






















?? 2 : ?? ?? (Virtual Devices)

在 2000 年 10 月，苹果公司发布了 Mac OS X 操作系统。Mac OS X 是基于 Unix 的操作系统，它使用 Mach 内核和 BSD 系统调用接口。Mac OS X 的发布标志着苹果公司在操作系统领域的重大突破，也为后来的 iPhone 和 iPad 等设备的开发奠定了基础。

ZFS                     

??? ? ?? ?? ?? (Disks and Other Storage Media)

ZFS 的 数据 是 以 块 的 形式 存储 的 。 FreeBSD 的 GEOM 系统 是 一个 数据 转换 层 ， 它 在 物理 设备 和 文件系统 之间 提供 一个 抽象 的 接口 。 ZFS 的 数据 是 以 块 的 形式 存储 的 ， 而 不是 以 文件 的 形式 存储 的 。 ZFS 的 数据 是 以 块 的 形式 存储 的 ， 而 不是 以 文件 的 形式 存储 的 。

?? ??? ??? (Raw Disk Storage)

1. 在“文件”菜单中选择“打开”，在弹出的“打开”对话框中，选择要打开的文件。

1. 在 1990 年，Linux 操作系统是由 Linus Torvalds 创建的。它是一个开源的操作系统，旨在提供一个类似于 Unix 的操作系统。Linux 最初是在芬兰的一个大学实验室中开发的，后来成为了一个全球性的开源项目。

0 00 00 ,00 0000 00 0000 000 00 00000 0 00 000 0000 . 6TB 0000
 000 00 00 00 000 ,000000 000 000 0 000000 0000 .0000 000 00
 000 000 0000 000 0000 000 00 00 00000 .0000 000000 0 0000
 00 00 0000 00 00 000 00 00 000 00 000 0000 0000 00 0 0000 0
 0 0000 .00 512000 000 0000 00000 4096000 (4K, 0000000 0) 000
 0000 00000 0000 000 00 000000 .00 000000 80 0000 00 0 00 00
 000 000 0000 0000 .000 000000 0000 00 00 000 0 00000 00
 000000 0 0000 0 0000 0 0000 0 00 00 00 0 0000 .

???

01 02 0304 0506 07 0809 1011 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 103

本文將探討 Solaris ZFS 在 FreeBSD 上的實現與配置。

Solaris ZFS 是一個高度可靠的存儲系統，它最初是由 Sun 公司開發的。Solaris 系統使用 ZFS 作為其默認的存儲系統。FreeBSD 也支持 ZFS，這使得它成為一個非常靈活和強大的存儲解決方案。FreeBSD 的 ZFS 實現與 Solaris 的實現非常相似，這使得數據可以在兩者之間輕鬆遷移。

FreeBSD 的 ZFS 實現支持多種分區表格式，包括 MBR 和 GPT。GPT 格式支持更大的分區大小，最高可達 128TB。MBR 格式則支持最大 2TB 的分區大小。FreeBSD 的 ZFS 實現還支持 64KB 的塊大小，這使得它在處理大量數據時非常高效。FreeBSD 的 ZFS 實現還支持多種加密算法，包括 AES-XTS 和 AES-NI。

在 FreeBSD 上配置 ZFS，您可以使用 `bsdlabe(8)` 命令來創建和配置 ZFS 池。

FreeBSD 的 ZFS 實現還支持多種快照和克隆功能，這使得它成為一個非常適合數據備份和恢復的系統。此外，ZFS 還支持多種數據壓縮和去重算法，這可以顯著減少存儲空間的佔用。

GEOM 設備存儲 (GEOM Device Storage)

FreeBSD 的 GEOM 設備存儲系統是一個非常強大的存儲管理工具。它允許您將多個物理存儲設備組合成一個邏輯存儲池，並對該池進行各種操作，如加密、壓縮和去重。GEOM 設備存儲系統還支持多種數據格式，包括 UFS 和 ZFS。此外，GEOM 設備存儲系統還支持多種數據傳輸協議，包括 SATA 和 SCSI。

FreeBSD 的 GEOM 設備存儲系統還支持多種加密算法，包括 AES-XTS 和 AES-NI。此外，GEOM 設備存儲系統還支持多種數據壓縮和去重算法，這可以顯著減少存儲空間的佔用。GEOM 設備存儲系統還支持多種數據傳輸協議，包括 SATA 和 SCSI。此外，GEOM 設備存儲系統還支持多種數據格式，包括 UFS 和 ZFS。此外，GEOM 設備存儲系統還支持多種數據傳輸協議，包括 SATA 和 SCSI。

FreeBSD 的 GEOM 設備存儲系統還支持多種數據格式，包括 UFS 和 ZFS。此外，GEOM 設備存儲系統還支持多種數據傳輸協議，包括 SATA 和 SCSI。此外，GEOM 設備存儲系統還支持多種數據格式，包括 UFS 和 ZFS。此外，GEOM 設備存儲系統還支持多種數據傳輸協議，包括 SATA 和 SCSI。

GEOM 提供了 对 物理 磁盘 的 抽象 和 管理 。 disk ident, gptid, GPT 分区 表 的 信息 通过 GEOM-specific glabel 来 标识 。 这 使得 磁盘 的 管理 更加 灵活 和 安全 。

GEOM 的 核心 是 对 物理 磁盘 的 抽象 。 通过 使用 GEOM 的 接口 ， 用户 可以 对 磁盘 进行 分区 、 格式化 和 挂载 操作 。 例如 ， 使用 HBA(RAID 控制器 的 磁盘 阵列) 时 ， 用户 可以 通过 GEOM 来 管理 阵列 的 状态 和 性能 。 此外 ， GEOM 还支持 对 磁盘 的 加密 和 压缩 操作 。 通过 使用 GEOM 的 加密 和 压缩 接口 ， 用户 可以 对 磁盘 进行 加密 和 压缩 ， 以 保护 数据 的 安全 和 节省 空间 。 这 使得 GEOM 成为 一个 非常 强大 和 灵活 的 磁盘 管理 工具 。

在 FreeBSD 中 ， GEOM 的 使用 非常 广泛 。 它 不仅 支持 对 物理 磁盘 的 管理 ， 还支持 对 虚拟 磁盘 的 管理 。 通过 使用 GEOM 的 接口 ， 用户 可以 对 虚拟 磁盘 进行 分区 、 格式化 和 挂载 操作 。 此外 ， GEOM 还支持 对 虚拟 磁盘 的 加密 和 压缩 操作 。 通过 使用 GEOM 的 加密 和 压缩 接口 ， 用户 可以 对 虚拟 磁盘 进行 加密 和 压缩 ， 以 保护 数据 的 安全 和 节省 空间 。 这 使得 GEOM 成为 一个 非常 强大 和 灵活 的 磁盘 管理 工具 。

File-Backed Storage

在 FreeBSD 中 ， ZFS 文件系统 可以 使用 文件 作为 存储 介质 。 通过 使用 ZFS 的 接口 ， 用户 可以 对 文件 进行 分区 、 格式化 和 挂载 操作 。 此外 ， ZFS 还支持 对 文件 的 加密 和 压缩 操作 。 通过 使用 ZFS 的 加密 和 压缩 接口 ， 用户 可以 对 文件 进行 加密 和 压缩 ， 以 保护 数据 的 安全 和 节省 空间 。 这 使得 ZFS 成为 一个 非常 强大 和 灵活 的 文件 存储 系统 。

Providers vs. Disks

在 FreeBSD 中 ， GEOM 的 使用 非常 广泛 。 它 不仅 支持 对 物理 磁盘 的 管理 ， 还支持 对 虚拟 磁盘 的 管理 。 通过 使用 GEOM 的 接口 ， 用户 可以 对 虚拟 磁盘 进行 分区 、 格式化 和 挂载 操作 。 此外 ， GEOM 还支持 对 虚拟 磁盘 的 加密 和 压缩 操作 。 通过 使用 GEOM 的 加密 和 压缩 接口 ， 用户 可以 对 虚拟 磁盘 进行 加密 和 压缩 ， 以 保护 数据 的 安全 和 节省 空间 。 这 使得 GEOM 成为 一个 非常 强大 和 灵活 的 磁盘 管理 工具 。

FreeBSD 的 磁盘 管理 系统 非常 强大 和 灵活 。 它 不仅 支持 对 物理 磁盘 的 管理 ， 还支持 对 虚拟 磁盘 的 管理 。 通过 使用 GEOM 的 接口 ， 用户 可以 对 虚拟 磁盘 进行 分区 、 格式化 和 挂载 操作 。 此外 ， GEOM 还支持 对 虚拟 磁盘 的 加密 和 压缩 操作 。 通过 使用 GEOM 的 加密 和 压缩 接口 ， 用户 可以 对 虚拟 磁盘 进行 加密 和 压缩 ， 以 保护 数据 的 安全 和 节省 空间 。 这 使得 GEOM 成为 一个 非常 强大 和 灵活 的 磁盘 管理 工具 。

在 FreeBSD 中 ， ZFS 文件系统 可以 使用 文件 作为 存储 介质 。 通过 使用 ZFS 的 接口 ， 用户 可以 对 文件 进行 分区 、 格式化 和 挂载 操作 。 此外 ， ZFS 还支持 对 文件 的 加密 和 压缩 操作 。 通过 使用 ZFS 的 加密 和 压缩 接口 ， 用户 可以 对 文件 进行 加密 和 压缩 ， 以 保护 数据 的 安全 和 节省 空间 。 这 使得 ZFS 成为 一个 非常 强大 和 灵活 的 文件 存储 系统 。

VDEV: Virtual Devices

在 FreeBSD 中 ， VDEV 是 一个 非常 重要 的 概念 。 它 允许 用户 对 物理 磁盘 进行 抽象 和 管理 。 通过 使用 VDEV 的 接口 ， 用户 可以 对 物理 磁盘 进行 分区 、 格式化 和 挂载 操作 。 此外 ， VDEV 还支持 对 物理 磁盘 的 加密 和 压缩 操作 。 通过 使用 VDEV 的 加密 和 压缩 接口 ， 用户 可以 对 物理 磁盘 进行 加密 和 压缩 ， 以 保护 数据 的 安全 和 节省 空间 。 这 使得 VDEV 成为 一个 非常 强大 和 灵活 的 磁盘 管理 工具 。

RAID ☐ ☐ ☐ ☐ .

VDEV 3 4 . 4 ZFS

[illegible]

VDEV을 RAID로 구성하는 방법에는 여러 가지가 있습니다. 예를 들어, RAID-Z2 VDEV을 RAID-6으로 구성하거나, RAID-Z2 VDEV을 ZFS RAID-60으로 구성할 수 있습니다. 또한, RAID-1 VDEV을 RAID-10으로 구성할 수도 있습니다. 이 문서에서는 이러한 다양한 구성 방법을 소개합니다.

VDEV ??? (VDEVs Redundancy)

1. 在 `/etc/fstab` 文件中添加以下配置：


```

# 挂载点: /mnt/usb
# 文件系统: vfat
# 挂载选项: rw, noexec, nosuid, uid=1000, gid=1000, dmask=0077, fmask=0077, x-systemd.automount
# 备份: 0
# 检查: 0
/dev/sdb1 /mnt/usb vfat rw, noexec, nosuid, uid=1000, gid=1000, dmask=0077, fmask=0077, x-systemd.automount 0 0

```

2. 创建挂载点目录：


```

sudo mkdir -p /mnt/usb

```

3. 设置权限：


```

sudo chown 1000:1000 /mnt/usb

```

4. 验证配置：


```

cat /etc/fstab

```

5. 重启系统或重新挂载：


```

sudo systemctl restart systemd

```

Stripe (1? ???)

\square $\square\square\square$ $\square\square$ VDEV $\square\square\square\square\square$ \square , $\square\square\square$ $\square\square\square$. $\square\square$ \square $\square\square$ \square $\square\square\square$

$\square\square\square$ $\square\square\square$ \square $\square\square\square$ $\square\square\square\square$. $\square\square\square\square$ $\square\square$ \square $\square\square$ VDEV $\square\square\square\square$.

ZFS 11 11 11 VDEV 11 1111 11111111 1111 1111 11 VDEV 111111 .
 1111 111111 111111 1111 1111 11 11 1111 111111 . 11 1111 11 1111
 111111 . 111111 111111 11 1 1111 111111 111111 111111 111111 111111 VDEV
 1111 111111 .

Mirrors (2? ??? ???)

[illegible]

RAID-Z1(3? ??? ???)

$2n+3(5, 7, 9 \dots)$ 的 奇数 个 数据 副本 . 每 个 数据 副本 2 个 副本 的 奇数 个 副本 副本 副本 副本 . 副本 的 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 .

VDEV ?? (Repairing VDEVs)

当 VDEV 的 副本 副本 副本 副本 , 副本 副本 副本 VDEV "degraded(副本 副本)" 副本 副本 . 副本 副本 VDEV 副本 副本 副本 副本 副本 副本 副本 . 5 副本 副本 副本 副本 副本 副本 副本 .

副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 副本 副本 . RAID-Z 副本 , 副本 副本 副本 副本 副本 .

ZFS 副本 RAID 副本 副本 副本 副本 ZFS 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 ZFS 副本 副本 副本 副本 副本 副本 . 副本 RAID 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . RAID 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 , 副本 ZFS RAID-Z 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 5 副本 ZFS 副本 副本 副本 副本 .

RAID-Z? ?? RAID ??(RAID-Z versus Traditional RAID)

RAID-Z 副本 RAID 副本 副本 副本 副本 副本 , 副本 副本 副本 ZFS 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 .

副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 RAID 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 100MB 副本 副本 副本 副本 副本 副本 副本 ! 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 RAID 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 RAID 副本 副本 副本 副本 副本 副本 .

副本 副本 副本 副本 副本 副本 ZFS 副本 副本 副本 副本 副本 , 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 ZFS 副本 副本 副本 (3) 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 . ZFS 副本 副本 副本 副本 副本 副本 副本 , 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 .

副本 RAID 副本 2 副本 副本 副本 副本 "副本 副本 (write hole)" 副本 副本 副本 . RAID 5 副本 6 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 .

副本 副本 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 RAID 副本 副本 副本 副本 副本 副本 副本 副本 . 副本 副本 副本 副本 副本 副本 副本 副本 .

数据 写入 磁盘 顺序。

ZFS 的 写 操作 是 顺序 的。它 使用 写前一致性 (Copy-on-write, COW) 来 保证 数据 的 完整性。当 需要 写入 数据 时，ZFS 会 先 将 旧 数据 复制到 新 空间，然后 再 写入 新 数据。这 样 做 的 好处 是，即使 在 写入 过程 中 发生 故障，数据 也 不会 丢失。ZFS 还 支持 快照 功能，可以 随时 创建 数据 的 副本，以便 在 需要 时 进行 恢复。此外，ZFS 还 支持 数据 压缩 和 加密 功能，可以 提高 数据 的 安全性和 存储 效率。

?? VDEV (Special VDEVs)

在 使用 ZFS 时，VDEV (Virtual Device) 是 一个 重要 的 概念。VDEV 是 一个 逻辑 设备，它 可以 由 一个 或多个 物理 设备 组成。通过 使用 VDEV，ZFS 可以 实现 数据 的 冗余 和 负载均衡。例如，你可以 将 数据 分布在 多个 硬盘 上，这样 即使 其中一个 硬盘 发生故障，数据 也 不会 丢失。此外，VDEV 还 支持 特殊 的 配置，如 镜像 和 条带化，可以 进一步提高 数据 的 可靠性和 性能。

??? ??? ??(Seperate Intent Log; SLOG, ZIL)

ZFS 使用 两个 特殊 的 VDEV 来 提高 性能 和 可靠性：SLOG (Seperate Intent Log) 和 ZIL (ZFS Intent Log)。SLOG 是 一个 专门 用于 存储 写 操作 日志 的 VDEV，它 通常 使用 高速 闪存 设备。ZIL 是 一个 专门 用于 存储 读 操作 日志 的 VDEV，它 通常 使用 硬盘 设备。通过 使用 SLOG 和 ZIL，ZFS 可以 减少 写 操作 对 主 数据 设备 的 影响，从而提高 写入 性能。同时，ZIL 还可以 保证 读 操作 的 数据 一致性。

在 配置 ZFS 时，选择合适的 SLOG 和 ZIL 设备 非常 重要。对于 SLOG，建议选择 性能 高、延迟 低 的 闪存 设备。对于 ZIL，建议选择 容量 大、可靠性 高 的 硬盘 设备。此外，还需要 注意 SLOG 和 ZIL 的 容量 规划，确保 它们 能够 满足 系统 的 需求。

ZFS 还支持 其他 一些 特殊 的 VDEV 配置，如 镜像 和 条带化。镜像 配置 可以 提高 数据 的 冗余 性，而 条带化 配置 可以 提高 数据 的 读写 性能。通过 合理 配置 这些 特殊 VDEV，可以 进一步优化 ZFS 的 性能和 可靠性。

在 使用 ZFS 时，还需要 注意 数据 的 备份 和 恢复。ZFS 提供了 丰富 的 快照 功能，可以 方便 地 创建 和 管理 数据 的 副本。此外，ZFS 还支持 数据 的 加密 和 解密 操作，可以 保护 数据 的安全。通过 合理 使用 这些 功能，可以 确保 数据 的安全性和 完整性。

ZIL 通常 使用 硬盘 作为 存储 设备。

??(Cache: L2ARC)

L2ARC (Level 2 Adaptive Replacement Cache) 是 ZFS 中 的一个 重要 功能。它 是一个 缓存 机制，用于 存储 最近 访问 过 的数据。当 需要 读取 数据 时，ZFS 会 首先 检查 L2ARC 中 是否有 该 数据。如果 有，则 直接从 L2ARC 中 读取，从而 提高 读取 性能。如果 没有，则 从 主 数据 设备 中 读取。L2ARC 还可以 用于 缓存 写 操作，从而 减少 对 主 数据 设备 的 写入 次数。

在 配置 L2ARC 时，需要 选择 合适 的 缓存 设备。通常，建议使用 高速 闪存 设备 作为 L2ARC 的 存储 介质。此外，还需要 注意 L2ARC 的 容量 规划，确保 它 能够 满足 系统 的 需求。

ZFS 还支持 其他 一些 缓存 功能，如 L1ARC (Level 1 Adaptive Replacement Cache) 和 L2S (Level 2 Secondary Cache)。L1ARC 是 一个 位于 内存 中 的 缓存，可以 进一步提高 读取 性能。L2S 是 一个 位于 硬盘 中 的 缓存，可以 进一步提高 写入 性能。通过 合理 配置 这些 缓存 功能，可以 进一步优化 ZFS 的 性能和 可靠性。

2 ARC L2ARC

RAM ZFS L2ARC SSD NVMe SSD ZFS

VDEV? ??? ?? (How VDEVs Affect Performance)

VDEV

IOPS

12 1 (12 x 1TB) 6 2 (6 x 2TB) ZFS IOPS

IOPS RAID-Z

VDEV VDEV ZFS VDEV

VDEV 1TB 250 IOPS 100MB/s

??? ?? (One Disk)

1. 1 disk, 1 VDEV, 1 ZFS pool. This configuration is suitable for low I/O workloads.

Table 1. Single Disk Virtual Device Configurations

Disks	Config	Read IOPS	Write IOPS	Read MB/s	Write MB/s	Usable Space	Fault Tolerance
1	Stripe	250	250	100	100	1 TB (100%)	none

This configuration is suitable for low I/O workloads.

2. 2 disks, 2 VDEVs, 1 ZFS pool (Two Disks)

This configuration is suitable for low I/O workloads.

This configuration is suitable for low I/O workloads.

This configuration is suitable for low I/O workloads.

Table 2: Two-Disk Virtual Device Configurations

Disks	Config	Read IOPS	Write IOPS	Read MB/s	Write MB/s	Usable Space	Fault Tolerance
2	2 x Stripe	500	500	200	200	2 TB (100%)	none
2	1 x 2 disk Mirror	500	250	200	100	1 TB (50%)	1

This configuration is suitable for low I/O workloads.

3. 3 disks, 3 VDEVs, 1 ZFS pool (Three Disks)

This configuration is suitable for low I/O workloads.

This configuration is suitable for low I/O workloads.

RAID-Z1 的 性能 比 VDEV 的 性能 低 很多 倍 。 但是 它 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

Table 3: Three-Disk Virtual Device Configurations

Disks	Config	Read IOPS	Write IOPS	Read MB/s	Write MB/s	Usable Space	Fault Tolerance
3	1 x 3 disk Mirror	750	250	300	100	1 TB (33%)	2
3	1 x 3 disk RAID-Z1	250	250	200	200	2 TB (66%)	1

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

4 or 5 Disks (Four or Five Disks)

4 or 5 disks can be configured in several ways to provide different levels of performance and fault tolerance.

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

在 需要 高 性能 的 场景 中 ， RAID-Z1 的 性能 比 RAID-Z 的 性能 高 很多 倍 。 因此 在 需要 高 性能 的 场景 中 ， RAID-Z1 是 一个 不错 的 选择 。

Table 4: Four- or Five-Disk Virtual Device Configurations

Disks	Config	Read IOPS	Write IOPS	Read MB/s	Write MB/s	Usable Space	Fault Tolerance
4	2 x 2 disk Mirror	1000	500	400	200	2 TB (50%)	2 (1/VDEV)
4	1 x 4 disk RAIDZ-Z1	250	250	300	300	3 TB (75%)	1

4	1 x 4 disk RAIDZ-Z2	250	250	200	200	2 TB (50%)	2
5	1 x 5 disk RAIDZ-Z1	250	250	400	400	4 TB (80%)	1
5	1 x 5 disk RAIDZ-Z2	250	250	300	300	3 TB (60%)	2
5	1 x 5 disk RAIDZ-Z3	250	250	200	200	2 TB (40%)	3

RAID-Z1의 읽기 속도 (MB/s)는 쓰기 속도보다 빠르며, RAID-Z2의 읽기 속도는 쓰기 속도의 절반이며, RAID-Z3의 읽기 속도는 쓰기 속도의三分之一입니다.

각각의 VDEV은 독립적으로 운영됩니다. 따라서 하나의 VDEV에 장애가 발생하면 다른 VDEV에 영향을 주지 않습니다. 각 VDEV은 n - 1개의 디스크를 포함합니다. 각 VDEV은 하나의 디스크만 포함할 수 있으며, 각 VDEV은 하나의 디스크만 포함할 수 있습니다. 각 VDEV은 하나의 디스크만 포함할 수 있습니다. 각 VDEV은 하나의 디스크만 포함할 수 있습니다.

6~12??? ??? (Six to Twelve Disks)

각각의 디스크는 하나의 VDEV에 포함될 수 있으며, 각 VDEV은 하나의 디스크만 포함할 수 있습니다.

6개의 디스크는 3개의 2개의 디스크 VDEV을 구성할 수 있으며, 각 VDEV은 하나의 디스크만 포함할 수 있습니다. 3개의 3-디스크 VDEV을 구성할 수 있으며, 각 VDEV은 하나의 디스크만 포함할 수 있습니다. 2개의 2-디스크 VDEV을 구성할 수 있으며, 각 VDEV은 하나의 디스크만 포함할 수 있습니다.

각각의 6개의 디스크는 하나의 RAID-Z VDEV을 구성할 수 있으며, 12개의 디스크는 하나의 VDEV을 구성할 수 있습니다. 각 VDEV은 하나의 디스크만 포함할 수 있습니다. 각 VDEV은 하나의 디스크만 포함할 수 있습니다.

Table 5: Six- to Twelve-Disk Virtual Device Configurations

Disks	Config	Read IOPS	Write IOPS	Read MB/s	Write MB/s	Usable Space	Fault Tolerance
6	3 x 2 disk Mirror	1500	750	600	300	3 TB (50%)	3 (1/VDEV)
6	2 x 3 disk Mirror	1500	500	600	200	2 TB (33%)	4 (2/VDEV)
6	1 x 6 disk RAIDZ-Z1	250	250	500	500	5 TB (83%)	1
6	1 x 6 disk RAIDZ-Z2	250	250	400	400	4 TB (66%)	2
6	1 x 6 disk RAIDZ-Z3	250	250	300	300	3 TB (50%)	3

12	6 x 2 disk Mirror	3000	1500	1200	600	6 TB (50%)	6 (1/VDEV)
12	4 x 3 disk Mirror	3000	1000	1200	400	4 TB (33%)	8 (2/VDEV)
12	1 x 12 disk RAIDZ-Z1	250	250	1100	1100	11 TB (92%)	1
12	2 x 6 disk RAIDZ-Z1	500	500	1000	1000	10 TB (83%)	2 (1/VDEV)
12	3 x 4 disk RAIDZ-Z1	750	750	900	900	9 TB (75%)	3 (1/VDEV)
12	1 x 12 disk RAIDZ-Z2	250	250	1000	1000	10 TB (83%)	2
12	2 x 6 disk RAIDZ-Z2	500	500	800	800	8 TB (66%)	4 (2/VDEV)
12	1 x 12 disk RAIDZ-Z3	250	250	900	900	9 TB (75%)	3
12	2 x 6 disk RAIDZ-Z3	500	500	600	600	6 TB (50%)	6 (3/VDEV)

RAID-Z 12 디스크 3000 IOPS, 1500 쓰기 IOPS, 1200 읽기 MB/s, 600 쓰기 MB/s, 6 TB (50%) 사용 가능 공간, 6 (1/VDEV) 내구성. RAID-Z 2는 12 디스크 500 IOPS, 500 쓰기 IOPS, 1000 읽기 MB/s, 1000 쓰기 MB/s, 10 TB (83%) 사용 가능 공간, 2 (1/VDEV) 내구성. RAID-Z 3은 12 디스크 250 IOPS, 250 쓰기 IOPS, 900 읽기 MB/s, 900 쓰기 MB/s, 9 TB (75%) 사용 가능 공간, 3 (1/VDEV) 내구성. RAID-Z 4는 12 디스크 125 IOPS, 125 쓰기 IOPS, 600 읽기 MB/s, 600 쓰기 MB/s, 6 TB (50%) 사용 가능 공간, 6 (3/VDEV) 내구성.

RAID-Z 2 (Many Disks)

RAID-Z 2는 12 디스크 500 IOPS, 500 쓰기 IOPS, 1000 읽기 MB/s, 1000 쓰기 MB/s, 10 TB (83%) 사용 가능 공간, 2 (1/VDEV) 내구성. RAID-Z 3은 12 디스크 250 IOPS, 250 쓰기 IOPS, 900 읽기 MB/s, 900 쓰기 MB/s, 9 TB (75%) 사용 가능 공간, 3 (1/VDEV) 내구성. RAID-Z 4는 12 디스크 125 IOPS, 125 쓰기 IOPS, 600 읽기 MB/s, 600 쓰기 MB/s, 6 TB (50%) 사용 가능 공간, 6 (3/VDEV) 내구성.

Table 5: Six- to Twelve-Disk Virtual Device Configurations

Disks	Config	Read IOPS	Write IOPS	Read MB/s	Write MB/s	Usable Space	Fault Tolerance
36	18 x 2 disk Mirror	9000	4500	3600	1800	18 TB (50%)	18 (1/VDEV)
36	12 x 3 disk Mirror	9000	3000	3600	1200	12 TB (33%)	24 (2/VDEV)
36	1 x 36 disk RAID-Z2	250	250	3400	3400	34 TB (94%)	2
36	2 x 18 disk RAID-Z2	500	500	3200	3200	32 TB (89%)	4 (2/VDEV)
36	4 x 9 disk RAID-Z2	1000	1000	2800	2800	28 TB (78%)	8 (2/VDEV)

36	6 x 6 disk RAID-Z2	1500	1500	2400	2400	24 TB (66%)	12 (2/VDEV)
----	-----------------------	------	------	------	------	-------------	-------------

Each VDEV contains 12 disks in a RAID-Z2 configuration. 18 VDEVs are used, each with 12 disks. The total number of disks is 216. The total capacity is 24 TB (66%). The total number of VDEVs is 12 (2/VDEV).

VDEVs are created using the following command: `zpool create -O raidz2 tank vdev1 vdev2 vdev3 vdev4 vdev5 vdev6 vdev7 vdev8 vdev9 vdev10 vdev11 vdev12`.

VDEVs are created using the following command: `zpool create -O raidz2 tank vdev1 vdev2 vdev3 vdev4 vdev5 vdev6 vdev7 vdev8 vdev9 vdev10 vdev11 vdev12`. The L2ARC and SLOG are also created.

VDEVs are created using the following command: `zpool create -O raidz2 tank vdev1 vdev2 vdev3 vdev4 vdev5 vdev6 vdev7 vdev8 vdev9 vdev10 vdev11 vdev12`. The L2ARC and SLOG are also created. The total number of VDEVs is 12 (2/VDEV). The total number of disks is 216. The total capacity is 24 TB (66%). The total number of VDEVs is 12 (2/VDEV).

VDEVs are created using the following command: `zpool create -O raidz2 tank vdev1 vdev2 vdev3 vdev4 vdev5 vdev6 vdev7 vdev8 vdev9 vdev10 vdev11 vdev12`.