

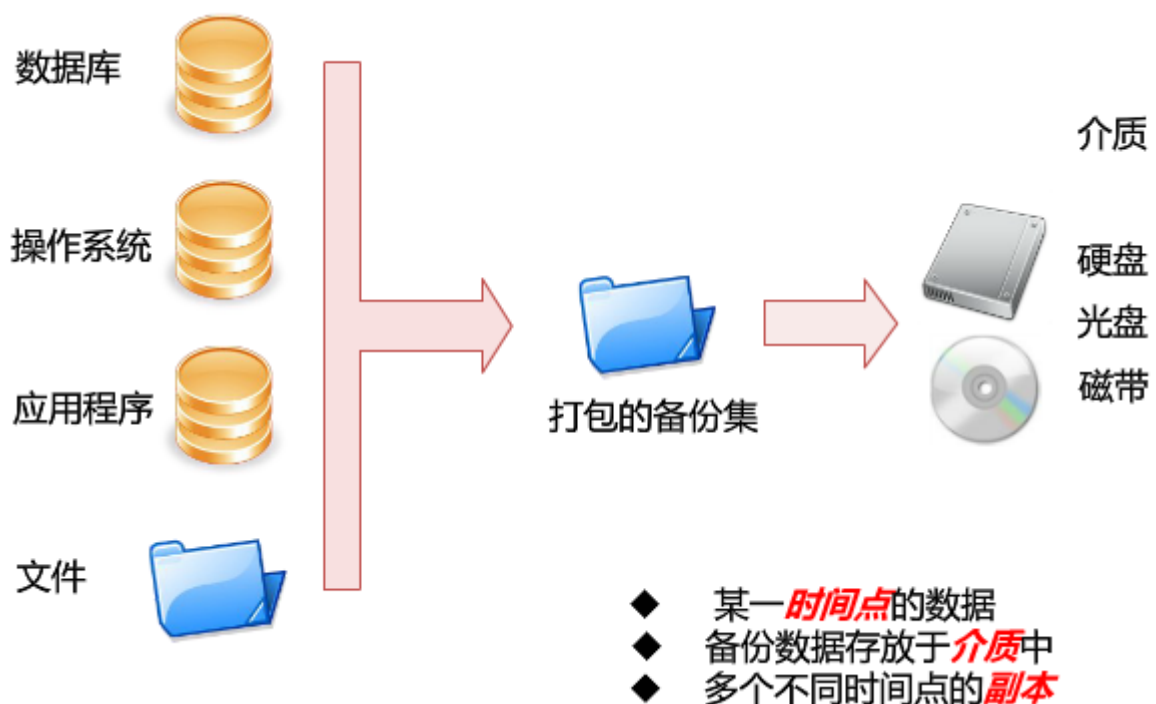
linux下如何实现mysql数据库每天自动备份定时备份_微wx笑的博客-CSDN博客_mysql数据库定时备份

blog.csdn.net/testcs_dn/article/details/48829785

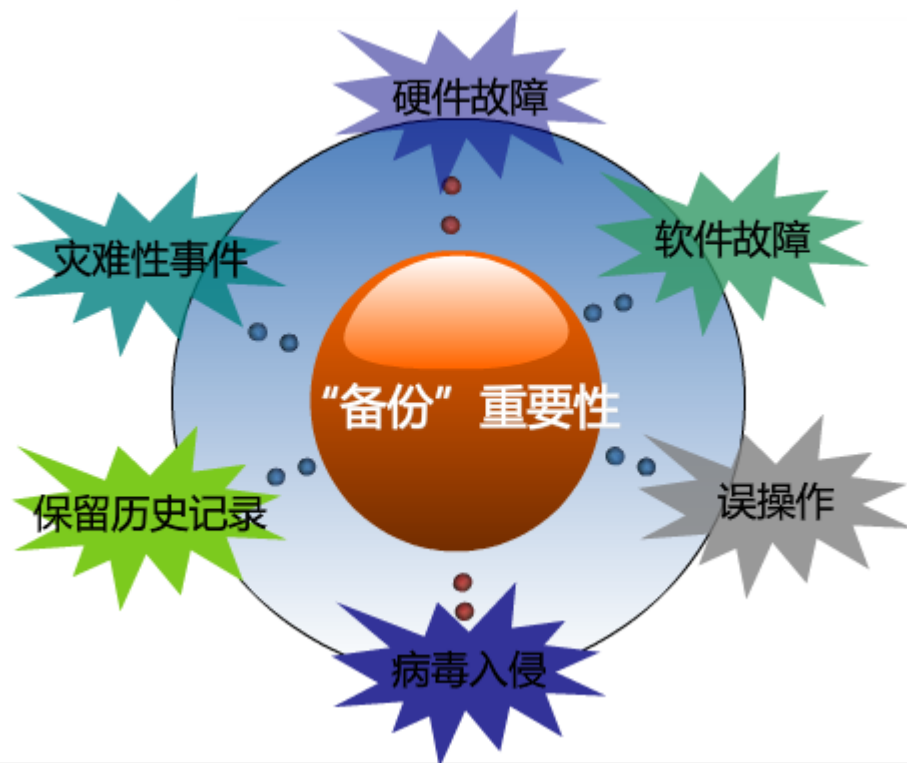
概述

备份是容灾的基础，是指为防止系统出现操作失误或系统故障导致数据丢失，而将全部或部分数据集合从应用主机的硬盘或阵列复制到其它的存储介质的过程。而对于一些网站、系统来说，数据库就是一切，所以做好数据库的备份是至关重要的！

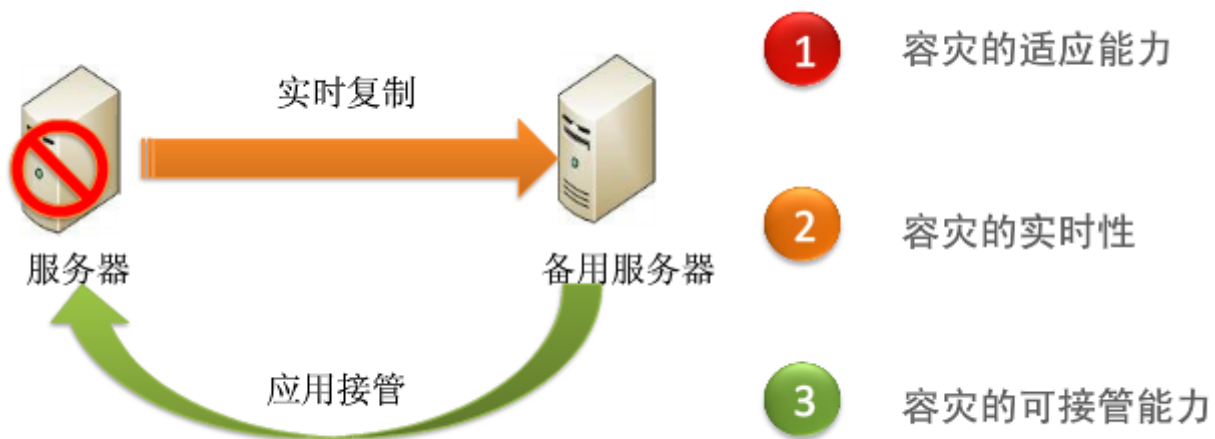
备份是什么？



为什么要备份



容灾方案建设



存储介质

光盘

磁带

硬盘

磁盘阵列

DAS：直接附加存储

NAS：网络附加存储

SAN：存储区域网络

云存储

这里主要以本地磁盘为存储介质讲一下计划任务的添加使用，基本的备份脚本，其它存储介质只是介质的访问方式可能不大一样。

1、查看磁盘空间情况：

既然是定时备份，就要选择一个空间充足的磁盘空间，避免出现因空间不足导致备份失败，数据丢失的恶果！

存储到当前磁盘这是最简单，却是最不推荐的；服务器有多块硬盘，最好是把备份存放到另一块硬盘上；有条件就选择更好更安全的存储介质；

```
# df -h
Filesystem                Size      Used Avail Use% Mounted on
/dev/mapper/VolGroup-lv_root 50G        46G    1.6G   97% /
tmpfs                     1.9G        92K    1.9G    1% /dev/shm
/dev/sda1                  485M        39M    421M    9% /boot
/dev/mapper/VolGroup-lv_home 534G       3.6G   503G    1% /home
```

- 1
- 2
- 3
- 4
- 5
- 6

2、创建备份目录：

上面我们使用命令看出/home下空间比较充足，所以可以考虑在/home保存备份文件；

```
cd /home
mkdir backup
cd backup
```

- 1
- 2
- 3

3、创建备份Shell脚本:

注意把以下命令中的DatabaseName换为实际的数据库名称；

当然，你也可以使用其它的命名规则！

```
vi bkDatabaseName.sh
1
```

输入/粘贴以下内容：

```
#!/bin/bash
mysqldump -uusername -ppassword DatabaseName > /home/backup/DatabaseName_$(date
+%Y%m%d_%H%M%S).sql
```

- 1
- 2

对备份进行压缩：

```
#!/bin/bash
mysqldump -uusername -ppassword DatabaseName | gzip >
/home/backup/DatabaseName_$(date +%Y%m%d_%H%M%S).sql.gz
    • 1
    • 2
```

注意：

把 username 替换为实际的用户名；

把 password 替换为实际的密码；

把 DatabaseName 替换为实际的数据库名；

4、添加可执行权限：

```
chmod u+x bkDatabaseName.sh
    1
```

添加可执行权限之后先执行一下，看看脚本有没有错误，能不能正常使用；

```
./bkDatabaseName.sh
    1
```

5、添加计划任务

检测或安装 crontab

确认crontab是否安装：

执行 crontab 命令如果报 command not found，就表明没有安装

```
# crontab
-bash: crontab: command not found
    • 1
    • 2
```

如时没有安装 crontab，需要先安装它，具体步骤请参考：

[CentOS下使用yum命令安装计划任务程序crontab](#)

[使用rpm命令从CentOS系统盘安装计划任务程序crontab](#)

添加计划任务

执行命令：

```
crontab -e
    1
```

这时就像使用vi编辑器一样，可以对计划任务进行编辑。

输入以下内容并保存：

```
*/1 * * * * /home/backup/bkDatabaseName.sh
1
```

具体是什么意思呢？

意思是每一分钟执行一次shell脚本“/home/backup/bkDatabaseName.sh”。

6、测试任务是否执行

很简单，我们就执行几次“ls”命令，看看一分钟过后文件有没有被创建就可以了！

如果任务执行失败了，可以通过以下命令查看任务日志：

```
# tail -f /var/log/cron
1
```

输出类似如下：

```
Sep 30 14:01:01 bogon run-parts(/etc/cron.hourly)[2503]: starting 0anacron
Sep 30 14:01:01 bogon run-parts(/etc/cron.hourly)[2512]: finished 0anacron
Sep 30 15:01:01 bogon CROND[3092]: (root) CMD (run-parts /etc/cron.hourly)
Sep 30 15:01:01 bogon run-parts(/etc/cron.hourly)[3092]: starting 0anacron
Sep 30 15:01:02 bogon run-parts(/etc/cron.hourly)[3101]: finished 0anacron
Sep 30 15:50:44 bogon crontab[3598]: (root) BEGIN EDIT (root)
Sep 30 16:01:01 bogon CROND[3705]: (root) CMD (run-parts /etc/cron.hourly)
Sep 30 16:01:01 bogon run-parts(/etc/cron.hourly)[3705]: starting 0anacron
Sep 30 16:01:01 bogon run-parts(/etc/cron.hourly)[3714]: finished 0anacron
Sep 30 16:15:29 bogon crontab[3598]: (root) END EDIT (root)
• 1
• 2
• 3
• 4
• 5
• 6
• 7
• 8
• 9
• 10
```