LInux磁盘与文件系统管理

- 文件系统特性
  - filesystem
    - 每种os所设置的文件属性/权限不同，为存放这些文件所需的数据，就是分成各partition后需要进行format才能成为os可利用的文件系统
    - （每一个可被挂载的数据为一个filesystem而不是partition）
  - 机制
    - filesystem中的文件数据包含文件实际数据及其他文件属性，通常会将这两部分数据分别存放在不同的区块（如Linux filesystem的文件数据inode和文件属性owner/group/other，时间参数等记录于inode中，实际文件数据记录于block中）
  - superblock/inode/block
    - 记录此filesystem的整体信息，包括inode/block的总量、使用量、剩余量及filesystem的格式与相关信息
    - superblock
    - 记录文件的属性，一个文件占用一个inode，记录此文件实际数据所在的块号码
    - inode
    - 记录文件的实际数据（每文件大小/用途中一个或多个）
    - block
- 挂载点
  - 挂载动作
    - 将filesystem与目录树结合的动作（每个filesystem都有自己独立的inode/block/superblock等信息，此filesystem的根目录树才能被使用）
    - 挂载点必为目录
    - 该目录为进入该filesyy的入口
  - 判断不同文件名是否是相同文件
    - 同一个filesystem中的某个inode区会对应到一个文件内容
- 文件系统简单操作
  - 磁盘与目录容量
    - df
      - 功能
        - 列出文件系统的整体磁盘使用量
      - 用法
        - df [-ahkHTmi] [目录或文件名]
      - 常用选项和参数
        - -h
          - 以GBytes等格式显示
        - -i
          - 使用inode数量取代磁盘容量显示
    - du
      - 功能
        - 评估目录所占容量
      - 用法
        - du [-ahskm] 文件或目录名称
      - 常用选项和参数
        - -s
          - 仅列出总容量，而不列出每个子目录所占容量
  - 创建链接 ln
    - 链接类型
      - 实体链接（Hard Link）
        - 含义
          - 不同文件名对应同一inode号码（两文件版文件名外其他所有属性均相同）
        - 特性
          - 无论使用哪个文件名进行修改，最终结果均会反映到对应的同一inode和block中；
            通常设置hard link不会改变inode/block数量
        - 限制
          - 不能跨filesystem
          - 不能链接目录
      - 符号链接（Symbolic Link）
        - 含义
          - 类似于Windows的快捷方式，创建的是一个独立的文件
        - 特性
          - 两文件指向不同的inode号码
            符号链接文件大小 = 源文件文件名占用空间
            （等/etc/crontab共17个英文字符为17BytesBytes）
    - 用法
      - ln [-sf] 来源文件 目标文件
    - 选项与参数
      - -s
        - 设置为符号链接
        - （默认无-s则为实体链接）
      - -f
        - 若目标文件存在，则移除目标文件后再创建
- 磁盘分区 格式化 检验 挂载
  - 检查
    - lsblk
      - 功能
        - 列出系统上的所有磁盘列表
      - 用法
        - lsblk [-dfmnp] 设备
      - blkid
        - 功能
          - 列出设备的UUID（全域唯一识别码）等参数
    - MBR分区
      - fdisk
    - GPT分区
      - gdisk
    - 更新Linux核心分区表信息
      - partprobe
        - -s
          - 屏幕显示信息
  - 分区
  - 格式化 mkfs
    - mkfs.xfs
    - mkfs.ext4
    - mkfs.其他文件系统
      （mkfs是一个综合指令）
  - 检验 xfs_repair
    - fsck.ext4
  - 挂载
    - 挂载原则
      - ①根目录必须挂载，为子所有其他挂载点的根挂载
      - ②其他挂载点必须为已经建立等，最可行+原则
      - ③所有挂载点在同一时间之内，只能挂载一次
      - ④所有partition在同一时间之内，只能挂载一次
      - 系统挂载的限制
        - ⑤刻意挂到工作目录必须处于要卸载的目录及其子目录之外
    - 挂载
      - 用法
        - mount -a
          - @mount [-t 文件系统] LABEL=" 挂载点
        - mount -l
          - @mount [-t 文件系统] UUID=" 挂载点（推荐）
        - mount [-t 文件系统] 设备文件名 挂载点
      - mount
        - 常用选项与参数
          - -a
            - 依配置文件/etc/fstab的数据列所有未挂载的磁盘进行挂载
          - -l
            - 在显示目前挂载信息的基础上增列LABEL名称
        - 重新挂载根目录
          - mount -o remount,rw,auto /
        - 将某目录挂载到其他目录
          - mount --bind 源目录 目标目录
        - /etc/filesystems
          - 记录系统指定的测试挂载文件系统类型的优先顺序
        - /proc/filesystems
          - 记录Linux系统已经载入的文件系统类型
        - /lib/modules/$ (uname -r)/kernel/fs/
          - 存放Linux支持的文件系统驱动程序
    - 卸载 umount
      - 用法
        - umount [-fn] 设备文件名或挂载点
      - 选项与参数
        - -f
          - 强制卸载
        - -l
          - 立即取消文件系统（效率比-f强）
        - -n
          - 不更新/etc/mtab情况下卸载
  - 磁盘/文件系统参数修订
    - mknod
      - 功能
        - 修改主要、次要设备代码
      - 用法
        - mknod 设置文件名 [b|p] [Major] [Minor]
    - uuidgen
      - 功能
        - 生成新的UUID
    - 修改UUID或Label name
      - xfs_admin
        - 功能
          - 修改xfs文件系统的UUID和Label name
        - 用法
          - xfs_admin [-lu] [-L label] [-U uuid] 设置文件名
      - tune2fs
        - 功能
          - 修改ext的UUID和Label name
        - 用法
          - tune2fs [-l] [-L label] [-U uuid] 设置文件名
  - 设置开机挂载
    - /etc/fstab文件
      - 内容
        - [设备/UUID等] [挂载点] [文件系统] [文件系统参数] [dump] [fsck]
      - 重要内容
        - [文件系统参数]
          - async/sync
            （默认为async）
            - 设置磁盘是否以非同步方式运行
          - auto/noauto
            （自动/非自动）
            - 该filesystem是否会被mount -a主动测试挂载（默认为auto）
          - rw/ro
            （可读写/只读）
            - 该partition以可读写或只读的形态挂载
          - exec/noexec
            （可执行/不可执行）
            - 限制在此filesystem中是否可以进行"执行"操作
          - user /nouser
            （允许/不允许使用者挂载）
            - 是否允许一般用户（非root）使用mount指令挂载
          - suid/nosuid
            （有/没有SUID权限）
            - 该filesystem是否允许SUID的存在
          - defaults
            - 同时具有rw，suid，dev，exec，auto，nouser，async等参数
    - /etc/mtab和/proc/mounts
      - /etc/fstab是开机时的配置文件，而实际filesystem的挂载是记录到/etc/mtab和/proc/mounts这两个文件中的
  - 特殊设备loop挂载（mount -o loop）
    - 挂载镜像文件
      - mount -o loop 镜像文件 目标目录
      - （免除单面版光碟烦恼）
    - 创建大文件制作loop设备文件
  - swap（内存交换空间）的创建
    - 使用实体分区创建
      - ①gdisk新增分区
      - ②使用指令"mkswap 设置文件名"format此分区
      - ③使用指令"swapon 设置文件名"启动此swap设备
      - ④使用"free"或"swapon -s"观察内存用量
    - 使用文件创建
      - ①dd指令新增文件
      - ②mkswap格式化文件
      - ③swapon启动
      - ④swapoff关闭并设置自动启用
- Ext2文件系统
  - 区段组成
    - block group
      - 放置文件实际数据的区域
      - 作用
        - n个block
        - 组成
      - data block（数据区块）
        - block的大小和数量在format完成后即固定；
          每个block内最多只能存放一个文件的数据（文件大小>block大小，占用多个block；文件大小<block大小，占用此一个block，block剩余容量浪费）

Block 大小	1KB	2KB	4KB
最大单一文件限制	16GB	256GB	2TB
最大文件系统容量	2TB	8TB	16TB

        - 因block的大小造成的限制
      - inode table（inode表）
        - n个inode
        - 组成
        - inode记录文件的数据
          - 存取模式rwx
          - 拥有者与群组owner/group
          - 文件的容量
          - 创建/状态改变时间 ctime
          - 最近一次读取时间 atime
          - 最近修改时间 mtime
          - 相关时间
          - SUID/SGID等 文件特性flag
          - 文件实际数据指针pointer
        - inode特性
          - 大小
            - 固定128Bytes Ext2
            - 可变256Bytes Ext4/XFS
          - 一占一
            - 每个文件均仅占用一个inode
              （故Filesystem能创建的文件数量与inode数量有关）
            - 先找到inode比对权限，通过则转记录开始实际读取block内容
            - 文件读取顺序
            - 文件名记录在目录的block中 不记录文件名
      - 区段组成
        - pointer
          （block号码记录区域）
          （共15个）
          - 直接指向
            - 机制
            - 12个 数量
          - 间接指向
            - 机制
            - 1个间接指向
            - 1个双指间接指向
            - 1个三指间接指向
            - 数量
  - superblock（超级区块）
    - 记录存放整个filesystem相关信息 作用
    - 记录信息
      - inode/block总量
      - 未使用/已使用的inode/block总量
      - inode/block的大小
      - 时间信息
        - filesystem的mount的时间、最近一次写入数据的时间、最近一次检验磁盘的时间等filesystem的相关时间信息
      - valid bit数量
        - 已mount 0
        - 未mount 1
      - 一般为1024Bytes 大小
    - 第一个block group之后的其他block group若包含superblock，主要是作为一个block group中的superblock的备份 与block group关系
    - 记录使用/未使用的block的号码 作用 block bitmap（区块对照表）
    - 记录使用/未使用的inode号码 作用 inode bitmap（inode对照表）
    - 此区段记录每个block group的开始与结束的block号码；
      说明每个区段（superblock、bitmap、inodemap、data block）分别占了哪一个block号码之间 作用 Filesystem Description
      （文件系统描述说明）
- Ext3文件系统
  - 直接Ext家族+superblock信息 作用
  - dumpe2fs [-bh] 设备文件名 用法 dumpe2fs Ext家族指令
  - 列出保留为坏轨的部分 b 选项与参数
  - 仅列出superblock的数据，不列出其他区段数据 h
- 中介数据（metadata）
  （数据变动频繁）
- 与目录树关系
  - 记录该目录的相关权限、属性和分配到的block号码 1个inode
    - 组成 目录

Inode number	檔名
53735697	mycondz-ks.cfg
53745858	initial-setup-ks.cfg
…	…

    - block内记录了该目录下所属文件的文件名和对应的inode号码 至少1个block
    - 目录树读取
      - 通过mount的信息找到mount点的inode号码，得到根目录的inode内容，根据此inode配置信息的block内的文件名数据，在逐层往下直至读到正确的文件名，再根据文件名对应的inode找到实际数据的block并读取
- 日志式文件系统
  - 数据不一致性
    - metadata与数据存放区不一致
    - e2fsck 检查指令
  - 日志区块
    - 作用
      - 在filesystem中划出一个区块，专门在记录写入或修改文件时的步骤，可强化一致性检查
    - 机制
      - 当要写入一个文件时，先在日志记录区块中记录此文件准备要写入的信息 预备
      - 开始写入文件的权限与数据，开始更新metadata数据 实际写入
      - 完成实际数据与metadata的更新后，在日志记录区块中完成此文件的记录 结束
- XFS文件系统
  - 组成
    - superblock
      - 剩余空间管理机制
      - allocation groups（存储区群组AG）
        （与Ext家族的block group基本相同）
      - data section（数据区）
      - inode/block的分配与追踪
        - 动态配置产生（不同于Ext家族的format时固定）
        - 256Bytes——2M inode
        - 512Bytes——64K
          （Linux为存取效能限制为4K，若超过4K则kernel不允许挂载）
        - block
        - 容量可调整
    - 记录文件系统的变化，即日志区块
      （可挂载外部磁盘）
      - log section（文件系统活动登录区）
    - extent区块 realtime section（实时运行区）
      - 4K——1G
        （必须在format时固定）
        - 大小
      - 当有文件要被创建时，xfs会在此区段中找到至少一个extent区块，将文件放置于这些区块内；待分配完毕后，再写入到data section的inode和block中去 功能
    - xfs_info 挂载点 设备文件名 用法
    - 列出指定xfs文件系统的描述数据 功能 xfs_info 指令

试用版 XMind:ZEN 试用版