argument is used, the output is formatted as follows:

NAME	WCHAN	CALLERPC	AVAIL/STACK	STATE
------	-------	----------	-------------	-------

NAME is the name of the process.

WCHAN is information about an internal communication channel.

CALLERPC is information about the internal program counter.

AVAIL/STACK is information about the stack size being used.

STATE is the current state of the process, which can be *ready*, *running*, or *sleeping*.

EXAMPLE

```
VS66:63> ps
TIME      STATE      NAME
0:03.240  sleeping   main
0:00.000  sleeping   rxe563:
0:00.000  sleeping   rxe563:
0:00.000  sleeping   sysloga
0:00.030  sleeping   cectime
0:00.000  sleeping   cecout
1:57.420  sleeping   worker
1:55.960  sleeping   worker
2:00.290  sleeping   worker
2:00.430  sleeping   worker
1:58.040  sleeping   worker
1:58.690  sleeping   worker
1:59.500  sleeping   worker
1:59.620  sleeping   worker
2:00.850  sleeping   worker
1:59.200  sleeping   worker
1:59.740  sleeping   worker
2:01.300  sleeping   worker
1:57.520  sleeping   worker
1:57.850  sleeping   worker
1:59.510  sleeping   worker
```

more...

VS66:63>

EXAMPLE CURTAILED FOR BREVITY

NAME

`psstat` - display power supply status

SYNOPSIS

`psstat`

DESCRIPTION

Psstat displays the status of the power supplies. The reported status is one of the following:

PS0 is down

PS1 is down

PS0 and PS1 are up

For the VS21, PS0 is the left power supply and PS1 is the right power supply when viewed from the rear.

DIAGNOSTICS

Psstat relies on a Coraid modification to the power supply backplane to fetch power supply status. If this modification is not done or is inoperable, *psstat* will report an unknown status and request the user contact Coraid support for assistance.

The VS has a background process that checks the power supplies every minute and reports via a syslog message when either power supply goes down or returns.

NAME

release - show software release date

SYNOPSIS

release

DESCRIPTION

Release displays the release date of the currently running version of firmware.

EXAMPLE

```
VS66:63> release
Running release 20081028 - Tue Oct 28 15:16:47 EDT 2008
VS66:63>
```

NAME

restore – restore a PV to its volume group

SYNOPSIS

restore *pv*

DESCRIPTION

When PV failures occur and a volume group is updated, every PV in the group is updated to a new generation number. Any PVs that are not available to have their generation numbers updated become "out of date" with respect to the other PVs in their volume group. When out of date PVs are brought back online, they are not inserted into the volume group in order to maintain a consistent volume group state.

The *restore* command can be used to bring these out of date PVs back into the volume group. The PV is brought back in by loading its extent descriptors into the pool and creating any LUNs on the PV. This is a potentially dangerous action and should only be performed in certain situations. In general, if the LUN configuration did not change since the PV went offline, it is safe to use restore.

The primary concern in restoring a PV to a volume group is extent mapping duplication. If LUNs were removed and recreated, it is possible for extent numbers for a LUN to be duplicated on the old PV. Restoring a PV in this scenario can lead to data corruption. Additionally, *restore* should not be used on an out of date pv that was in a mirrored configuration if the mirrored pair is currently in use. Restoring a PV in this case will cause the PV to be brought in as an independent PV and extent duplication will occur. To remirror out of date PVs, simply use the mirror command.

NAME

resume - continue paused work

SYNOPSIS

resume *workid*

DESCRIPTION

Resume restarts background work that has previously been paused using the *pause* command. The status of all background work is displayed with the *wstat* command.

EXAMPLE

```
VS66:63> wstat
ID          ACTION    DONE      RATE      TTC      STATE    NP
0          mirror 61.0->61.1  0.03%    25.54MB/s 105:56:57    R    64
VS66:63> pause 0
2008.03.22 14:52:53 work id 0 paused
VS66:63> wstat
ID          ACTION    DONE      RATE      TTC      STATE    NP
0          mirror 61.0->61.1  0.04%    68.50MB/s  38:56:53    R (P)    0
VS66:63> resume 0
2008.03.22 14:52:58 work id 0 resumed
VS66:63> wstat
ID          ACTION    DONE      RATE      TTC      STATE    NP
0          mirror 61.0->61.1  0.05%    81.60MB/s  32:41:43    R    64
VS66:63>
```

SEE ALSO

kill(1), mirror(1), pause(1), wstat(1)

NAME

rmlun - remove logical volume

SYNOPSIS

rmlun lun

DESCRIPTION

Rmlun removes the logical volume specified by *lun*. A logical volume must be offline to be removed.

A confirmation prompt is displayed to confirm the requested action. Once a logical volume is removed, it cannot be rebuilt. All data associated with it is lost.

SEE ALSO

offline(1), *lslun(1)*

NAME

rmpv - remove physical volume

SYNOPSIS

rmpv pv

DESCRIPTION

Rmpv removes the physical volume *pv* from its volume group. A physical volume cannot be removed if it is used by any LUN.

SEE ALSO

lspv(1)

NAME

`rmvg` - remove volume group

SYNOPSIS

`rmvg vg`

DESCRIPTION

Rmvg removes the volume group specified by *vg* from the system. The volume group must not contain any physical volumes.

SEE ALSO

lsvg(1)

NAME

setboot - set default boot image

SYNOPSIS

setboot [*A* | *B*]

DESCRIPTION

Setboot sets the default boot image to one of *A* or *B*. If no argument is given, the current default boot image is displayed.

EXAMPLE

```
VS66:63> setboot
will boot from B
VS66:63> setboot A
VS66:63> setboot
will boot from A
VS66:63>
```

DIAGNOSTICS

It is not currently possible to display the release of images *A* and *B* from the command line.

NAME

sethash, showhash, verifyhash – set, view, or verify flash signature

SYNOPSIS

```
sethash sig_s sig_r
```

```
showhash
```

```
verifyhash
```

DESCRIPTION

Every VS has a unique signature that validates it as a Coraid product. The VS must have a valid signature to pass AoE traffic. As each signature is unique to the VS on which it is installed, a signature **cannot** be copied from one unit to another.

Sethash sets the hash signature in the VS flash. This command should only be run at the direction of Coraid support staff as it **can destroy a valid signature**.

Showhash shows the hash key input used to generate the hash signature.

Verifyhash verifies the hash signature stored in flash.

EXAMPLE

```
VS66:63> verifyhash  
flash key is good  
VS66:63>
```

NAME

shelf, *srvshelf* – display or set administrative and AoE serve shelf addresses

SYNOPSIS

shelf [*shelf*]

srvshelf [*shelf*]

DESCRIPTION

Shelf displays the current admin shelf number. When provided with a *shelf* argument the admin shelf number is set to the specified *shelf*. The admin shelf address is used for management of the VS. It is used for flash update LUNs and CEC communication. Additionally, it is used for cluster election arbitration.

Srvshelf displays the current AoE serve shelf number. When provided with a *shelf* argument the serve shelf number is set to the specified *shelf*. The AoE serve shelf address is used to export AoE (SR) backed storage LUNs.

The admin and serve shelf addresses may be set to any value between 0 and 65534.

DIAGNOSTICS

The *shelf* and *srvshelf* addresses affect cluster state and should be changed carefully in a running cluster configuration. In a cluster, the *srvshelf* address denotes a cluster group. The admin shelf address distinguishes a VS from other nodes in the cluster group and determines whether a node is a master or slave. The VS with the highest admin shelf address for a given *srvshelf* is the master; all other VSs with the same *srvshelf* address are slaves.

If the admin shelf address is changed on either a master or a slave, a reelection is initiated. If the *srvshelf* address is changed on the slave, the slave loses the ability to use all storage it used as a part of the old *srvshelf* address. If the *srvshelf* address is changed on the master, the storage is relabeled to the new *srvshelf* address, effectively taking the storage away from any slaves existing under the old *srvshelf* address.

SEE ALSO

clrselves(1)

NAME

sos - output information for support

SYNOPSIS

sos

DESCRIPTION

sos displays information needed by Coraid support for supporting the VS appliance. Please capture the output in its entirety and send it along with any questions when contacting support about VS issues.

NAME

syslog - manage syslog configuration

SYNOPSIS

syslog [-cp] [*message*]

DESCRIPTION

The *syslog* command is used to configure and display the syslog configuration, as well as send test messages to the configured syslog server. The `-c` flag enters a configuration dialogue to specify the source and destination IP addresses to place in the syslog messages, as well as the local port to use in sending the message. IP addresses are specified in dotted quad, IPv4 notation. The `-p` flag displays the stored syslog configuration.

Without flags, a *message* may be specified to send in a syslog message to the configured syslog server in order to test the accuracy of the configuration. The message may contain spaces and needn't be quoted; syslog will catenate all arguments into a single syslog message appropriately.

EXAMPLE

```
VS66:63> syslog -p
destination IP: 192.168.0.100
source IP: 192.168.0.61
local port: 0
VS66:63> syslog -c
Configuring syslog. Enter IP addresses in dotted notation.
Local port is of the form [0-9]
IPv4 destination address [192.168.0.100]: 192.168.0.101
IPv4 source address [192.168.0.61]: 192.168.0.5
Local syslog interface [0]:
Configuration successful.
VS66:63> syslog -p
destination IP: 192.168.0.101
source IP: 192.168.0.5
local port: 0
VS66:63> syslog test message
2008.03.24 13:26:27 test message
VS66:63>
```

DIAGNOSTICS

The syslogging mechanism in the SR is tied into the timestamp logging that is printed to the console. Any message printed to the console with a timestamp has also been sent to syslog.

Each VS must have its syslog configuration set independently. Users with cluster installations should take care to ensure syslog is enabled on all cluster nodes.

TEMP(1)

TEMP(1)

NAME

temp - display CPU temperature

SYNOPSIS

temp

DESCRIPTION

Temp displays the temperature of the CPU in degrees celsius.

NAME

uptime - display system uptime

SYNOPSIS

uptime

DESCRIPTION

Uptime show the current system uptime.

EXAMPLE

```
VS63:63> uptime
up 0 days, 00:21:41
VS63:63>
```

NAME

wstat - list background work

SYNOPSIS

wstat

DESCRIPTION

The VS performs background work on the user's behalf for LUN copying and physical volume mirroring. *Wstat* displays the status of the background work in process. The output is formatted as follows:

ID - the numerical id of the background work entity

ACTION - a synopsis of the work being performed

DONE - the percent of total work completed

RATE - the total read+write rate of I/O being performed in megabytes per second

TTC - the estimated time to completion based upon the current I/O rate

STATE - the state of the work entity

The possible values for this field are:

R - Rebuilding a mirror

C - Copying a LUN

(p) - Pausing work entity

(P) - Paused work entity

(k) - Killing work entity

(b) - Bailing work entity

NP - the number of worker processes currently performing the work

EXAMPLE

```
VS66:63> wstat
ID          ACTION    DONE      RATE          TTC    STATE    NP
0          mirror 61.0->61.1  2.36%  350.71MB/s   7:23:28    R    64
VS66:63>
```

DIAGNOSTICS

The lower case ([kpb]) states are temporary states that denote what will be done when all workers finish the extents they are currently processing. The (k) and (p) states are triggered by the *kill* and *pause* commands respectively. The (b) state occurs when an election forces a master to bail on work in progress.

SEE ALSO

kill(1), *pause(1)*, *resume(1)*

NAME

Appendix A - Updating VS Firmware

DESCRIPTION

The VS firmware update procedure is very similar to the firmware update procedure for the SR. A special update LUN is created that exports the flash boot image to be updated. From the initiator, the new firmware release is written to the update LUN using coraid-update or dd. Unlike the SR, there is no "update" command to run on the VS after writing the new firmware image to the LUN from the client; a simple reboot is all that's necessary to run the updated firmware.

EXAMPLE

```
VS66:63> release
Running Release: Tue Jul 29 14:45:40 EDT 2008
VS66:63> setboot
will boot from A
VS66:63> mklun 15 flash A
VS66:63> online 15
2008.07.30 12:30:50 putting lun 15 online
VS66:63> lslun 15
LUN          LENGTH  ONLINE      TARGET  LABEL
15           0.008GB  ON          66.15   flash area A
VS66:63>
```

On the client system:

```
% aoe-stat | grep 66
   e66.15      0.008GB      eth0 8192  up
% # Verify the md5sum with that reported by the VS support page
% md5sum VS20080731
945be5f330fdb7944c10673842a8bc5b  VS20080731
% coraid-update VS20080731 /dev/etherd/e66.15
% # Or alternately, if your system lacks coraid-update
% dd if=VS20080731 of=/dev/etherd/e66.15
291+1 records in
291+1 records out
149060 bytes (149 kB) copied, 0.219686 s, 679 kB/s
%
```

Back on the VS appliance:

```
VS66:63> reboot

rebooting...

Coraid Boot ROM

Boot A?
booting...
Done
2,145,779,712 bytes of memory

VS: CORAID EtherDrive Storage Virtualization Appliance

VS Release: Thu Jul 31 12:38:28 EDT 2008
BIOS build 11/02/07
2,145,779,712 bytes
Flash size 32 MB
maxpe=68,572,963
Console password unset.  Access granted.
VS66:63>
```


The fonts used in this manual were Lucida Sans, in a special version incorporating over 1700 characters from the Unicode Standard, along with Lucida Sans Italic, Lucida Sans DemiBold, and Lucida Typewriter, designed by Bigelow & Holmes, Atherton, California. The hinted Adobe Type 1 representation of the fonts was provided by Y&Y Inc., 45 Walden Street, Concord, MA, 01742, USA.