

```

*****
50491 Fri Mar 25 18:28:30 2016
new/usr/src/man/man1m/zpool.1m
6781 zpool man page needs updated to remove duplicate entry of "cannot be" where
Reviewed by: Toomas Soome <tsoome@me.com>
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
1 .\"
2 .\" CDDL HEADER START
3 .\"
4 .\" The contents of this file are subject to the terms of the
5 .\" Common Development and Distribution License (the "License").
6 .\" You may not use this file except in compliance with the License.
7 .\"
8 .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 .\" or http://www.opensolaris.org/os/licensing.
10 .\" See the License for the specific language governing permissions
11 .\" and limitations under the License.
12 .\"
13 .\" When distributing Covered Code, include this CDDL HEADER in each
14 .\" file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 .\" If applicable, add the following below this CDDL HEADER, with the
16 .\" fields enclosed by brackets "[]" replaced with your own identifying
17 .\" information: Portions Copyright [yyyy] [name of copyright owner]
18 .\"
19 .\" CDDL HEADER END
20 .\"
21 .\"
22 .\" Copyright (c) 2007, Sun Microsystems, Inc. All Rights Reserved.
23 .\" Copyright (c) 2013 by Delphix. All rights reserved.
24 .\" Copyright 2016 Nexenta Systems, Inc.
25 .\"
26 .Dd March 25, 2016
26 .Dd February 15, 2016
27 .Dt ZPOOL 1M
28 .Os
29 .Sh NAME
30 .Nm zpool
31 .Nd configure ZFS storage pools
32 .Sh SYNOPSIS
33 .Nm
34 .Fl \?
35 .Nm
36 .Cm add
37 .Op Fl fn
38 .Ar pool vdev Ns ...
39 .Nm
40 .Cm attach
41 .Op Fl f
42 .Ar pool device new_device
43 .Nm
44 .Cm clear
45 .Ar pool
46 .Op Ar device
47 .Nm
48 .Cm create
49 .Op Fl dfn
50 .Op Fl m Ar mountpoint
51 .Oo Fl o Ar property Ns = Ns Ar value Oc Ns ...
52 .Oo Fl O Ar file-system-property Ns = Ns Ar value Oc Ns ...
53 .Op Fl R Ar root
54 .Ar pool vdev Ns ...
55 .Nm
56 .Cm destroy
57 .Op Fl f
58 .Ar pool

```

```

59 .Nm
60 .Cm detach
61 .Ar pool device
62 .Nm
63 .Cm export
64 .Op Fl f
65 .Ar pool Ns ...
66 .Nm
67 .Cm get
68 .Op Fl Hp
69 .Op Fl o Ar field Ns Oo , Ns Ar field Oc Ns ...
70 .Sy all Ns | Ns Ar property Ns Oo , Ns Ar property Oc Ns ...
71 .Ar pool Ns ...
72 .Nm
73 .Cm history
74 .Op Fl il
75 .Oo Ar pool Oc Ns ...
76 .Nm
77 .Cm import
78 .Op Fl D
79 .Op Fl d Ar dir
80 .Nm
81 .Cm import
82 .Fl a
83 .Op Fl DfmN
84 .Op Fl F Op Fl n
85 .Op Fl c Ar cachefile Ns | Ns Fl d Ar dir
86 .Op Fl o Ar mntopts
87 .Oo Fl o Ar property Ns = Ns Ar value Oc Ns ...
88 .Op Fl R Ar root
89 .Nm
90 .Cm import
91 .Op Fl Dfm
92 .Op Fl F Op Fl n
93 .Op Fl c Ar cachefile Ns | Ns Fl d Ar dir
94 .Op Fl o Ar mntopts
95 .Oo Fl o Ar property Ns = Ns Ar value Oc Ns ...
96 .Op Fl R Ar root
97 .Ar pool Ns | Ns Ar id
98 .Op Ar newpool
99 .Nm
100 .Cm iostat
101 .Op Fl v
102 .Op Fl T Sy u Ns | Ns Sy d
103 .Oo Ar pool Oc Ns ...
104 .Op Ar interval Op Ar count
105 .Nm
106 .Cm list
107 .Op Fl Hpv
108 .Op Fl o Ar property Ns Oo , Ns Ar property Oc Ns ...
109 .Op Fl T Sy u Ns | Ns Sy d
110 .Oo Ar pool Oc Ns ...
111 .Op Ar interval Op Ar count
112 .Nm
113 .Cm offline
114 .Op Fl t
115 .Ar pool Ar device Ns ...
116 .Nm
117 .Cm online
118 .Op Fl e
119 .Ar pool Ar device Ns ...
120 .Nm
121 .Cm reguid
122 .Ar pool
123 .Nm
124 .Cm reopen

```

125 .Ar pool  
 126 .Nm  
 127 .Cm remove  
 128 .Ar pool Ar device Ns ...  
 129 .Nm  
 130 .Cm replace  
 131 .Op Fl f  
 132 .Ar pool Ar device Op Ar new\_device  
 133 .Nm  
 134 .Cm scrub  
 135 .Op Fl s  
 136 .Ar pool Ns ...  
 137 .Nm  
 138 .Cm set  
 139 .Ar property Ns = Ns Ar value  
 140 .Ar pool  
 141 .Nm  
 142 .Cm split  
 143 .Op Fl n  
 144 .Oo Fl o Ar property Ns = Ns Ar value Oc Ns ...  
 145 .Op Fl R Ar root  
 146 .Ar pool newpool  
 147 .Nm  
 148 .Cm status  
 149 .Op Fl Dvx  
 150 .Op Fl T Sy u Ns | Ns Sy d  
 151 .Oo Ar pool Oc Ns ...  
 152 .Op Ar interval Op Ar count  
 153 .Nm  
 154 .Cm upgrade  
 155 .Nm  
 156 .Cm upgrade  
 157 .Fl v  
 158 .Nm  
 159 .Cm upgrade  
 160 .Op Fl V Ar version  
 161 .Fl a Ns | Ns Ar pool Ns ...  
 162 .Sh DESCRIPTION  
 163 The  
 164 .Nm  
 165 command configures ZFS storage pools. A storage pool is a collection of devices  
 166 that provides physical storage and data replication for ZFS datasets. All  
 167 datasets within a storage pool share the same space. See  
 168 .Xr zfs 1M  
 169 for information on managing datasets.  
 170 .Ss Virtual Devices (vdevs)  
 171 A "virtual device" describes a single device or a collection of devices  
 172 organized according to certain performance and fault characteristics. The  
 173 following virtual devices are supported:  
 174 .Bl -tag -width Ds  
 175 .It Sy disk  
 176 A block device, typically located under  
 177 .Pa /dev/dsk .  
 178 ZFS can use individual slices or partitions, though the recommended mode of  
 179 operation is to use whole disks. A disk can be specified by a full path, or it  
 180 can be a shorthand name  
 181 .Po the relative portion of the path under  
 182 .Pa /dev/dsk  
 183 .Pc .  
 184 A whole disk can be specified by omitting the slice or partition designation.  
 185 For example,  
 186 .Pa c0t0d0  
 187 is equivalent to  
 188 .Pa /dev/dsk/c0t0d0s2 .  
 189 When given a whole disk, ZFS automatically labels the disk, if necessary.  
 190 .It Sy file

191 A regular file. The use of files as a backing store is strongly discouraged. It  
 192 is designed primarily for experimental purposes, as the fault tolerance of a  
 193 file is only as good as the file system of which it is a part. A file must be  
 194 specified by a full path.  
 195 .It Sy mirror  
 196 A mirror of two or more devices. Data is replicated in an identical fashion  
 197 across all components of a mirror. A mirror with N disks of size X can hold X  
 198 bytes and can withstand (N-1) devices failing before data integrity is  
 199 compromised.  
 200 .It Sy raidz , raidz1 , raidz2 , raidz3  
 201 A variation on RAID-5 that allows for better distribution of parity and  
 202 eliminates the RAID-5  
 203 .Qq write hole  
 204 .Pq in which data and parity become inconsistent after a power loss .  
 205 Data and parity is striped across all disks within a raidz group.  
 206 .Pp  
 207 A raidz group can have single-, double-, or triple-parity, meaning that the  
 208 raidz group can sustain one, two, or three failures, respectively, without  
 209 losing any data. The  
 210 .Sy raidz1  
 211 vdev type specifies a single-parity raidz group; the  
 212 .Sy raidz2  
 213 vdev type specifies a double-parity raidz group; and the  
 214 .Sy raidz3  
 215 vdev type specifies a triple-parity raidz group. The  
 216 .Sy raidz  
 217 vdev type is an alias for  
 218 .Sy raidz1 .  
 219 .Pp  
 220 A raidz group with N disks of size X with P parity disks can hold approximately  
 221 (N-P)\*X bytes and can withstand P device(s) failing before data integrity is  
 222 compromised. The minimum number of devices in a raidz group is one more than  
 223 the number of parity disks. The recommended number is between 3 and 9 to help  
 224 increase performance.  
 225 .It Sy spare  
 226 A special pseudo-vdev which keeps track of available hot spares for a pool. For  
 227 more information, see the  
 228 .Sx Hot Spares  
 229 section.  
 230 .It Sy log  
 231 A separate intent log device. If more than one log device is specified, then  
 232 writes are load-balanced between devices. Log devices can be mirrored. However,  
 233 raidz vdev types are not supported for the intent log. For more information,  
 234 see the  
 235 .Sx Intent Log  
 236 section.  
 237 .It Sy cache  
 238 **A device used to cache storage pool data. A cache device cannot be configured**  
 239 **as a mirror or raidz group. For more information, see the**  
 238 **A device used to cache storage pool data. A cache device cannot be**  
 239 **configured as a mirror or raidz group. For more information, see the**  
 240 .Sx Cache Devices  
 241 section.  
 242 .El  
 243 .Pp  
 244 Virtual devices cannot be nested, so a mirror or raidz virtual device can only  
 245 contain files or disks. Mirrors of mirrors  
 246 .Pq or other combinations  
 247 are not allowed.  
 248 .Pp  
 249 A pool can have any number of virtual devices at the top of the configuration  
 250 .Po known as  
 251 .Qq root vdevs  
 252 .Pc .  
 253 Data is dynamically distributed across all top-level devices to balance data  
 254 among devices. As new virtual devices are added, ZFS automatically places data

```

255 on the newly available devices.
256 .Pp
257 Virtual devices are specified one at a time on the command line, separated by
258 whitespace. The keywords
259 .Sy mirror
260 and
261 .Sy raidz
262 are used to distinguish where a group ends and another begins. For example,
263 the following creates two root vdevs, each a mirror of two disks:
264 .Bd -literal
265 # zpool create mypool mirror c0t0d0 c0t1d0 mirror clt0d0 clt1d0
266 .Ed
267 .Ss Device Failure and Recovery
268 ZFS supports a rich set of mechanisms for handling device failure and data
269 corruption. All metadata and data is checksummed, and ZFS automatically repairs
270 bad data from a good copy when corruption is detected.
271 .Pp
272 In order to take advantage of these features, a pool must make use of some form
273 of redundancy, using either mirrored or raidz groups. While ZFS supports
274 running in a non-redundant configuration, where each root vdev is simply a disk
275 or file, this is strongly discouraged. A single case of bit corruption can
276 render some or all of your data unavailable.
277 .Pp
278 A pool's health status is described by one of three states: online, degraded,
279 or faulted. An online pool has all devices operating normally. A degraded pool
280 is one in which one or more devices have failed, but the data is still
281 available due to a redundant configuration. A faulted pool has corrupted
282 metadata, or one or more faulted devices, and insufficient replicas to continue
283 functioning.
284 .Pp
285 The health of the top-level vdev, such as mirror or raidz device, is
286 potentially impacted by the state of its associated vdevs, or component
287 devices. A top-level vdev or component device is in one of the following
288 states:
289 .Bl -tag -width "DEGRADED"
290 .It Sy DEGRADED
291 One or more top-level vdevs is in the degraded state because one or more
292 component devices are offline. Sufficient replicas exist to continue
293 functioning.
294 .Pp
295 One or more component devices is in the degraded or faulted state, but
296 sufficient replicas exist to continue functioning. The underlying conditions
297 are as follows:
298 .Bl -bullet
299 .It
300 The number of checksum errors exceeds acceptable levels and the device is
301 degraded as an indication that something may be wrong. ZFS continues to use the
302 device as necessary.
303 .It
304 The number of I/O errors exceeds acceptable levels. The device could not be
305 marked as faulted because there are insufficient replicas to continue
306 functioning.
307 .El
308 .It Sy FAULTED
309 One or more top-level vdevs is in the faulted state because one or more
310 component devices are offline. Insufficient replicas exist to continue
311 functioning.
312 .Pp
313 One or more component devices is in the faulted state, and insufficient
314 replicas exist to continue functioning. The underlying conditions are as
315 follows:
316 .Bl -bullet
317 .It
318 The device could be opened, but the contents did not match expected values.
319 .It
320 The number of I/O errors exceeds acceptable levels and the device is faulted to

```

```

321 prevent further use of the device.
322 .El
323 .It Sy OFFLINE
324 The device was explicitly taken offline by the
325 .Nm zpool Cm offline
326 command.
327 .It Sy ONLINE
328 The device is online and functioning.
329 .It Sy REMOVED
330 The device was physically removed while the system was running. Device removal
331 detection is hardware-dependent and may not be supported on all platforms.
332 .It Sy UNAVAIL
333 The device could not be opened. If a pool is imported when a device was
334 unavailable, then the device will be identified by a unique identifier instead
335 of its path since the path was never correct in the first place.
336 .El
337 .Pp
338 If a device is removed and later re-attached to the system, ZFS attempts
339 to put the device online automatically. Device attach detection is
340 hardware-dependent and might not be supported on all platforms.
341 .Ss Hot Spares
342 ZFS allows devices to be associated with pools as
343 .Qq hot spares .
344 These devices are not actively used in the pool, but when an active device
345 fails, it is automatically replaced by a hot spare. To create a pool with hot
346 spares, specify a
347 .Sy spare
348 vdev with any number of devices. For example,
349 .Bd -literal
350 # zpool create pool mirror c0d0 c1d0 spare c2d0 c3d0
351 .Ed
352 .Pp
353 Spares can be shared across multiple pools, and can be added with the
354 .Nm zpool Cm add
355 command and removed with the
356 .Nm zpool Cm remove
357 command. Once a spare replacement is initiated, a new
358 .Sy spare
359 vdev is created within the configuration that will remain there until the
360 original device is replaced. At this point, the hot spare becomes available
361 again if another device fails.
362 .Pp
363 If a pool has a shared spare that is currently being used, the pool can not be
364 exported since other pools may use this shared spare, which may lead to
365 potential data corruption.
366 .Pp
367 An in-progress spare replacement can be cancelled by detaching the hot spare.
368 If the original faulted device is detached, then the hot spare assumes its
369 place in the configuration, and is removed from the spare list of all active
370 pools.
371 .Pp
372 Spares cannot replace log devices.
373 .Ss Intent Log
374 The ZFS Intent Log (ZIL) satisfies POSIX requirements for synchronous
375 transactions. For instance, databases often require their transactions to be on
376 stable storage devices when returning from a system call. NFS and other
377 applications can also use
378 .Xr fsync 3C
379 to ensure data stability. By default, the intent log is allocated from blocks
380 within the main pool. However, it might be possible to get better performance
381 using separate intent log devices such as NVRAM or a dedicated disk. For
382 example:
383 .Bd -literal
384 # zpool create pool c0d0 c1d0 log c2d0
385 .Ed
386 .Pp

```

387 Multiple log devices can also be specified, and they can be mirrored. See the  
 388 .Sx EXAMPLES  
 389 section for an example of mirroring multiple log devices.  
 390 .Pp  
 391 Log devices can be added, replaced, attached, detached, and imported and  
 392 exported as part of the larger pool. Mirrored log devices can be removed by  
 393 specifying the top-level mirror for the log.  
 394 .Ss Cache Devices  
 395 Devices can be added to a storage pool as  
 396 .Qg cache devices .  
 397 These devices provide an additional layer of caching between main memory and  
 398 disk. For read-heavy workloads, where the working set size is much larger than  
 399 what can be cached in main memory, using cache devices allow much more of this  
 400 working set to be served from low latency media. Using cache devices provides  
 401 the greatest performance improvement for random read-workloads of mostly static  
 402 content.  
 403 .Pp  
 404 To create a pool with cache devices, specify a  
 405 .Sy cache  
 406 vdev with any number of devices. For example:  
 407 .Bd -literal  
 408 # zpool create pool c0d0 c1d0 cache c2d0 c3d0  
 409 .Ed  
 410 .Pp  
 411 Cache devices cannot be mirrored or part of a raidz configuration. If a read  
 412 error is encountered on a cache device, that read I/O is reissued to the  
 413 original storage pool device, which might be part of a mirrored or raidz  
 414 configuration.  
 415 .Pp  
 416 The content of the cache devices is considered volatile, as is the case with  
 417 other system caches.  
 418 .Ss Properties  
 419 Each pool has several properties associated with it. Some properties are  
 420 read-only statistics while others are configurable and change the behavior of  
 421 the pool.  
 422 .Pp  
 423 The following are read-only properties:  
 424 .Bl -tag -width Ds  
 425 .It Sy available  
 426 Amount of storage available within the pool. This property can also be referred  
 427 to by its shortened column name,  
 428 .Sy avail .  
 429 .It Sy capacity  
 430 Percentage of pool space used. This property can also be referred to by its  
 431 shortened column name,  
 432 .Sy cap .  
 433 .It Sy expandsize  
 434 Amount of uninitialized space within the pool or device that can be used to  
 435 increase the total capacity of the pool. Uninitialized space consists of  
 436 any space on an EFI labeled vdev which has not been brought online  
 437 .Po e.g, using  
 438 .Nm zpool Cm online Fl e  
 439 .Pc .  
 440 This space occurs when a LUN is dynamically expanded.  
 441 .It Sy fragmentation  
 442 The amount of fragmentation in the pool.  
 443 .It Sy free  
 444 The amount of free space available in the pool.  
 445 .It Sy freeing  
 446 After a file system or snapshot is destroyed, the space it was using is  
 447 returned to the pool asynchronously.  
 448 .Sy freeing  
 449 is the amount of space remaining to be reclaimed. Over time  
 450 .Sy freeing  
 451 will decrease while  
 452 .Sy free

453 increases.  
 454 .It Sy health  
 455 The current health of the pool. Health can be one of  
 456 .Sy ONLINE , DEGRADED , FAULTED , OFFLINE , REMOVED , UNAVAIL .  
 457 .It Sy guid  
 458 A unique identifier for the pool.  
 459 .It Sy size  
 460 Total size of the storage pool.  
 461 .It Sy unsupported@ Ns Em feature\_guid  
 462 Information about unsupported features that are enabled on the pool. See  
 463 .Xr zpool-features 5  
 464 for details.  
 465 .It Sy used  
 466 Amount of storage space used within the pool.  
 467 .El  
 468 .Pp  
 469 The space usage properties report actual physical space available to the  
 470 storage pool. The physical space can be different from the total amount of  
 471 space that any contained datasets can actually use. The amount of space used in  
 472 a raidz configuration depends on the characteristics of the data being  
 473 written. In addition, ZFS reserves some space for internal accounting  
 474 that the  
 475 .Xr zfs 1M  
 476 command takes into account, but the  
 477 .Nm  
 478 command does not. For non-full pools of a reasonable size, these effects should  
 479 be invisible. For small pools, or pools that are close to being completely  
 480 full, these discrepancies may become more noticeable.  
 481 .Pp  
 482 The following property can be set at creation time and import time:  
 483 .Bl -tag -width Ds  
 484 .It Sy altroot  
 485 Alternate root directory. If set, this directory is prepended to any mount  
 486 points within the pool. This can be used when examining an unknown pool where  
 487 the mount points cannot be trusted, or in an alternate boot environment, where  
 488 the typical paths are not valid.  
 489 .Sy altroot  
 490 is not a persistent property. It is valid only while the system is up. Setting  
 491 .Sy altroot  
 492 defaults to using  
 493 .Sy cachefile Ns = Ns Sy none ,  
 494 though this may be overridden using an explicit setting.  
 495 .El  
 496 .Pp  
 497 The following property can be set only at import time:  
 498 .Bl -tag -width Ds  
 499 .It Sy readonly Ns = Ns Sy on Ns | Ns Sy off  
 500 If set to  
 501 .Sy on ,  
 502 the pool will be imported in read-only mode. This property can also be referred  
 503 to by its shortened column name,  
 504 .Sy ronly .  
 505 .El  
 506 .Pp  
 507 The following properties can be set at creation time and import time, and later  
 508 changed with the  
 509 .Nm zpool Cm set  
 510 command:  
 511 .Bl -tag -width Ds  
 512 .It Sy autoexpand Ns = Ns Sy on Ns | Ns Sy off  
 513 Controls automatic pool expansion when the underlying LUN is grown. If set to  
 514 .Sy on ,  
 515 the pool will be resized according to the size of the expanded device. If the  
 516 device is part of a mirror or raidz then all devices within that mirror/raidz  
 517 group must be expanded before the new space is made available to the pool. The  
 518 default behavior is

519 .Sy off .  
 520 This property can also be referred to by its shortened column name,  
 521 .Sy expand .  
 522 .It Sy autoreplace Ns = Ns Sy on Ns | Ns Sy off  
 523 Controls automatic device replacement. If set to  
 524 .Sy off ,  
 525 device replacement must be initiated by the administrator by using the  
 526 .Nm zpool Cm replace  
 527 command. If set to  
 528 .Sy on ,  
 529 any new device, found in the same physical location as a device that previously  
 530 belonged to the pool, is automatically formatted and replaced. The default  
 531 behavior is  
 532 .Sy off .  
 533 This property can also be referred to by its shortened column name,  
 534 .Sy replace .  
 535 .It Sy bootfs Ns = Ns Ar pool Ns / Ns Ar dataset  
 536 Identifies the default bootable dataset for the root pool. This property is  
 537 expected to be set mainly by the installation and upgrade programs.  
 538 .It Sy cachefile Ns = Ns Ar path Ns | Ns Sy none  
 539 Controls the location of where the pool configuration is cached. Discovering  
 540 all pools on system startup requires a cached copy of the configuration data  
 541 that is stored on the root file system. All pools in this cache are  
 542 automatically imported when the system boots. Some environments, such as  
 543 install and clustering, need to cache this information in a different location  
 544 so that pools are not automatically imported. Setting this property caches the  
 545 pool configuration in a different location that can later be imported with  
 546 .Nm zpool Cm import Fl c .  
 547 Setting it to the special value  
 548 .Sy none  
 549 creates a temporary pool that is never cached, and the special value  
 550 .Qq  
 551 .Pg empty string  
 552 uses the default location.  
 553 .Pp  
 554 Multiple pools can share the same cache file. Because the kernel destroys and  
 555 recreates this file when pools are added and removed, care should be taken when  
 556 attempting to access this file. When the last pool using a  
 557 .Sy cachefile  
 558 is exported or destroyed, the file is removed.  
 559 .It Sy comment Ns = Ns Ar text  
 560 A text string consisting of printable ASCII characters that will be stored  
 561 such that it is available even if the pool becomes faulted. An administrator  
 562 can provide additional information about a pool using this property.  
 563 .It Sy dedupditto Ns = Ns Ar number  
 564 Threshold for the number of block ditto copies. If the reference count for a  
 565 deduplicated block increases above this number, a new ditto copy of this block  
 566 is automatically stored. The default setting is  
 567 .Sy 0  
 568 which causes no ditto copies to be created for deduplicated blocks. The minimum  
 569 legal nonzero setting is  
 570 .Sy 100 .  
 571 .It Sy delegation Ns = Ns Sy on Ns | Ns Sy off  
 572 Controls whether a non-privileged user is granted access based on the dataset  
 573 permissions defined on the dataset. See  
 574 .Xr zfs 1M  
 575 for more information on ZFS delegated administration.  
 576 .It Sy failmode Ns = Ns Sy wait Ns | Ns Sy continue Ns | Ns Sy panic  
 577 Controls the system behavior in the event of catastrophic pool failure. This  
 578 condition is typically a result of a loss of connectivity to the underlying  
 579 storage device(s) or a failure of all devices within the pool. The behavior of  
 580 such an event is determined as follows:  
 581 .Bl -tag -width "continue"  
 582 .It Sy wait  
 583 Blocks all I/O access until the device connectivity is recovered and the errors  
 584 are cleared. This is the default behavior.

585 .It Sy continue  
 586 Returns  
 587 .Er EIO  
 588 to any new write I/O requests but allows reads to any of the remaining healthy  
 589 devices. Any write requests that have yet to be committed to disk would be  
 590 blocked.  
 591 .It Sy panic  
 592 Prints out a message to the console and generates a system crash dump.  
 593 .El  
 594 .It Sy feature@ Ns Ar feature\_name Ns = Ns Sy enabled  
 595 The value of this property is the current state of  
 596 .Ar feature\_name .  
 597 The only valid value when setting this property is  
 598 .Sy enabled  
 599 which moves  
 600 .Ar feature\_name  
 601 to the enabled state. See  
 602 .Xr zpool-features 5  
 603 for details on feature states.  
 604 .It Sy listsnaps Ns = Ns Sy on Ns | Ns Sy off  
 605 Controls whether information about snapshots associated with this pool is  
 606 output when  
 607 .Nm zfs Cm list  
 608 is run without the  
 609 .Fl t  
 610 option. The default value is  
 611 .Sy off .  
 612 .It Sy version Ns = Ns Ar version  
 613 The current on-disk version of the pool. This can be increased, but never  
 614 decreased. The preferred method of updating pools is with the  
 615 .Nm zpool Cm upgrade  
 616 command, though this property can be used when a specific version is needed for  
 617 backwards compatibility. Once feature flags is enabled on a pool this property  
 618 will no longer have a value.  
 619 .El  
 620 .Ss Subcommands  
 621 All subcommands that modify state are logged persistently to the pool in their  
 622 original form.  
 623 .Pp  
 624 The  
 625 .Nm  
 626 command provides subcommands to create and destroy storage pools, add capacity  
 627 to storage pools, and provide information about the storage pools. The  
 628 following subcommands are supported:  
 629 .Bl -tag -width Ds  
 630 .It Xo  
 631 .Nm  
 632 .Fl \?  
 633 .Xc  
 634 Displays a help message.  
 635 .It Xo  
 636 .Nm  
 637 .Cm add  
 638 .Op Fl fn  
 639 .Ar pool vdev Ns ...  
 640 .Xc  
 641 Adds the specified virtual devices to the given pool. The  
 642 .Ar vdev  
 643 specification is described in the  
 644 .Sx Virtual Devices  
 645 section. The behavior of the  
 646 .Fl f  
 647 option, and the device checks performed are described in the  
 648 .Nm zpool Cm create  
 649 subcommand.  
 650 .Bl -tag -width Ds

```

651 .It Fl f
652 Forces use of
653 .Ar vdev Ns s ,
654 even if they appear in use or specify a conflicting replication level. Not all
655 devices can be overridden in this manner.
656 .It Fl n
657 Displays the configuration that would be used without actually adding the
658 .Ar vdev Ns s .
659 The actual pool creation can still fail due to insufficient privileges or
660 device sharing.
661 .El
662 .It Xo
663 .Nm
664 .Cm attach
665 .Op Fl f
666 .Ar pool device new_device
667 .Xc
668 Attaches
669 .Ar new_device
670 to the existing
671 .Ar device .
672 The existing device cannot be part of a raidz configuration. If
673 .Ar device
674 is not currently part of a mirrored configuration,
675 .Ar device
676 automatically transforms into a two-way mirror of
677 .Ar device
678 and
679 .Ar new_device .
680 If
681 .Ar device
682 is part of a two-way mirror, attaching
683 .Ar new_device
684 creates a three-way mirror, and so on. In either case,
685 .Ar new_device
686 begins to resilver immediately.
687 .Bl -tag -width Ds
688 .It Fl f
689 Forces use of
690 .Ar new_device ,
691 even if its appears to be in use. Not all devices can be overridden in this
692 manner.
693 .El
694 .It Xo
695 .Nm
696 .Cm clear
697 .Ar pool
698 .Op Ar device
699 .Xc
700 Clears device errors in a pool. If no arguments are specified, all device
701 errors within the pool are cleared. If one or more devices is specified, only
702 those errors associated with the specified device or devices are cleared.
703 .It Xo
704 .Nm
705 .Cm create
706 .Op Fl dfn
707 .Op Fl m Ar mountpoint
708 .Oo Fl o Ar property Ns = Ns Ar value Oc Ns ...
709 .Oo Fl O Ar file-system-property Ns = Ns Ar value Oc Ns ...
710 .Op Fl R Ar root
711 .Ar pool vdev Ns ...
712 .Xc
713 Creates a new storage pool containing the virtual devices specified on the
714 command line. The pool name must begin with a letter, and can only contain
715 alphanumeric characters as well as underscore
716 .Pq Qq Sy _ ,

```

```

717 dash
718 .Pq Qq Sy - ,
719 and period
720 .Pq Qq Sy \&. .
721 The pool names
722 .Sy mirror ,
723 .Sy raidz ,
724 .Sy spare
725 and
726 .Sy log
727 are reserved, as are names beginning with the pattern
728 .Sy c[0-9] .
729 The
730 .Ar vdev
731 specification is described in the
732 .Sx Virtual Devices
733 section.
734 .Pp
735 The command verifies that each device specified is accessible and not currently
736 in use by another subsystem. There are some uses, such as being currently
737 mounted, or specified as the dedicated dump device, that prevents a device from
738 ever being used by ZFS . Other uses, such as having a preexisting UFS file
739 system, can be overridden with the
740 .Fl f
741 option.
742 .Pp
743 The command also checks that the replication strategy for the pool is
744 consistent. An attempt to combine redundant and non-redundant storage in a
745 single pool, or to mix disks and files, results in an error unless
746 .Fl f
747 is specified. The use of differently sized devices within a single raidz or
748 mirror group is also flagged as an error unless
749 .Fl f
750 is specified.
751 .Pp
752 Unless the
753 .Fl R
754 option is specified, the default mount point is
755 .Pa / Ns Ar pool .
756 The mount point must not exist or must be empty, or else the root dataset
757 cannot be mounted. This can be overridden with the
758 .Fl m
759 option.
760 .Pp
761 By default all supported features are enabled on the new pool unless the
762 .Fl d
763 option is specified.
764 .Bl -tag -width Ds
765 .It Fl d
766 Do not enable any features on the new pool. Individual features can be enabled
767 by setting their corresponding properties to
768 .Sy enabled
769 with the
770 .Fl o
771 option. See
772 .Xr zpool-features 5
773 for details about feature properties.
774 .It Fl f
775 Forces use of
776 .Ar vdev Ns s ,
777 even if they appear in use or specify a conflicting replication level. Not all
778 devices can be overridden in this manner.
779 .It Fl m Ar mountpoint
780 Sets the mount point for the root dataset. The default mount point is
781 .Pa /pool
782 or

```

```

783 .Pa altroot/pool
784 if
785 .Ar altroot
786 is specified. The mount point must be an absolute path,
787 .Sy legacy ,
788 or
789 .Sy none .
790 For more information on dataset mount points, see
791 .Xr zfs 1M .
792 .It Fl n
793 Displays the configuration that would be used without actually creating the
794 pool. The actual pool creation can still fail due to insufficient privileges or
795 device sharing.
796 .It Fl o Ar property Ns = Ns Ar value
797 Sets the given pool properties. See the
798 .Sx Properties
799 section for a list of valid properties that can be set.
800 .It Fl O Ar file-system-property Ns = Ns Ar value
801 Sets the given file system properties in the root file system of the pool. See
802 the
803 .Sx Properties
804 section of
805 .Xr zfs 1M
806 for a list of valid properties that can be set.
807 .It Fl R Ar root
808 Equivalent to
809 .Fl o Sy cachefile Ns = Ns Sy none Fl o Sy altroot Ns = Ns Ar root
810 .El
811 .It Xo
812 .Nm
813 .Cm destroy
814 .Op Fl f
815 .Ar pool
816 .Xc
817 Destroys the given pool, freeing up any devices for other use. This command
818 tries to unmount any active datasets before destroying the pool.
819 .Bl -tag -width Ds
820 .It Fl f
821 Forces any active datasets contained within the pool to be unmounted.
822 .El
823 .It Xo
824 .Nm
825 .Cm detach
826 .Ar pool device
827 .Xc
828 Detaches
829 .Ar device
830 from a mirror. The operation is refused if there are no other valid replicas of
831 the data.
832 .It Xo
833 .Nm
834 .Cm export
835 .Op Fl f
836 .Ar pool Ns ...
837 .Xc
838 Exports the given pools from the system. All devices are marked as exported,
839 but are still considered in use by other subsystems. The devices can be moved
840 between systems
841 .Pq even those of different endianness
842 and imported as long as a sufficient number of devices are present.
843 .Pp
844 Before exporting the pool, all datasets within the pool are unmounted. A pool
845 can not be exported if it has a shared spare that is currently being used.
846 .Pp
847 For pools to be portable, you must give the
848 .Nm

```

```

849 command whole disks, not just slices, so that ZFS can label the disks with
850 portable EFI labels. Otherwise, disk drivers on platforms of different
851 endianness will not recognize the disks.
852 .Bl -tag -width Ds
853 .It Fl f
854 Forcefully unmount all datasets, using the
855 .Nm unmount Fl f
856 command.
857 .Pp
858 This command will forcefully export the pool even if it has a shared spare that
859 is currently being used. This may lead to potential data corruption.
860 .El
861 .It Xo
862 .Nm
863 .Cm get
864 .Op Fl Hp
865 .Op Fl o Ar field Ns Oo , Ns Ar field Oc Ns ...
866 .Sy all Ns | Ns Ar property Ns Oo , Ns Ar property Oc Ns ...
867 .Ar pool Ns ...
868 .Xc
869 Retrieves the given list of properties
870 .Po
871 or all properties if
872 .Sy all
873 is used
874 .Pc
875 for the specified storage pool(s). These properties are displayed with
876 the following fields:
877 .Bd -literal
878     name           Name of storage pool
879     property       Property name
880     value          Property value
881     source         Property source, either 'default' or 'local'.
882 .Ed
883 .Pp
884 See the
885 .Sx Properties
886 section for more information on the available pool properties.
887 .Bl -tag -width Ds
888 .It Fl H
889 Scripted mode. Do not display headers, and separate fields by a single tab
890 instead of arbitrary space.
891 .It Fl o Ar field
892 A comma-separated list of columns to display.
893 .Sy name Ns , Ns Sy property Ns , Ns Sy value Ns , Ns Sy source
894 is the default value.
895 .It Fl p
896 Display numbers in parsable (exact) values.
897 .El
898 .It Xo
899 .Nm
900 .Cm history
901 .Op Fl il
902 .Oo Ar pool Oc Ns ...
903 .Xc
904 Displays the command history of the specified pool(s) or all pools if no pool is
905 specified.
906 .Bl -tag -width Ds
907 .It Fl i
908 Displays internally logged ZFS events in addition to user initiated events.
909 .It Fl l
910 Displays log records in long format, which in addition to standard format
911 includes, the user name, the hostname, and the zone in which the operation was
912 performed.
913 .El
914 .It Xo

```

```

915 .Nm
916 .Cm import
917 .Op Fl D
918 .Op Fl d Ar dir
919 .Xc
920 Lists pools available to import. If the
921 .Fl d
922 option is not specified, this command searches for devices in
923 .Pa /dev/dsk .
924 The
925 .Fl d
926 option can be specified multiple times, and all directories are searched. If the
927 device appears to be part of an exported pool, this command displays a summary
928 of the pool with the name of the pool, a numeric identifier, as well as the vdev
929 layout and current health of the device for each device or file. Destroyed
930 pools, pools that were previously destroyed with the
931 .Nm zpool Cm destroy
932 command, are not listed unless the
933 .Fl D
934 option is specified.
935 .Pp
936 The numeric identifier is unique, and can be used instead of the pool name when
937 multiple exported pools of the same name are available.
938 .Bl -tag -width Ds
939 .It Fl c Ar cachefile
940 Reads configuration from the given
941 .Ar cachefile
942 that was created with the
943 .Sy cachefile
944 pool property. This
945 .Ar cachefile
946 is used instead of searching for devices.
947 .It Fl d Ar dir
948 Searches for devices or files in
949 .Ar dir .
950 The
951 .Fl d
952 option can be specified multiple times.
953 .It Fl D
954 Lists destroyed pools only.
955 .El
956 .It Xo
957 .Nm
958 .Cm import
959 .Fl a
960 .Op Fl DfmN
961 .Op Fl F Op Fl n
962 .Op Fl c Ar cachefile Ns | Ns Fl d Ar dir
963 .Op Fl o Ar mntopts
964 .Oo Fl o Ar property Ns = Ns Ar value Oc Ns ...
965 .Op Fl R Ar root
966 .Xc
967 Imports all pools found in the search directories. Identical to the previous
968 command, except that all pools with a sufficient number of devices available are
969 imported. Destroyed pools, pools that were previously destroyed with the
970 .Nm zpool Cm destroy
971 command, will not be imported unless the
972 .Fl D
973 option is specified.
974 .Bl -tag -width Ds
975 .It Fl a
976 Searches for and imports all pools found.
977 .It Fl c Ar cachefile
978 Reads configuration from the given
979 .Ar cachefile
980 that was created with the

```

```

981 .Sy cachefile
982 pool property. This
983 .Ar cachefile
984 is used instead of searching for devices.
985 .It Fl d Ar dir
986 Searches for devices or files in
987 .Ar dir .
988 The
989 .Fl d
990 option can be specified multiple times. This option is incompatible with the
991 .Fl c
992 option.
993 .It Fl D
994 Imports destroyed pools only. The
995 .Fl f
996 option is also required.
997 .It Fl f
998 Forces import, even if the pool appears to be potentially active.
999 .It Fl F
1000 Recovery mode for a non-importable pool. Attempt to return the pool to an
1001 importable state by discarding the last few transactions. Not all damaged pools
1002 can be recovered by using this option. If successful, the data from the
1003 discarded transactions is irretrievably lost. This option is ignored if the pool
1004 is importable or already imported.
1005 .It Fl m
1006 Allows a pool to import when there is a missing log device. Recent transactions
1007 can be lost because the log device will be discarded.
1008 .It Fl n
1009 Used with the
1010 .Fl F
1011 recovery option. Determines whether a non-importable pool can be made importable
1012 again, but does not actually perform the pool recovery. For more details about
1013 pool recovery mode, see the
1014 .Fl F
1015 option, above.
1016 .It Fl N
1017 Import the pool without mounting any file systems.
1018 .It Fl o Ar mntopts
1019 Comma-separated list of mount options to use when mounting datasets within the
1020 pool. See
1021 .Xr zfs 1M
1022 for a description of dataset properties and mount options.
1023 .It Fl o Ar property Ns = Ns Ar value
1024 Sets the specified property on the imported pool. See the
1025 .Sx Properties
1026 section for more information on the available pool properties.
1027 .It Fl R Ar root
1028 Sets the
1029 .Sy cachefile
1030 property to
1031 .Sy none
1032 and the
1033 .Sy altroot
1034 property to
1035 .Ar root .
1036 .El
1037 .It Xo
1038 .Nm
1039 .Cm import
1040 .Op Fl Dfm
1041 .Op Fl F Op Fl n
1042 .Op Fl c Ar cachefile Ns | Ns Fl d Ar dir
1043 .Op Fl o Ar mntopts
1044 .Oo Fl o Ar property Ns = Ns Ar value Oc Ns ...
1045 .Op Fl R Ar root
1046 .Ar pool Ns | Ns Ar id

```



1047 .Op Ar newpool  
 1048 .Xc  
 1049 Imports a specific pool. A pool can be identified by its name or the numeric  
 1050 identifier. If  
 1051 .Ar newpool  
 1052 is specified, the pool is imported using the name  
 1053 .Ar newpool .  
 1054 Otherwise, it is imported with the same name as its exported name.  
 1055 .Pp  
 1056 If a device is removed from a system without running  
 1057 .Nm zpool Cm export  
 1058 first, the device appears as potentially active. It cannot be determined if  
 1059 this was a failed export, or whether the device is really in use from another  
 1060 host. To import a pool in this state, the  
 1061 .Fl f  
 1062 option is required.  
 1063 .Bl -tag -width Ds  
 1064 .It Fl c Ar cachefile  
 1065 Reads configuration from the given  
 1066 .Ar cachefile  
 1067 that was created with the  
 1068 .Sy cachefile  
 1069 pool property. This  
 1070 .Ar cachefile  
 1071 is used instead of searching for devices.  
 1072 .It Fl d Ar dir  
 1073 Searches for devices or files in  
 1074 .Ar dir .  
 1075 The  
 1076 .Fl d  
 1077 option can be specified multiple times. This option is incompatible with the  
 1078 .Fl c  
 1079 option.  
 1080 .It Fl D  
 1081 Imports destroyed pool. The  
 1082 .Fl f  
 1083 option is also required.  
 1084 .It Fl f  
 1085 Forces import, even if the pool appears to be potentially active.  
 1086 .It Fl F  
 1087 Recovery mode for a non-importable pool. Attempt to return the pool to an  
 1088 importable state by discarding the last few transactions. Not all damaged pools  
 1089 can be recovered by using this option. If successful, the data from the  
 1090 discarded transactions is irretrievably lost. This option is ignored if the pool  
 1091 is importable or already imported.  
 1092 .It Fl m  
 1093 Allows a pool to import when there is a missing log device. Recent transactions  
 1094 can be lost because the log device will be discarded.  
 1095 .It Fl n  
 1096 Used with the  
 1097 .Fl F  
 1098 recovery option. Determines whether a non-importable pool can be made importable  
 1099 again, but does not actually perform the pool recovery. For more details about  
 1100 pool recovery mode, see the  
 1101 .Fl F  
 1102 option, above.  
 1103 .It Fl o Ar mntopts  
 1104 Comma-separated list of mount options to use when mounting datasets within the  
 1105 pool. See  
 1106 .Xr zfs 1M  
 1107 for a description of dataset properties and mount options.  
 1108 .It Fl o Ar property Ns = Ns Ar value  
 1109 Sets the specified property on the imported pool. See the  
 1110 .Sx Properties  
 1111 section for more information on the available pool properties.  
 1112 .It Fl R Ar root

1113 Sets the  
 1114 .Sy cachefile  
 1115 property to  
 1116 .Sy none  
 1117 and the  
 1118 .Sy altroot  
 1119 property to  
 1120 .Ar root .  
 1121 .El  
 1122 .It Xo  
 1123 .Nm  
 1124 .Cm iostat  
 1125 .Op Fl v  
 1126 .Op Fl T Sy u Ns | Ns Sy d  
 1127 .Oo Ar pool Oc Ns ...  
 1128 .Op Ar interval Op Ar count  
 1129 .Xc  
 1130 Displays I/O statistics for the given pools. When given an  
 1131 .Ar interval ,  
 1132 the statistics are printed every  
 1133 .Ar interval  
 1134 seconds until ^C is pressed. If no  
 1135 .Ar pool Ns s  
 1136 are specified, statistics for every pool in the system is shown. If  
 1137 .Ar count  
 1138 is specified, the command exits after  
 1139 .Ar count  
 1140 reports are printed.  
 1141 .Bl -tag -width Ds  
 1142 .It Fl T Sy u Ns | Ns Sy d  
 1143 Display a time stamp. Specify  
 1144 .Sy u  
 1145 for a printed representation of the internal representation of time. See  
 1146 .Xr time 2 .  
 1147 Specify  
 1148 .Sy d  
 1149 for standard date format. See  
 1150 .Xr date 1 .  
 1151 .It Fl v  
 1152 Verbose statistics. Reports usage statistics for individual vdevs within the  
 1153 pool, in addition to the pool-wide statistics.  
 1154 .El  
 1155 .It Xo  
 1156 .Nm  
 1157 .Cm list  
 1158 .Op Fl Hpv  
 1159 .Op Fl o Ar property Ns Oo , Ns Ar property Oc Ns ...  
 1160 .Op Fl T Sy u Ns | Ns Sy d  
 1161 .Oo Ar pool Oc Ns ...  
 1162 .Op Ar interval Op Ar count  
 1163 .Xc  
 1164 Lists the given pools along with a health status and space usage. If no  
 1165 .Ar pool Ns s  
 1166 are specified, all pools in the system are listed. When given an  
 1167 .Ar interval ,  
 1168 the information is printed every  
 1169 .Ar interval  
 1170 seconds until ^C is pressed. If  
 1171 .Ar count  
 1172 is specified, the command exits after  
 1173 .Ar count  
 1174 reports are printed.  
 1175 .Bl -tag -width Ds  
 1176 .It Fl H  
 1177 Scripted mode. Do not display headers, and separate fields by a single tab  
 1178 instead of arbitrary space.

1179 .It Fl o Ar property  
 1180 Comma-separated list of properties to display. See the  
 1181 .Sx Properties  
 1182 section for a list of valid properties. The default list is  
 1183 .Sy name , size , used , available , fragmentation , expandsize , capacity ,  
 1184 .Sy dedupratio , health , altroot .  
 1185 .It Fl p  
 1186 Display numbers in parsable  
 1187 .Pq exact  
 1188 values.  
 1189 .It Fl T Sy u Ns | Ns Sy d  
 1190 Display a time stamp. Specify  
 1191 .Fl u  
 1192 for a printed representation of the internal representation of time. See  
 1193 .Xr time 2 .  
 1194 Specify  
 1195 .Fl d  
 1196 for standard date format. See  
 1197 .Xr date 1 .  
 1198 .It Fl v  
 1199 Verbose statistics. Reports usage statistics for individual vdevs within the  
 1200 pool, in addition to the pool-wise statistics.  
 1201 .El  
 1202 .It Xo  
 1203 .Nm  
 1204 .Cm offline  
 1205 .Op Fl t  
 1206 .Ar pool Ar device Ns ...  
 1207 .Xc  
 1208 Takes the specified physical device offline. While the  
 1209 .Ar device  
 1210 is offline, no attempt is made to read or write to the device. This command is  
 1211 not applicable to spares.  
 1212 .Bl -tag -width Ds  
 1213 .It Fl t  
 1214 Temporary. Upon reboot, the specified physical device reverts to its previous  
 1215 state.  
 1216 .El  
 1217 .It Xo  
 1218 .Nm  
 1219 .Cm online  
 1220 .Op Fl e  
 1221 .Ar pool Ar device Ns ...  
 1222 .Xc  
 1223 Brings the specified physical device online. This command is not applicable to  
 1224 spares.  
 1225 .Bl -tag -width Ds  
 1226 .It Fl e  
 1227 Expand the device to use all available space. If the device is part of a mirror  
 1228 or raidz then all devices must be expanded before the new space will become  
 1229 available to the pool.  
 1230 .El  
 1231 .It Xo  
 1232 .Nm  
 1233 .Cm reguid  
 1234 .Ar pool  
 1235 .Xc  
 1236 Generates a new unique identifier for the pool. You must ensure that all devices  
 1237 in this pool are online and healthy before performing this action.  
 1238 .It Xo  
 1239 .Nm  
 1240 .Cm reopen  
 1241 .Ar pool  
 1242 .Xc  
 1243 Reopen all the vdevs associated with the pool.  
 1244 .It Xo

1245 .Nm  
 1246 .Cm remove  
 1247 .Ar pool Ar device Ns ...  
 1248 .Xc  
 1249 Removes the specified device from the pool. This command currently only supports  
 1250 removing hot spares, cache, and log devices. A mirrored log device can be  
 1251 removed by specifying the top-level mirror for the log. Non-log devices that are  
 1252 part of a mirrored configuration can be removed using the  
 1253 .Nm zpool Cm detach  
 1254 command. Non-redundant and raidz devices cannot be removed from a pool.  
 1255 .It Xo  
 1256 .Nm  
 1257 .Cm replace  
 1258 .Op Fl f  
 1259 .Ar pool Ar device Op Ar new\_device  
 1260 .Xc  
 1261 Replaces  
 1262 .Ar old\_device  
 1263 with  
 1264 .Ar new\_device .  
 1265 This is equivalent to attaching  
 1266 .Ar new\_device ,  
 1267 waiting for it to resilver, and then detaching  
 1268 .Ar old\_device .  
 1269 .Pp  
 1270 The size of  
 1271 .Ar new\_device  
 1272 must be greater than or equal to the minimum size of all the devices in a mirror  
 1273 or raidz configuration.  
 1274 .Pp  
 1275 .Ar new\_device  
 1276 is required if the pool is not redundant. If  
 1277 .Ar new\_device  
 1278 is not specified, it defaults to  
 1279 .Ar old\_device .  
 1280 This form of replacement is useful after an existing disk has failed and has  
 1281 been physically replaced. In this case, the new disk may have the same  
 1282 .Pa /dev/dsk  
 1283 path as the old device, even though it is actually a different disk. ZFS  
 1284 recognizes this.  
 1285 .Bl -tag -width Ds  
 1286 .It Fl f  
 1287 Forces use of  
 1288 .Ar new\_device ,  
 1289 even if its appears to be in use. Not all devices can be overridden in this  
 1290 manner.  
 1291 .El  
 1292 .It Xo  
 1293 .Nm  
 1294 .Cm scrub  
 1295 .Op Fl s  
 1296 .Ar pool Ns ...  
 1297 .Xc  
 1298 Begins a scrub. The scrub examines all data in the specified pools to verify  
 1299 that it checksums correctly. For replicated  
 1300 .Pq mirror or raidz  
 1301 devices, ZFS automatically repairs any damage discovered during the scrub. The  
 1302 .Nm zpool Cm status  
 1303 command reports the progress of the scrub and summarizes the results of the  
 1304 scrub upon completion.  
 1305 .Pp  
 1306 Scrubbing and resilvering are very similar operations. The difference is that  
 1307 resilvering only examines data that ZFS knows to be out of date  
 1308 .Po  
 1309 for example, when attaching a new device to a mirror or replacing an existing  
 1310 device

1311 .Pc ,  
 1312 whereas scrubbing examines all data to discover silent errors due to hardware  
 1313 faults or disk failure.  
 1314 .Pp  
 1315 Because scrubbing and resilvering are I/O-intensive operations, ZFS only allows  
 1316 one at a time. If a scrub is already in progress, the  
 1317 .Nm zpool Cm scrub  
 1318 command terminates it and starts a new scrub. If a resilver is in progress, ZFS  
 1319 does not allow a scrub to be started until the resilver completes.  
 1320 .Bl -tag -width Ds  
 1321 .It Fl s  
 1322 Stop scrubbing.  
 1323 .El  
 1324 .It Xo  
 1325 .Nm  
 1326 .Cm set  
 1327 .Ar property Ns = Ns Ar value  
 1328 .Ar pool  
 1329 .Xc  
 1330 Sets the given property on the specified pool. See the  
 1331 .Sx Properties  
 1332 section for more information on what properties can be set and acceptable  
 1333 values.  
 1334 .It Xo  
 1335 .Nm  
 1336 .Cm split  
 1337 .Op Fl n  
 1338 .Oo Fl o Ar property Ns = Ns Ar value Oc Ns ...  
 1339 .Op Fl R Ar root  
 1340 .Ar pool newpool  
 1341 .Xc  
 1342 Splits devices off  
 1343 .Ar pool  
 1344 creating  
 1345 .Ar newpool .  
 1346 All vdevs in  
 1347 .Ar pool  
 1348 must be mirrors. At the time of the split,  
 1349 .Ar newpool  
 1350 will be a replica of  
 1351 .Ar pool .  
 1352 .Bl -tag -width Ds  
 1353 .It Fl n  
 1354 Do dry run, do not actually perform the split. Print out the expected  
 1355 configuration of  
 1356 .Ar newpool .  
 1357 .It Fl o Ar property Ns = Ns Ar value  
 1358 Sets the specified property for  
 1359 .Ar newpool .  
 1360 See the  
 1361 .Sx Properties  
 1362 section for more information on the available pool properties.  
 1363 .It Fl R Ar root  
 1364 Set  
 1365 .Sy altroot  
 1366 for  
 1367 .Ar newpool  
 1368 to  
 1369 .Ar root  
 1370 and automatically import it.  
 1371 .El  
 1372 .It Xo  
 1373 .Nm  
 1374 .Cm status  
 1375 .Op Fl Dvx  
 1376 .Op Fl T Sy u Ns | Ns Sy d

1377 .Oo Ar pool Oc Ns ...  
 1378 .Op Ar interval Op Ar count  
 1379 .Xc  
 1380 Displays the detailed health status for the given pools. If no  
 1381 .Ar pool  
 1382 is specified, then the status of each pool in the system is displayed. For more  
 1383 information on pool and device health, see the  
 1384 .Sx Device Failure and Recovery  
 1385 section.  
 1386 .Pp  
 1387 If a scrub or resilver is in progress, this command reports the percentage done  
 1388 and the estimated time to completion. Both of these are only approximate,  
 1389 because the amount of data in the pool and the other workloads on the system can  
 1390 change.  
 1391 .Bl -tag -width Ds  
 1392 .It Fl D  
 1393 Display a histogram of deduplication statistics, showing the allocated  
 1394 .Pq physically present on disk  
 1395 and referenced  
 1396 .Pq logically referenced in the pool  
 1397 block counts and sizes by reference count.  
 1398 .It Fl T Sy u Ns | Ns Sy d  
 1399 Display a time stamp. Specify  
 1400 .Fl u  
 1401 for a printed representation of the internal representation of time. See  
 1402 .Xr time 2 .  
 1403 Specify  
 1404 .Fl d  
 1405 for standard date format. See  
 1406 .Xr date 1 .  
 1407 .It Fl v  
 1408 Displays verbose data error information, printing out a complete list of all  
 1409 data errors since the last complete pool scrub.  
 1410 .It Fl x  
 1411 Only display status for pools that are exhibiting errors or are otherwise  
 1412 unavailable. Warnings about pools not using the latest on-disk format will not  
 1413 be included.  
 1414 .El  
 1415 .It Xo  
 1416 .Nm  
 1417 .Cm upgrade  
 1418 .Xc  
 1419 Displays pools which do not have all supported features enabled and pools  
 1420 formatted using a legacy ZFS version number. These pools can continue to be  
 1421 used, but some features may not be available. Use  
 1422 .Nm zpool Cm upgrade Fl a  
 1423 to enable all features on all pools.  
 1424 .It Xo  
 1425 .Nm  
 1426 .Cm upgrade  
 1427 .Fl v  
 1428 .Xc  
 1429 Displays legacy ZFS versions supported by the current software. See  
 1430 .Xr zpool-features 5  
 1431 for a description of feature flags features supported by the current software.  
 1432 .It Xo  
 1433 .Nm  
 1434 .Cm upgrade  
 1435 .Op Fl V Ar version  
 1436 .Fl a Ns | Ns Ar pool Ns ...  
 1437 .Xc  
 1438 Enables all supported features on the given pool. Once this is done, the pool  
 1439 will no longer be accessible on systems that do not support feature flags. See  
 1440 .Xr zpool-features 5  
 1441 for details on compatibility with systems that support feature flags, but do not  
 1442 support all features enabled on the pool.

```

1443 .Bl -tag -width Ds
1444 .It Fl a
1445 Enables all supported features on all pools.
1446 .It Fl V Ar version
1447 Upgrade to the specified legacy version. If the
1448 .Fl V
1449 flag is specified, no features will be enabled on the pool. This option can only
1450 be used to increase the version number up to the last supported legacy version
1451 number.
1452 .El
1453 .El
1454 .Sh EXIT STATUS
1455 The following exit values are returned:
1456 .Bl -tag -width Ds
1457 .It Sy 0
1458 Successful completion.
1459 .It Sy 1
1460 An error occurred.
1461 .It Sy 2
1462 Invalid command line options were specified.
1463 .El
1464 .Sh EXAMPLES
1465 .Bl -tag -width Ds
1466 .It Sy Example 1 No Creating a RAID-Z Storage Pool
1467 The following command creates a pool with a single raidz root vdev that
1468 consists of six disks.
1469 .Bd -literal
1470 # zpool create tank raidz c0t0d0 c0t1d0 c0t2d0 c0t3d0 c0t4d0 c0t5d0
1471 .Ed
1472 .It Sy Example 2 No Creating a Mirrored Storage Pool
1473 The following command creates a pool with two mirrors, where each mirror
1474 contains two disks.
1475 .Bd -literal
1476 # zpool create tank mirror c0t0d0 c0t1d0 mirror c0t2d0 c0t3d0
1477 .Ed
1478 .It Sy Example 3 No Creating a ZFS Storage Pool by Using Slices
1479 The following command creates an unmirrored pool using two disk slices.
1480 .Bd -literal
1481 # zpool create tank /dev/dsk/c0t0d0s1 c0t1d0s4
1482 .Ed
1483 .It Sy Example 4 No Creating a ZFS Storage Pool by Using Files
1484 The following command creates an unmirrored pool using files. While not
1485 recommended, a pool based on files can be useful for experimental purposes.
1486 .Bd -literal
1487 # zpool create tank /path/to/file/a /path/to/file/b
1488 .Ed
1489 .It Sy Example 5 No Adding a Mirror to a ZFS Storage Pool
1490 The following command adds two mirrored disks to the pool
1491 .Em tank ,
1492 assuming the pool is already made up of two-way mirrors. The additional space
1493 is immediately available to any datasets within the pool.
1494 .Bd -literal
1495 # zpool add tank mirror c1t0d0 c1t1d0
1496 .Ed
1497 .It Sy Example 6 No Listing Available ZFS Storage Pools
1498 The following command lists all available pools on the system. In this case,
1499 the pool
1500 .Em zion
1501 is faulted due to a missing device. The results from this command are similar
1502 to the following:
1503 .Bd -literal
1504 # zpool list
1505 NAME      SIZE  ALLOC   FREE   FRAG  EXPANDSZ   CAP  DEDUP  HEALTH  ALTROOT
1506 rpool    19.9G  8.43G  11.4G   33%    -         42%  1.00x  ONLINE  -
1507 tank     61.5G  20.0G  41.5G   48%    -         32%  1.00x  ONLINE  -
1508 zion      -      -      -      -      -         -    -     FAULTED -

```

```

1509 .Ed
1510 .It Sy Example 7 No Destroying a ZFS Storage Pool
1511 The following command destroys the pool
1512 .Em tank
1513 and any datasets contained within.
1514 .Bd -literal
1515 # zpool destroy -f tank
1516 .Ed
1517 .It Sy Example 8 No Exporting a ZFS Storage Pool
1518 The following command exports the devices in pool
1519 .Em tank
1520 so that they can be relocated or later imported.
1521 .Bd -literal
1522 # zpool export tank
1523 .Ed
1524 .It Sy Example 9 No Importing a ZFS Storage Pool
1525 The following command displays available pools, and then imports the pool
1526 .Em tank
1527 for use on the system. The results from this command are similar to the
1528 following:
1529 .Bd -literal
1530 # zpool import
1531   pool: tank
1532   id: 15451357997522795478
1533   state: ONLINE
1534 action: The pool can be imported using its name or numeric identifier.
1535 config:
1536
1537         tank      ONLINE
1538         mirror    ONLINE
1539         c1t2d0    ONLINE
1540         c1t3d0    ONLINE
1541
1542 # zpool import tank
1543 .Ed
1544 .It Sy Example 10 No Upgrading All ZFS Storage Pools to the Current Version
1545 The following command upgrades all ZFS Storage pools to the current version of
1546 the software.
1547 .Bd -literal
1548 # zpool upgrade -a
1549 This system is currently running ZFS version 2.
1550 .Ed
1551 .It Sy Example 11 No Managing Hot Spares
1552 The following command creates a new pool with an available hot spare:
1553 .Bd -literal
1554 # zpool create tank mirror c0t0d0 c0t1d0 spare c0t2d0
1555 .Ed
1556 .Pp
1557 If one of the disks were to fail, the pool would be reduced to the degraded
1558 state. The failed device can be replaced using the following command:
1559 .Bd -literal
1560 # zpool replace tank c0t0d0 c0t3d0
1561 .Ed
1562 .Pp
1563 Once the data has been resilvered, the spare is automatically removed and is
1564 made available should another device fails. The hot spare can be permanently
1565 removed from the pool using the following command:
1566 .Bd -literal
1567 # zpool remove tank c0t2d0
1568 .Ed
1569 .It Sy Example 12 No Creating a ZFS Pool with Mirrored Separate Intent Logs
1570 The following command creates a ZFS storage pool consisting of two, two-way
1571 mirrors and mirrored log devices:
1572 .Bd -literal
1573 # zpool create pool mirror c0d0 c1d0 mirror c2d0 c3d0 log mirror \e
1574   c4d0 c5d0

```

```

1575 .Ed
1576 .It Sy Example 13 No Adding Cache Devices to a ZFS Pool
1577 The following command adds two disks for use as cache devices to a ZFS storage
1578 pool:
1579 .Bd -literal
1580 # zpool add pool cache c2d0 c3d0
1581 .Ed
1582 .Pp
1583 Once added, the cache devices gradually fill with content from main memory.
1584 Depending on the size of your cache devices, it could take over an hour for
1585 them to fill. Capacity and reads can be monitored using the
1586 .Cm iostat
1587 option as follows:
1588 .Bd -literal
1589 # zpool iostat -v pool 5
1590 .Ed
1591 .It Sy Example 14 No Removing a Mirrored Log Device
1592 The following command removes the mirrored log device
1593 .Sy mirror-2 .
1594 Given this configuration:
1595 .Bd -literal
1596 pool: tank
1597 state: ONLINE
1598 scrub: none requested
1599 config:

```

NAME	STATE	READ	WRITE	CKSUM
tank	ONLINE	0	0	0
mirror-0	ONLINE	0	0	0
c6t0d0	ONLINE	0	0	0
c6t1d0	ONLINE	0	0	0
mirror-1	ONLINE	0	0	0
c6t2d0	ONLINE	0	0	0
c6t3d0	ONLINE	0	0	0
logs				
mirror-2	ONLINE	0	0	0
c4t0d0	ONLINE	0	0	0
c4t1d0	ONLINE	0	0	0

```

1613 .Ed
1614 .Pp
1615 The command to remove the mirrored log
1616 .Sy mirror-2
1617 is:
1618 .Bd -literal
1619 # zpool remove tank mirror-2
1620 .Ed
1621 .It Sy Example 15 No Displaying expanded space on a device
1622 The following command displays the detailed information for the pool
1623 .Em data .
1624 This pool is comprised of a single raidz vdev where one of its devices
1625 increased its capacity by 10GB. In this example, the pool will not be able to
1626 utilize this extra capacity until all the devices under the raidz vdev have
1627 been expanded.
1628 .Bd -literal
1629 # zpool list -v data
1630 NAME      SIZE  ALLOC  FREE  FRAG  EXPANDSZ  CAP  DEDUP  HEALTH  ALTROOT
1631 data     23.9G 14.6G  9.30G  48%   -         61%  1.00x  ONLINE  -
1632 raidz1   23.9G 14.6G  9.30G  48%   -         -
1633  clt1d0   -     -     -     -     -         -
1634  clt2d0   -     -     -     -     10G      -
1635  clt3d0   -     -     -     -     -         -
1636 .Ed
1637 .El
1638 .Sh INTERFACE STABILITY
1639 .Sy Evolving
1640 .Sh SEE ALSO

```

```

1641 .Xr zfs 1M ,
1642 .Xr attributes 5 ,
1643 .Xr zpool-features 5

```