

1.分区表的备份与恢复

解决方案:

1、复制分区表信息

```
1 [root@exercise1 ~]# fdisk -l >> /opt/sdb.txt
```

恢复：使用fdisk 手动按照备份内容重新分区

2、dd工具直接备份分区，再恢复

```
1 [root@exercise1 ~]# dd if=/dev/sdb of=/opt/sdb.txt  
bs=512 count=1 #备份分区表
```

```
[root@exercise1 ~]# dd if=/dev/zero of=/dev/sdb bs=512 count=1  
#损坏分区表
```

```
[root@exercise1 ~]# fdisk -l
```

磁盘 /dev/sdb: 21.5 GB, 21474836480 字节, 41943040 个扇区

Units = 扇区 of 1 * 512 = 512 bytes

扇区大小(逻辑/物理): 512 字节 / 512 字节

I/O 大小(最小/最佳): 512 字节 / 512 字节

```
[root@exercise1 ~]# dd if=/opt/sdb.txt of=/dev/sdb bs=512 count=1  
#恢复分区表
```

```
[root@exercise1 ~]# fdisk -l
```

磁盘 /dev/sdb: 21.5 GB, 21474836480 字节, 41943040 个扇区

Units = 扇区 of 1 * 512 = 512 bytes

扇区大小(逻辑/物理): 512 字节 / 512 字节

I/O 大小(最小/最佳): 512 字节 / 512 字节

磁盘标签类型: dos

磁盘标识符: 0x32b4b72b

设备	Boot	Start	End	Blocks	Id	System
/dev/sdb1		2048	2099199	1048576	83	Linux

3.xfs 文件系统的备份和恢复

XFS 提供了 **xfsdump** 和 **xfsrestore** 工具协助备份 XFS 文件系统中的数据。
xfsdump 按 inode 序号备份一个 XFS 文件系统。

centos7 选择 xfs 格式作为默认文件系统，而且不再使用以前的 ext，仍然支持 ext4，xfs 为大数据而生，每个单个文件系统最大可以支持 8eb，单个文件可以支持 16tb，不仅数据量大，而且扩展性高。还可以通过 **xfsdump**，**xfsrestore** 来备份和恢复。

不传统的 UNIX 文件系统不同，XFS 不需要在备份前被卸载；对使用中的 XFS 文件系统做备份就可以保证镜像的一致性。

XFS 的备份和恢复的过程是可以被中断然后继续的，无须冻结文件系统。

xfsdump 甚至提供了高性能的多线程备份操作——它把一次 dump 拆分成多个数据流，**每个数据流可以被发往不同的目的地**

首先了解一下 **xfsdump** 的备份级别有以下两种，默认为 0（即完全备份）

0 级别代表：完全备份

1 到 9 级别代表：增量备份

扩展：

完全备份：每次都把指定的备份目录完整的复制一遍，不管目录下的文件有没有变化；

增量备份：每次将之前（第一次、第二次、直到前一次）做过备份之后有变化的文件进行备份；

差异备份：每次都较上次备份有变化的文件进行备份。

完全备份>差异备份>增量备份

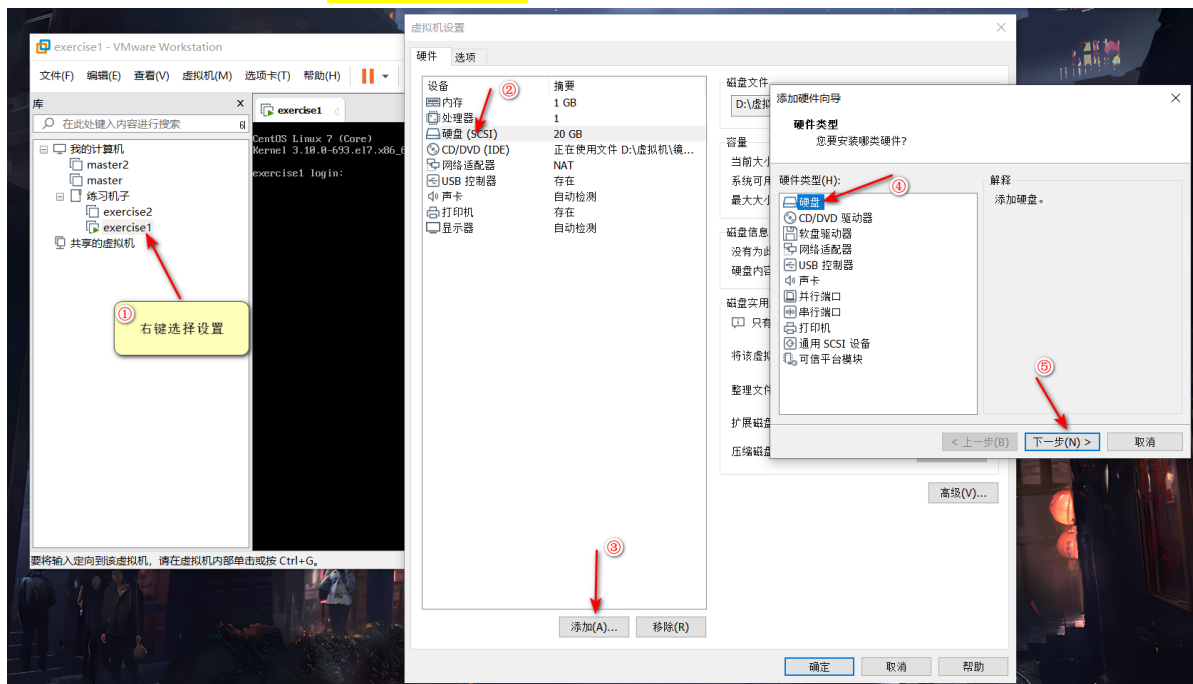
2. 环境准备

实验环境：

添加一块虚拟硬盘（准备一个测试分区）

正常使用一块磁盘过程如下：

添加磁盘大小：20G>分区格式化>挂载



对新添加的硬盘进行格式化：

```
1 [root@exercise1 ~]# fdisk /dev/sdb
2 欢迎使用 fdisk (util-linux 2.23.2)。
3
4 更改将停留在内存中，直到您决定将更改写入磁盘。
5 使用写入命令前请三思。
6
7 Device does not contain a recognized partition table
8 使用磁盘标识符 0xfa91d0bb 创建新的 DOS 磁盘标签。
9
10 命令(输入 m 获取帮助): n    #创建一个新的分区
11 Partition type:
12     p   primary (0 primary, 0 extended, 4 free)
13     e   extended
14 select (default p): p    #创建一个主分区
15 分区号 (1-4, 默认 1):
16 起始 扇区 (2048-41943039, 默认为 2048):
17 将使用默认值 2048
```

```

18 Last 扇区, +扇区 or +size{K,M,G} (2048-41943039, 默认为
   41943039): +1G    #指定分区大小
19 分区 1 已设置为 Linux 类型, 大小设为 1 GiB
20
21 命令(输入 m 获取帮助): p    #打印分区表
22
23 磁盘 /dev/sdb: 21.5 GB, 21474836480 字节, 41943040 个扇区
24 Units = 扇区 of 1 * 512 = 512 bytes
25 扇区大小(逻辑/物理): 512 字节 / 512 字节
26 I/O 大小(最小/最佳): 512 字节 / 512 字节
27 磁盘标签类型: dos
28 磁盘标识符: 0xfa91d0bb
29
30      设备 Boot          Start          End          Blocks      Id
   System
31 /dev/sdb1              2048          2099199      1048576      83
   Linux
32
33 命令(输入 m 获取帮助): w    #保存
34 The partition table has been altered!
35
36 Calling ioctl() to re-read partition table.
37 正在同步磁盘。

```

```

[root@exercise1 ~]# ls /dev/sdb*  #查看是否有分区
/dev/sdb /dev/sdb1

```

```

[root@exercise1 ~]#

```

使用新的分区，格式化分区，并进行挂载

```
1 [root@exercise1 ~]# mkfs.xfs /dev/sdb1
2 meta-data=/dev/sdb1          isize=512
  agcount=4, agsize=65536 blks
3          =                   sectsz=512   attr=2,
  projid32bit=1
4          =                   crc=1       finobt=0,
  sparse=0
5 data      =                   bsize=4096
  blocks=262144, imaxpct=25
6          =                   sunit=0     swidth=0
  blks
7 naming    =version 2         bsize=4096   ascii-
  ci=0 ftype=1
8 log       =internal log      bsize=4096
  blocks=2560, version=2
9          =                   sectsz=512   sunit=0
  blks, lazy-count=1
10 realtime =none              extsz=4096   blocks=0,
  rtextents=0
```

[root@exercise1 ~]# mkdir /opt/test #创建挂载点

[root@exercise1 ~]# mount /dev/sdb1 /opt/test/ #挂载

[root@exercise1 ~]#

准备备份测试文件

```
1 [root@exercise1 ~]# cd /opt/test/
```

[root@exercise1 test]# cp /etc/passwd ./

[root@exercise1 test]# mkdir abc

[root@exercise1 test]# touch ./abc/a.txt

3.备份

1、备份整个分区。（这个功能就像是虚拟机的快照，服务器被黑后，进行快速恢复）

语法：**xfsdump -f 备份存放位置 要备份路径或设备文件**

注意：备份的路径这里不能写成/opt/test/。 可以是/opt/test

```
1 [root@exercise1 test]# yum -y install xfsdump
```

```
[root@exercise1 test]# xfsdump -f /opt/dump_opt_test /opt/test
xfsdump: using file dump (drive_simple) strategy
xfsdump: version 3.1.7 (dump format 3.0) - type ^C for status and
control
```

```
1  ===== dump label dialog
   =====
2
3  please enter label for this dump session (timeout in
   300 sec)
4  -> 第一次完备
5  session label entered: "第一次完备"
6
7  ----- end dialog -----
   -----
8
9  xfsdump: level 0 dump of exercise1:/opt/test
10 xfsdump: dump date: Fri Feb  4 19:45:03 2022
11 xfsdump: session id: 11d1db37-a53c-4325-b5cd-
   6c4b0d13365e
12 xfsdump: session label: "第一次完备"
```

```
13 xfsdump: ino map phase 1: constructing initial dump
    list
14 xfsdump: ino map phase 2: skipping (no pruning
    necessary)
15 xfsdump: ino map phase 3: skipping (only one dump
    stream)
16 xfsdump: ino map construction complete
17 xfsdump: estimated dump size: 25856 bytes
18 xfsdump: /var/lib/xfsdump/inventory created
19
20 ===== media label dialog
    =====
21
22 please enter label for media in drive 0 (timeout in 300
    sec)
23   -> 测试目录
24 media label entered: "测试目录"
25
26 ----- end dialog -----
    -----
27
28 xfsdump: creating dump session media file 0 (media 0,
    file 0)
29 xfsdump: dumping ino map
30 xfsdump: dumping directories
31 xfsdump: dumping non-directory files
32 xfsdump: ending media file
33 xfsdump: media file size 24016 bytes
34 xfsdump: dump size (non-dir files) : 1056 bytes
35 xfsdump: dump complete: 47 seconds elapsed
36 xfsdump: Dump Summary:
37 xfsdump:   stream 0 /opt/dump_opt_test OK (success)
38 xfsdump: Dump Status: SUCCESS
```

那个是要保存的名字，不是要先mkdir /opt/dump_opt_test

2、指定备份时免交互操作，方便后期做定时备份

```
1 [root@exercise1 test]# xfsdump -f /opt/dump_passwd  
/opt/test -L 第二次完备 -M 测试目录二
```

xfsdump: using file dump (drive_simple) strategy
xfsdump: version 3.1.7 (dump format 3.0) - type ^C for status and control

xfsdump: level 0 dump of exercise1:/opt/test
xfsdump: dump date: Fri Feb 4 19:47:40 2022
xfsdump: session id: 1e828012-1415-4f98-93bc-ff5c43559351
xfsdump: session label: "第二次完备"
xfsdump: ino map phase 1: constructing initial dump list
xfsdump: ino map phase 2: skipping (no pruning necessary)
xfsdump: ino map phase 3: skipping (only one dump stream)
xfsdump: ino map construction complete
xfsdump: estimated dump size: 25856 bytes
xfsdump: creating dump session media file 0 (media 0, file 0)
xfsdump: dumping ino map
xfsdump: dumping directories
xfsdump: dumping non-directory files
xfsdump: ending media file
xfsdump: media file size 24016 bytes
xfsdump: dump size (non-dir files) : 1056 bytes
xfsdump: dump complete: 10 seconds elapsed
xfsdump: Dump Summary:
xfsdump: stream 0 /opt/dump_passwd OK (success)
xfsdump: Dump Status: SUCCESS

**[root@exercise1 test]# xfsdump -f /opt/dump_passwd /opt/test -L
第二次完备(对应: /opt/dump_passwd) -M 测试目录二(对
应: /opt/test)**

-L : xfsdump 记录每次备份的 session 标签, 这里可以填写针对此备份文件的简易说明

-M : xfsdump 可以记录储存媒体的标头, 设备标签的简易说明

3、指定只备份分区中某个目录

参数: **-s 文件路径** 只对指定的文件进行备份, **-s 指定时, 路径写的是相对路径(-s 可以是文件或目录)**

```
1 [root@exercise1 test]# xfsdump -f /opt/dump_grub2 /boot  
-s grub2/grub.cfg -L dump_grup2 -M boot-sda1
```

```
xfsdump: using file dump (drive_simple) strategy  
xfsdump: version 3.1.7 (dump format 3.0) - type ^C for status and  
control  
xfsdump: level 0 dump of exercise1:/boot  
xfsdump: dump date: Fri Feb  4 19:52:29 2022  
xfsdump: session id: 402d3c6a-5dc3-4974-a008-9e9166221ef0  
xfsdump: session label: "dump_grup2"  
xfsdump: ino map phase 1: constructing initial dump list  
xfsdump: ino map phase 2: skipping (no pruning necessary)  
xfsdump: ino map phase 3: skipping (only one dump stream)  
xfsdump: ino map construction complete  
xfsdump: estimated dump size: 29632 bytes  
xfsdump: creating dump session media file 0 (media 0, file 0)  
xfsdump: dumping ino map  
xfsdump: dumping directories  
xfsdump: dumping non-directory files  
xfsdump: ending media file  
xfsdump: media file size 27976 bytes  
xfsdump: dump size (non-dir files) : 4640 bytes  
xfsdump: dump complete: 10 seconds elapsed  
xfsdump: Dump Summary:  
xfsdump:  stream 0 /opt/dump_grub2 OK (success)  
xfsdump: Dump Status: SUCCESS
```

4、查看备份信息与内容

备份成功后, 我们就可以在 **/var/lib/xfsdump/inventory** 目录下看到生成的档案信息

```
1 [root@exercise1 test]# xfsdump -I #大写字母I
2 file system 0:
3     fs id:          e0c51d06-cb7c-4e3a-8f18-0bf4803a8b28
4     session 0:
5         mount point:    exercise1:/opt/test
6         device:         exercise1:/dev/sdb1
7         time:           Fri Feb  4 19:45:03 2022
8         session label:  "第一次完备"
9         session id: 11d1db37-a53c-4325-b5cd-
10        6c4b0d13365e
11         level:         0
12         resumed:       NO
13         subtree:       NO
14         streams:       1
15         stream 0:
16             pathname:   /opt/dump_opt_test
17             start:       ino 67 offset 0
18             end:         ino 70 offset 0
19             interrupted: NO
20             media files: 1
21             media file 0:
22                 mfile index: 0
23                 mfile type: data
24                 mfile size: 24016
25                 mfile start: ino 67 offset 0
26                 mfile end: ino 70 offset 0
27                 media label: "测试目录"
28                 media id: e64900bd-eca6-418a-a7e6-
29        4310efdb0fcd
30     session 1:
31         mount point:    exercise1:/opt/test
32         device:         exercise1:/dev/sdb1
33         time:           Fri Feb  4 19:47:40 2022
34         session label:  "第二次完备"
35         session id: 1e828012-1415-4f98-93bc-
36        ff5c43559351
37         level:         0
38         resumed:       NO
39         subtree:       NO
40         streams:       1
```

```

38         stream 0:
39             pathname:    /opt/dump_passwd
40             start:       ino 67 offset 0
41             end:         ino 70 offset 0
42             interrupted: NO
43             media files: 1
44             media file 0:
45                 mfile index:    0
46                 mfile type: data
47                 mfile size: 24016
48                 mfile start:    ino 67 offset 0
49                 mfile end:     ino 70 offset 0
50                 media label:    "测试目录二"
51                 media id:      bab0658a-b310-431d-9d0d-
52 92a8f0fc56c1
53 file system 1:
54     fs id:      7d95eb01-b37e-4430-b021-a0fdaa96e782
55     session 0:
56         mount point:    exercise1:/boot
57         device:         exercise1:/dev/sda1
58         time:           Fri Feb  4 19:52:29 2022
59         session label:  "dump_grup2"
60         session id: 402d3c6a-5dc3-4974-a008-
61 9e9166221ef0
62         level:         0
63         resumed:       NO
64         subtree:       YES
65         streams:       1
66         stream 0:
67             pathname:    /opt/dump_grub2
68             start:       ino 393285 offset 0
69             end:         ino 393286 offset 0
70             interrupted: NO
71             media files: 1
72             media file 0:
73                 mfile index:    0
74                 mfile type: data
75                 mfile size: 27976
76                 mfile start:    ino 393285 offset 0
77                 mfile end:     ino 393286 offset 0

```

```
76          media label:    "boot-sda1"
77          media id:      9fc025c4-b23e-4ed0-8ff9-
dcbf98e2ae93
78 xfsdump: Dump Status: SUCCESS
79 [root@exercise1 test]#
```

测试恢复：先删除之前创建的内容

```
1 [root@exercise1 test]# ll
2 总用量 4
3 drwxr-xr-x. 2 root root 19 2月  4 19:43 abc
4 -rw-r--r--. 1 root root 933 2月  4 19:43 passwd
5 [root@exercise1 test]# pwd
6 /opt/test
7 [root@exercise1 test]# rm -rf ./*
8 [root@exercise1 test]#
```

4.文件系统恢复

语法: **xfsrestore -f** 指定备份文件的位置 指定存放恢复后的文件的路径

```
1 [root@exercise1 test]# xfsrestore -f /opt/dump_opt_test
/opt/test
```

查看恢复情况

```
1 [root@exercise1 test]# ls /opt/
```

恢复单个文件如下

```
1 [root@exercise1 test]# mkdir /var/test
2 [root@exercise1 test]# xfsrestore -f /opt/dump_grub2 -s
grub2/grub.cfg /var/test/
3 [root@exercise1 test]# xfsrestore -f /opt/dump_grub2 -s
grub2 /var/test/    #恢复目录
```

注：

使用 xfsdump 时，请注意下面下面的几个限制：

- 1、xfsdump 不支持没有挂载的文件系统备份！所以只能备份已挂载的！
- 2、xfsdump 必须使用 root 的权限才能操作 (涉及文件系统的关系)
- 3、xfsdump 只能备份 XFS 文件系统
- 4、xfsdump备份下来的数据（档案或储存媒体）只能让 xfsrestore解析
- 5、xfsdump是透过文件系统的UUID来分辨各个备份档的，因些不能备份两个具有相同UUID的文件系统

增量备份

概念

增量备份是指在一次全备份或上一次增量备份后，以后每次的备份只需备份不前一次相比增加或者被修改的文件。

这就意味着，第一次增量备份的对象是进行全备后所产生的增加和修改的文件；第二次增量备份的对象是进行第一次增量备份后所产生的增加和修改的文件，以此类推。

优缺点

优点：没有重复的备份数据，因此备份的数据量不大，备份所需的时间很短。

缺点：数据恢复相对比较麻烦，它需要上一次全备份和所有增量备份的内容才能够完全恢复成功，并且它们必须沿着从 全备份到依次增量备份的时间顺序逐个反推恢复，因此可能会延长恢复时间

实战：增量备份文件系统

1.准备一个备份目录进行备份

```
1 [root@exercise1 opt]# touch ./test/file-{1..9}
2 [root@exercise1 opt]# tree /opt/test/
3 /opt/test/
4 |— file-1
5 |— file-2
6 |— file-3
7 |— file-4
8 |— file-5
```

```
9 | └─ file-6
10 | └─ file-7
11 | └─ file-8
12 | └─ file-9
13 |
14 | 0 directories, 9 files
15 | [root@exercise1 opt]#
```

2.对上面的内容进行第一次全备

```
1 | [root@exercise1 opt]# xfsdump -f /opt/dump-full
  | /opt/test -L dump-full -M /opt/test/
```

3.增加一些内容，然后进行第 1 次增量备份

```
1 | [root@exercise1 opt]# touch /opt/test/1.txt
  | /opt/test/2.txt
```

```
[root@exercise1 opt]# tree /opt/test/
```

```
/opt/test/
```

```
├─ 1.txt
├─ 2.txt
├─ file-1
├─ file-2
├─ file-3
├─ file-4
├─ file-5
├─ file-6
├─ file-7
├─ file-8
└─ file-9
```

```
0 directories, 11 files
```

```
1 | [root@exercise1 opt]# xfsdump -l 1 -f /opt/test-bak1
  | /opt/test -L test-bak1 -M /opt/test/
```

-l 做一个等级为 1 的备份

4.再次增加内容，然后进行 level 2 级别的增量备份

```
1 [root@exercise1 opt]# touch /opt/test/a.txt
  /opt/test/b.txt
2 [root@exercise1 opt]# tree /opt/test
3 /opt/test
4 |— 1.txt
5 |— 2.txt
6 |— a.txt
7 |— b.txt
8 |— file-1
9 |— file-2
10 |— file-3
11 |— file-4
12 |— file-5
13 |— file-6
14 |— file-7
15 |— file-8
16 |— file-9
17
18 0 directories, 13 files
```

```
[root@exercise1 opt]# xfsdump -l 2 -f /opt/test-bak2 /opt/test -L
test-bal2 -M /opt/test
```

```
1 [root@exercise1 opt]# rm -rf /opt/test/*    #删除所有数据
2 [root@exercise1 opt]# tree /opt/test
3 /opt/test
4
5 0 directories, 0 files
```

现在进行恢复，要想恢复全部全部数据，包括新添加的文件，如何恢复？

步骤：

1、先恢复完全备份

2、情况 1: 恢复最后一次增量备份（如果两次增量备份都是 1 级的，所以只需要恢复最后一个增量就可以了。）(也叫差异备份恢复)

3、情况 2：如果你做的是第一次是 1 级备，第二次是 2 级备，那么你在恢复的时候就需要先恢复完全备份，然后是 1 级备，最后是 2 级备)

可以理解成当时是顺着去的，现在就倒着来恢复

```
1 [root@exercise1 opt]# xfsrestore -f /opt/dump-full  
/opt/test
```

[root@exercise1 opt]# xfsrestore -f /opt/test-bak2 /opt/test #故意先恢复 bak2，查看

[root@exercise1 opt]# tree /opt/test #查看一下，发现没有1.txt和2.txt

```
/opt/test  
├── a.txt  
├── b.txt  
├── file-1  
├── file-2  
├── file-3  
├── file-4  
├── file-5  
├── file-6  
├── file-7  
├── file-8  
└── file-9
```

0 directories, 11 files

[root@exercise1 opt]# xfsrestore -f /opt/test-bak1 /opt/test


```
[root@exercise1 opt]# tree /opt/test
```

```
/opt/test
├── 1.txt
├── 2.txt
├── a.txt
├── b.txt
├── file-1
├── file-2
├── file-3
├── file-4
├── file-5
├── file-6
├── file-7
├── file-8
└── file-9
```

0 directories, 13 files

#到此，数据恢复成功了

当做了一次全量备份之后，再多次的增量备份时,不改变增量level值，实现差异备份

```
1 [root@exercise1 opt]# xfsdump -f /opt/dump-full2
/opt/test -L dump-full -M /opt/test
```

```
[root@exercise1 opt]# xfsdump -l 1 -f /opt/test-back1 /opt/test -L
test-back1 -M /opt/test
```

```
[root@exercise1 opt]# xfsdump -l 1 -f /opt/test-back1-2 /opt/test -L
test-back1-2 -M /opt/test
```

#-l1就是对同一级别的增量的差异备份

```
[root@exercise1 opt]# xfsdump -l 0 -f /opt/test-back1-2 /opt/test -L  
test-back1-2 -M /opt/test
```

#-l0就是对完备的差异备份

```
[root@exercise1 opt]# cd  
[root@exercise1 ~]# rm -rf /opt/test/* #删除所有数据  
[root@exercise1 ~]# ls /opt/test/  
[root@exercise1 ~]# xfsrestore -f /opt/dump-full2 /opt/test #先恢复  
完备  
[root@exercise1 ~]# xfsrestore -f /opt/test-back1-2 /opt/test #再恢  
复差异备份，就能实现数据全部恢复
```

5.会话管理工具

xfsinvutil - xfsdump 库存数据库检查和修剪实用程序

xfsinvutil [-F|-i] [-m media_label] -M 挂载点 mm/dd/yyyy

xfsinvutil [-F|-i] [-m media_label] -u UUID mm/dd/yyyy

xfsinvutil -i

xfsinvutil -C

- | | |
|---|--|
| 1 | -F |
| 2 | 不要提示操作员。当xfsinvutil遇到匹配的转储会话时，xfsinvutil通常会请求操作员允许删除该条目。 |
| 3 | 使用此选项，将执行删除，而不会提出任何问题 |

-i

交互模式。使xfsinvutil以允许操作员浏览和修改清单内容的模式运行。
有关详细信息，
请参阅下面的交互模式部分。

-M mount_point 毫米/日/年

修剪由指定挂载点标识的在指定日期之前创建的转储会话。可以选择指定-m以进一步限制

按媒体标签匹配的转储会话。xfsinvutil将在修剪转储会话之前提示操作员，除非给出了-F或-i选项。

-u UUID mm/dd/yyyy

像-M一样，除了匹配的文件系统是使用其通用唯一标识符（UUID）而不是其挂载点指定的。

-m media_label

如果指定，则只有至少具有一个标签与此值匹配的媒体文件的会话才有资格进行修剪。

-C

使用此选项，xfsinvutil将对清单数据库中的所有条目执行一致性检查。它可以修复发现的任何问题。如果未找到一致的条目，则删除相应的清单数据库文件。

交互模式

当使用-i运行时，xfsinvutil将向操作员提供xfsdump清单的分层表示形式。在此模式下，

操作员可以使用箭头键（或h j k l键）来导航清单，还可以使用以下键和功能：

- | | |
|----|-----------------------|
| 1 | + |
| 2 | 展开树的分支。 |
| 3 | |
| 4 | - |
| 5 | 折叠树的树枝。 |
| 6 | |
| 7 | * |
| 8 | 完全展开树的分支。 |
| 9 | |
| 10 | % |
| 11 | 完全折叠树的树枝。 |
| 12 | |
| 13 | d |
| 14 | 标记要删除的当前记录 and 所有子记录。 |
| 15 | |

16	u
17	清除当前记录 and 所有父记录，使其不被删除。
18	
19	i
20	导入另一个xfs 转储库存。
21	
22	x
23	提交对清单所做的任何更改，然后退出程序。
24	
25	q
26	退出程序，而不对清单进行任何更改。

Linux备份策略（完全备份、增量备份和差异备份）详解

完全备份

完全备份是指把所有需要备份的数据全部备份。当然，完全备份可以备份整块硬盘、整个分区或某个具体的目录。对于 Linux 操作系统来说，完全备份指的就是将根目录下的所有文件进行备份。

完全备份的好处是，所有数据都进行了备份，系统中任何数据丢失都能恢复，且恢复效率较高。如果完全备份备份的是整块硬盘，那么甚至不需要数据恢复，只要把备份硬盘安装上，服务器就会恢复正常。

完全备份的缺点也很明显，那就是需要备份的数据量较大，备份时间较长，备份了很多无用数据，占用的空间较大，所以完全备份不可能每天执行。

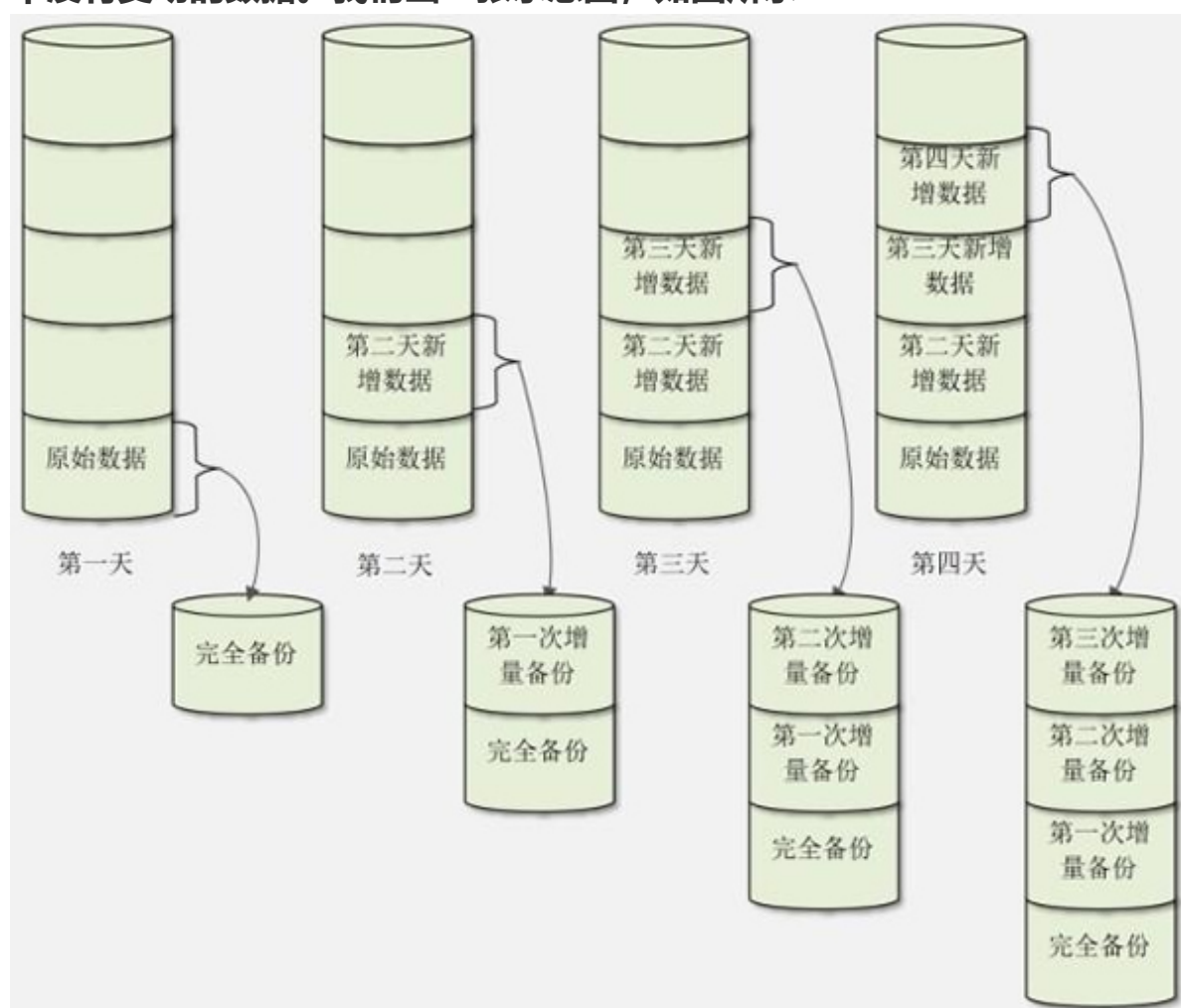
我们一般会对关键服务器进行整盘完全备份，如果出现问题，则可以很快地使用备份硬盘进行替换，从而减少损失。我们甚至会对关键服务器搭设一台一模一样的服务器，这样只要远程几个命令（或使用 Shell 脚本自动检测，自动进行服务器替换），备份服务器就会接替原本的服务器，使故障响应时间大大缩短。

累计增量备份

在一个数据量很大的业务应用中，每天对 Linux 系统进行完全备份是不现实的，这就需要用到增量备份策略。

累计增量备份是指先进行一次完全备份，服务器运行一段时间之后，比较当前系统和完全备份的备份数据之间的差异，只备份有差异的数据。服务器继续运行，再经过一段时间，进行第二次增量备份。在进行第二次增量备份时，当前系统和第一次增量备份的数据进行比较，也是只备份有差异的数据。第三次增量备份是和第二次增量备份的数据进行比较，以此类推。

因此，累计增量备份就是只备份每天增加或者变化的数据，而不备份系统中没有变动的数据。我们画一张示意图，如图所示



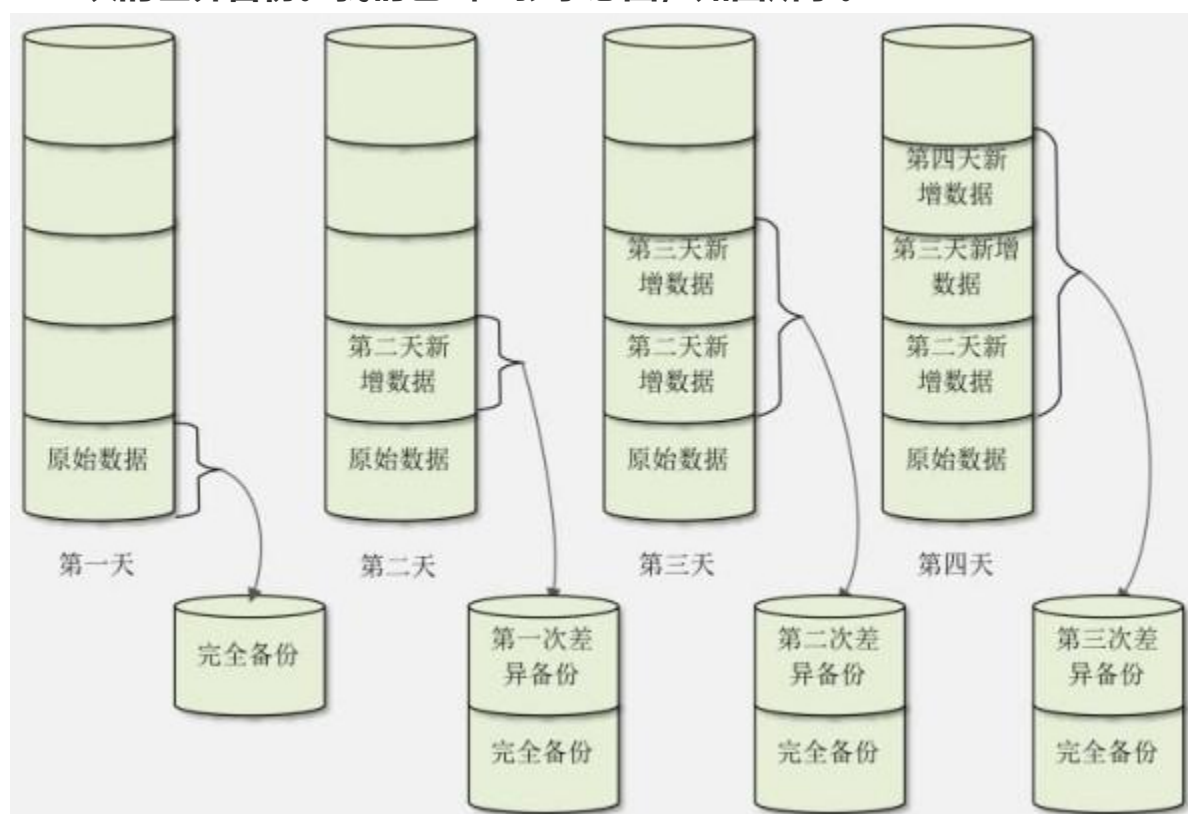
假设我们在第一天进行一次完全备份。第二天增量备份时，只会备份第二天和第一天之间的差异数据，但是第二天的总备份数据是完全备份加第一次增量备份的数据。第三天增量备份时，只会备份第三天和第二天之间的差异数据，但是第三天的总备份数据是完全备份加第一次增量备份的数据，再加第二次增量备份的数据。当然，第四天增量备份时，只会备份第四天和第三天的差异数据，但是第四天的总备份数据是完全备份加第一次

增量备份的数据，加第二次增量备份的数据，再加第三次增量备份的数据。

采用累计增量备份的好处是，每次备份需要备份的数据较少，耗时较短，占用的空间较小；坏处是数据恢复比较麻烦，如果是图 1 的例子，那么当进行数据恢复时，就要先恢复完全备份的数据，再依次恢复第一次增量备份的数据、第二次增量备份的数据和第三次增量备份的数据，最终才能恢复所有的数据。

差异增量备份

差异增量备份（后续简称差异备份）也要先进行一次完全备份，但是和累计增量备份不同的是，每次差异备份都备份和原始的完全备份不同的数据。也就是说，差异备份每次备份的参照物都是原始的完全备份，而不是上一次的差异备份。我们也画一张示意图，如图所示。



假设我们在第一天也进行一次完全备份。第二天差异备份时，会备份第二天和第一天之间的差异数据，而第二天的备份数据是完全备份加第一次差异备份的数据。第三天进行差异备份时，仍和第一天的原始数据进行对比，把第二天和第三天所有的数据都备份在第二次差异备份中，第三天的备份数据是完全备份加第二次差异备份的数据。第四天进行差异备份时，仍和第一天的原始数据进行对比，把第二天、第三天和第四天所有的不同

数据都备份到第三次差异备份中，第四天的备份数据是完全备份加第三次差异备份的数据。

相比较而言，差异备份既不像完全备份一样把所有数据都进行备份，也不像增量备份在进行数据恢复时那么麻烦，只要先恢复完全备份的数据，再恢复差异备份的数据即可。不过，随着时间的增加，和完全备份相比，变动的数据越来越多，那么差异备份也可能会变得数据量庞大、备份速度缓慢、占用空间较大。

一个比较的备份策略是，对于数据量不大，并且每天数据量增加不多的系统，优先选择完全备份；对于数据量巨大，每天新增数据也很多的系统，视情况选择差异备份或者增量备份。

