**Ext2 v.s. Ext3 v.s. Ext4 性能比拼**

Linux kernel 自 2.6.28 开始正式支持新的文件系统 Ext4。 Ext4 是 Ext3 的改进版，修改了 Ext3 中部分重要的数据结构，而不仅仅像 Ext3 对 Ext2 那样，只是增加了一个日志功能而已。Ext4 可以提供更佳的性能和可靠性，还有更为丰富的功能：  
  
**1. 与 Ext3 兼容。**执行若干条命令，就能从 Ext3 在线迁移到 Ext4，而无须重新格式化磁盘或重新安装系统。原有 Ext3 数据结构照样保留，Ext4 作用于新数据，当然，整个文件系统因此也就获得了 Ext4 所支持的更大容量。  
  
**2. 更大的文件系统和更大的文件。**较之 Ext3 目前所支持的最大 16TB 文件系统和最大 2TB 文件，Ext4 分别支持 1EB（1,048,576TB， 1EB=1024PB， 1PB=1024TB）的文件系统，以及 16TB 的文件。  
  
**3. 无限数量的子目录。**Ext3 目前只支持 32,000 个子目录，而 Ext4 支持无限数量的子目录。  
  
**4. Extents。**Ext3 采用间接块映射，当操作大文件时，效率极其低下。比如一个 100MB 大小的文件，在 Ext3 中要建立 25,600 个数据块（每个数据块大小为 4KB）的映射表。而 Ext4 引入了现代文件系统中流行的 extents 概念，每个 extent 为一组连续的数据块，上述文件则表示为“该文件数据保存在接下来的 25,600 个数据块中”，提高了不少效率。  
  
**5. 多块分配。**当写入数据到 Ext3 文件系统中时，Ext3 的数据块分配器每次只能分配一个 4KB 的块，写一个 100MB 文件就要调用 25,600 次数据块分配器，而 Ext4 的多块分配器“multiblock allocator”（mballoc） 支持一次调用分配多个数据块。  
  
**6. 延迟分配。**Ext3 的数据块分配策略是尽快分配，而 Ext4 和其它现代文件操作系统的策略是尽可能地延迟分配，直到文件在 cache 中写完才开始分配数据块并写入磁盘，这样就能优化整个文件的数据块分配，与前两种特性搭配起来可以显著提升性能。  
  
**7. 快速 fsck。**以前执行 fsck 第一步就会很慢，因为它要检查所有的 inode，现在 Ext4 给每个组的 inode 表中都添加了一份未使用 inode 的列表，今后 fsck Ext4 文件系统就可以跳过它们而只去检查那些在用的 inode 了。  
  
**8. 日志校验。**日志是最常用的部分，也极易导致磁盘硬件故障，而从损坏的日志中恢复数据会导致更多的数据损坏。Ext4 的日志校验功能可以很方便地判断日志数据是否损坏，而且它将 Ext3 的两阶段日志机制合并成一个阶段，在增加安全性的同时提高了性能。  
  
**9. “无日志”（No Journaling）模式。**日志总归有一些开销，Ext4 允许关闭日志，以便某些有特殊需求的用户可以借此提升性能。  
  
**10. 在线碎片整理。**尽管延迟分配、多块分配和 extents 能有效减少文件系统碎片，但碎片还是不可避免会产生。Ext4 支持在线碎片整理，并将提供 e4defrag 工具进行个别文件或整个文件系统的碎片整理。  
  
**11. inode 相关特性。**Ext4 支持更大的 inode，较之 Ext3 默认的 inode 大小 128 字节，Ext4 为了在 inode 中容纳更多的扩展属性（如纳秒时间戳或 inode 版本），默认 inode 大小为 256 字节。Ext4 还支持快速扩展属性（fast extended attributes）和 inode 保留（inodes reservation）。  
  
**12. 持久预分配（Persistent preallocation）。**P2P 软件为了保证下载文件有足够的空间存放，常常会预先创建一个与所下载文件大小相同的空文件，以免未来的数小时或数天之内磁盘空间不足导致下载失败。Ext4 在文件系统层面实现了持久预分配并提供相应的 API（libc 中的 posix\_fallocate()），比应用软件自己实现更有效率。  
  
**13. 默认启用 barrier。**磁盘上配有内部缓存，以便重新调整批量数据的写操作顺序，优化写入性能，因此文件系统必须在日志数据写入磁盘之后才能写 commit 记录，若 commit 记录写入在先，而日志有可能损坏，那么就会影响数据完整性。Ext4 默认启用 barrier，只有当 barrier 之前的数据全部写入磁盘，才能写 barrier 之后的数据。（可通过 "mount -o barrier=0" 命令禁用该特性。）  
  
  
Ext4 随 Linux kernel 2.6.28 正式发布已有数周，一直苦于找不到测试用的磁盘，正巧年前 Intel 送来几块 SSD 测试样品，这两天就顺带把 SSD 也测了。测试所使用的 Linux 内核版本为 2.6.28.2，测试工具为 IOzone 3.318。  
  
[IOzone](http://www.iozone.org/) 测试命令为：

time /opt/iozone/bin/iozone -a -s 4G -q 256 -y 4 >|/root/ext4-iozone-stdout.txt

上述命令的说明如下：

Auto Mode  
File size set to 4194304 KB  
Using Maximum Record Size 256 KB  
Using Minimum Record Size 4 KB  
Command line used: /opt/iozone/bin/iozone -a -s 4G -q 256 -y 4  
Output is in Kbytes/sec  
Time Resolution = 0.000001 seconds.  
Processor cache size set to 1024 Kbytes.  
Processor cache line size set to 32 bytes.  
File stride size set to 17 \* record size.

测试结果除了表明 Intel SSD 的读写速度快得令人咋舌之外，还可以说明 Ext4 的各方面性能都超过了上一代 Ext3，甚至在大多数情况下，比没有日志功能的 Ext2 还要快出不少：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **reclen** | **write** | **rewrite** | **read** | **reread** | **random read** | **random write** | **bkwd read** | **record rewrite** | **stride read** | **fwrite** | **frewrite** | **fread** | **freread** |
| **Ext2  real 28m12.718s user 0m10.725s sys 5m8.265s** | 4 | 218,680 | 216,596 | 630,248 | 245,802 | 88,700 | 138,065 | 106,112 | 1,882,623 | 73,538 | 214,175 | 218,364 | 566,570 | 247,381 |
| 8 | 215,308 | 218,690 | 556,064 | 246,260 | 154,680 | 150,052 | 188,397 | 2,462,367 | 130,896 | 217,157 | 216,647 | 583,808 | 248,397 |
| 16 | 216,457 | 216,843 | 575,046 | 245,701 | 258,660 | 158,750 | 306,842 | 2,654,320 | 220,939 | 216,061 | 218,140 | 598,174 | 246,581 |
| 32 | 217,925 | 214,289 | 537,976 | 243,081 | 394,013 | 167,002 | 464,240 | 2,397,831 | 340,775 | 217,434 | 219,353 | 583,463 | 246,341 |
| 64 | 215,460 | 219,256 | 527,919 | 244,362 | 503,227 | 162,917 | 609,546 | 2,546,079 | 456,243 | 216,875 | 217,692 | 571,707 | 244,264 |
| 128 | 219,081 | 216,173 | 540,831 | 242,526 | 609,750 | 161,442 | 721,701 | 2,656,729 | 551,122 | 217,780 | 217,427 | 579,271 | 242,291 |
| 256 | 216,091 | 217,631 | 565,111 | 245,157 | 654,274 | 173,955 | 870,547 | 2,574,261 | 634,835 | 216,638 | 219,693 | 563,735 | 247,101 |
| **Ext3  real 27m42.449s user 0m11.529s sys 7m17.049s** | 4 | 218,242 | 213,039 | 482,132 | 243,986 | 88,007 | 156,926 | 105,557 | 1,540,739 | 75,010 | 216,028 | 216,432 | 522,704 | 243,385 |
| 8 | 218,390 | 217,915 | 544,892 | 244,979 | 152,424 | 190,454 | 181,486 | 1,945,603 | 130,737 | 218,364 | 216,431 | 530,853 | 243,222 |
| 16 | 218,083 | 217,683 | 561,038 | 244,506 | 255,244 | 200,032 | 300,212 | 2,096,495 | 221,329 | 216,930 | 216,661 | 514,177 | 244,069 |
| 32 | 216,258 | 217,013 | 569,246 | 243,811 | 389,745 | 198,275 | 446,462 | 1,934,853 | 338,785 | 216,809 | 219,296 | 530,634 | 243,446 |
| 64 | 218,850 | 217,711 | 577,529 | 243,725 | 497,689 | 201,693 | 589,535 | 2,036,412 | 450,449 | 219,387 | 214,900 | 514,353 | 244,809 |
| 128 | 220,234 | 215,687 | 530,519 | 241,615 | 608,244 | 199,619 | 714,295 | 1,992,168 | 553,022 | 217,828 | 218,454 | 513,596 | 241,510 |
| 256 | 216,011 | 220,188 | 592,578 | 242,548 | 642,341 | 199,408 | 834,240 | 2,092,959 | 624,043 | 217,682 | 218,165 | 529,358 | 242,878 |
| **Ext4  real 27m3.485s user 0m10.847s sys 6m9.578s** | 4 | 221,823 | 216,992 | 532,488 | 273,668 | 85,210 | 183,195 | 103,036 | 1,862,817 | 74,781 | 225,841 | 220,620 | 523,799 | 272,848 |
| 8 | 226,028 | 218,580 | 561,960 | 272,036 | 154,972 | 216,505 | 178,482 | 2,135,372 | 132,506 | 227,423 | 215,766 | 641,021 | 271,328 |
| 16 | 222,241 | 217,746 | 547,548 | 270,895 | 260,899 | 223,895 | 295,288 | 2,095,966 | 223,135 | 226,055 | 216,210 | 621,287 | 273,475 |
| 32 | 220,121 | 213,025 | 240,426 | 247,628 | 345,210 | 175,977 | 451,631 | 2,145,351 | 342,236 | 225,796 | 213,427 | 598,331 | 269,759 |
| 64 | 223,983 | 214,437 | 308,696 | 551,577 | 754,941 | 225,897 | 523,130 | 2,218,016 | 448,086 | 227,030 | 214,706 | 582,795 | 272,323 |
| 128 | 222,576 | 217,816 | 624,636 | 271,293 | 644,500 | 224,997 | 720,468 | 2,308,315 | 582,943 | 225,971 | 217,373 | 552,335 | 274,237 |
| 256 | 221,202 | 222,238 | 541,685 | 270,898 | 671,748 | 228,085 | 845,494 | 2,215,381 | 643,715 | 225,411 | 219,166 | 580,066 | 273,342 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | **Kbytes/sec** |

注：  
1. 关于 IOzone 测试方法，参考 Ben Martin 的文章：*[IOzone for filesystem performance benchmarking](http://www.linux.com/feature/139744)*  
2. 关于 Ext4 的相关内容，参考 Kernel Newbies 专页： <http://kernelnewbies.org/Ext4>

本文出自：<http://hutuworm.blogspot.com/2009/02/ext2-vs-ext3-vs-ext4.html>