使用Xtrabackup进行MySQL备份：

一、安装

1、简介

Xtrabackup是由percona提供的mysql数据库备份工具，据官方介绍，这也是世界上惟一一款开源的能够对innodb和xtradb数据库进行(物理)热备的工具。特点：

(1)备份过程快速、可靠；

(2)备份过程不会打断正在执行的事务；

(3)能够基于压缩等功能节约磁盘空间和流量；

(4)自动实现备份检验；

(5)还原速度快；

(6)可以使用完全备份和增量备份

2、安装

其最新版的软件可从 http://www.percona.com/software/percona-xtrabackup/ 获得。本文基于RHEL5.8的系统，因此，直接下载相应版本的rpm包安装即可，这里不再演示其过程。二进制包直接解压创建/bin下的软连接即可

//更高级的单表备份功能需要启用innodb\_file\_per\_table这个选项,因为默认的都在同一个表空间中存储。在配置文件中修改(这个这么好为什么不默认这样呢？)

// xtrabackup是用于热备份innodb, xtradb表中数据的工具，不能备份其他类型的表，也不能备份数据表结构；

innobackupex是将xtrabackup进行封装的perl脚本，提供了备份myisam表的能力。

//innodb表的导入导出功能不能通过拷贝物理文件来实现，需要借助它的导入导出功能

二、备份的实现

1、完全备份

一般库不大（< 200G）,我们就选择用全量备份

# innobackupex --user=DBUSER --password=DBUSERPASS /path/to/BACKUP-DIR/

如果要使用一个最小权限的用户进行备份，则可基于如下命令创建此类用户：

mysql> CREATE USER ’bkpuser’@’localhost’ IDENTIFIED BY ’s3cret’;

mysql> REVOKE ALL PRIVILEGES, GRANT OPTION FROM ’bkpuser’;

mysql> GRANT RELOAD, LOCK TABLES, REPLICATION CLIENT ON \*.\* TO ’bkpuser’@’localhost’;

mysql> FLUSH PRIVILEGES;

使用innobakupex备份时，其会调用xtrabackup备份所有的InnoDB表，复制所有关于表结构定义的相关文件(.frm)、以及MyISAM、MERGE、CSV和ARCHIVE表的相关文件，同时还会备份触发器和数据库配置信息相关的文件。这些文件会被保存至一个以时间命令的目录中。

在备份的同时，innobackupex还会在备份目录中创建如下文件：

(1)xtrabackup\_checkpoints —— 备份类型（如完全或增量）、备份状态（如是否已经为prepared状态）和LSN(日志序列号)范围信息；

每个InnoDB页(通常为16k大小)都会包含一个日志序列号，即LSN。LSN是整个数据库系统的系统版本号，每个页面相关的LSN能够表明此页面最近是如何发生改变的。

(2)xtrabackup\_binlog\_info —— mysql服务器当前正在使用的二进制日志文件及至备份这一刻为止二进制日志事件的位置。

(3)xtrabackup\_binlog\_pos\_innodb —— 二进制日志文件及用于InnoDB或XtraDB表的二进制日志文件的当前position。

(4)xtrabackup\_binary —— 备份中用到的xtrabackup的可执行文件；

(5)backup-my.cnf —— 备份命令用到的配置选项信息；

在使用innobackupex进行备份时，还可以使用--no-timestamp选项来阻止命令自动创建一个以时间命名的目录；如此一来，innobackupex命令将会创建一个BACKUP-DIR目录来存储备份数据。

2、准备(prepare)一个完全备份

一般情况下，在备份完成后，数据尚且不能用于恢复操作，因为备份的数据中可能会包含尚未提交的事务或已经提交但尚未同步至数据文件中的事务。因此，此时数据文件仍处理不一致状态。“准备”的主要作用正是通过回滚未提交的事务及同步已经提交的事务至数据文件也使得数据文件处于一致性状态。

//如果还有增量备份不能这样做！

innobakupex命令的--apply-log选项可用于实现上述功能。如下面的命令：

# innobackupex --apply-log /path/to/BACKUP-DIR

如果执行正确，其最后输出的几行信息通常如下：

xtrabackup: starting shutdown with innodb\_fast\_shutdown = 1

120407 9:01:36 InnoDB: Starting shutdown...

120407 9:01:40 InnoDB: Shutdown completed; log sequence number 92036620

120407 09:01:40 innobackupex: completed OK!

在实现“准备”的过程中，innobackupex通常还可以使用--use-memory选项来指定其可以使用的内存的大小，默认通常为100M。如果有足够的内存可用，可以多划分一些内存给prepare的过程，以提高其完成速度。

3、从一个完全备份中恢复数据

innobackupex命令的--copy-back选项用于执行恢复操作，其通过复制所有数据相关的文件至mysql服务器DATADIR目录中来执行恢复过程。innobackupex通过backup-my.cnf来获取DATADIR目录的相关信息。

# innobackupex --copy-back /path/to/BACKUP-DIR

如果执行正确，其输出信息的最后几行通常如下：

innobackupex: Starting to copy InnoDB log files

innobackupex: in '/backup/2012-04-07\_08-17-03'

innobackupex: back to original InnoDB log directory '/mydata/data'

innobackupex: Finished copying back files.

120407 09:36:10 innobackupex: completed OK!

请确保如上信息的最行一行出现“innobackupex: completed OK!”。

当数据恢复至DATADIR目录以后，还需要确保所有数据文件的属主和属组均为正确的用户，如mysql，否则，在启动mysqld之前还需要事先修改数据文件的属主和属组。如：

# chown -R mysql:mysql /mydata/data/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

4、使用innobackupex进行**增量备份**

每个InnoDB的页面都会包含一个LSN信息，每当相关的数据发生改变，相关的页面的LSN就会自动增长。这正是InnoDB表可以进行增量备份的基础，即innobackupex通过备份上次完全备份之后发生改变的页面来实现。

要实现第一次增量备份，可以使用下面的命令进行：

# innobackupex --incremental /backup --incremental-basedir=BASEDIR时间戳目录

其中，BASEDIR指的是**完全备份所在的目录**，此命令执行结束后，innobackupex命令会在/backup目录中创建一个**新的以时间命名的目录**以存放所有的增量备份数据。另外，在执行过**增量备份之后再一次进行增量备份时，其--incremental-basedir应该指向上一次的增量备份所在的目录**。

需要注意的是，增量备份仅能应用于InnoDB或XtraDB表，对于MyISAM表而言，执行增量备份时其实进行的是完全备份。

增量备份也是不能直接还原的需要做准备

“准备”(prepare)增量备份与整理完全备份有着一些不同，尤其要注意的是：

(1)需要在每个备份(包括完全和各个增量备份)上，将已经提交的事务进行“重放”。“重放”之后，所有的备份数据将合并到完全备份上。

(2)基于所有的备份将未提交的事务进行“回滚”。

于是，操作就变成了：

# innobackupex --apply-log **--redo-only** BASE-DIR

//如果不使用，后面的增量备份恢复都没办法生效

接着执行：

# innobackupex --apply-log --redo-only BASE-DIR --incremental-dir=INCREMENTAL-DIR-1

而后是第二个增量：

# innobackupex --apply-log --redo-only BASE-DIR --incremental-dir=INCREMENTAL-DIR-2

其中BASE-DIR指的是完全备份所在的目录，而INCREMENTAL-DIR-1指的是第一次增量备份的目录，INCREMENTAL-DIR-2指的是第二次增量备份的目录，其它依次类推，即如果有多次增量备份，每一次都要执行如上操作；

#################

从一个完全备份和增量备份中恢复数据以及二进制日志的即时点恢复

先找到二进制日志，找到最近一次增量备份之后位置的记录，最近一次位置在增量备份中的



> /tmp/mybinlog.sql







改权限：



然后再将即时点的数据 恢复回来



########################

5、Xtrabackup的“流”及“备份压缩”功能

Xtrabackup对备份的数据文件支持“流”功能，即可以将备份的数据通过STDOUT传输给tar程序进行归档，而不是默认的直接保存至某备份目录中。要使用此功能，仅需要使用--stream选项即可。如：

# innobackupex --stream=tar /backup | gzip > /backup/`date +%F\_%H-%M-%S`.tar.gz

甚至也可以使用类似如下命令将数据备份至其它服务器：

# innobackupex --stream=tar /backup | ssh user@www.magedu.com "cat - > /backups/`date +%F\_%H-%M-%S`.tar"

此外，在执行本地备份时，还可以使用--parallel选项对多个文件进行并行复制。此选项用于指定在复制时启动的线程数目。当然，在实际进行备份时要利用此功能的便利性，也需要启用innodb\_file\_per\_table选项或共享的表空间通过innodb\_data\_file\_path选项存储在多个ibdata文件中。对某一数据库的多个文件的复制无法利用到此功能。其简单使用方法如下：

# innobackupex --parallel /path/to/backup

同时，innobackupex备份的数据文件也可以存储至远程主机，这可以使用--remote-host选项来实现：

# innobackupex --remote-host=root@www.magedu.com /path/IN/REMOTE/HOST/to/backup

6、导入或导出单张表

默认情况下，InnoDB表不能通过直接复制表文件的方式在mysql服务器之间进行移植，即便使用了innodb\_file\_per\_table选项。而使用Xtrabackup工具可以实现此种功能，不过，此时需要“导出”表的mysql服务器启用了innodb\_file\_per\_table选项（严格来说，是要“导出”的表在其创建之前，mysql服务器就启用了innodb\_file\_per\_table选项），并且“导入”表的服务器同时启用了innodb\_file\_per\_table和innodb\_expand\_import选项。

(1)“导出”表

导出表是在备份的prepare阶段进行的，因此，一旦完全备份完成，就可以在preparef过程中通过--export选项将某表导出了：

# innobackupex --apply-log --export /path/to/backup

此命令会为每个innodb表的表空间创建一个以.exp结尾的文件，这些以.exp结尾的文件则可以用于导入至其它服务器。

(2)“导入”表

要在mysql服务器上导入来自于其它服务器的某innodb表，需要先在当前服务器上创建一个跟原表表结构一致的表，而后才能实现将表导入：

mysql> CREATE TABLE mytable (...) ENGINE=InnoDB;

然后将此表的表空间删除：

mysql> ALTER TABLE mydatabase.mytable DISCARD TABLESPACE;

接下来，将来自于“导出”表的服务器的mytable表的mytable.ibd和mytable.exp文件复制到当前服务器的数据目录，然后使用如下命令将其“导入”：

mysql> ALTER TABLE mydatabase.mytable IMPORT TABLESPACE;

7、使用Xtrabackup对数据库进行部分备份

Xtrabackup也可以实现部分备份，即只备份某个或某些指定的数据库或某数据库中的某个或某些表。但要使用此功能，必须启用innodb\_file\_per\_table选项，即每张表保存为一个独立的文件。同时，其也不支持--stream选项，即不支持将数据通过管道传输给其它程序进行处理。

此外，还原部分备份跟还原全部数据的备份也有所不同，即你不能通过简单地将prepared的部分备份使用--copy-back选项直接复制回数据目录，而是要通过导入表的方向来实现还原。当然，有些情况下，部分备份也可以直接通过--copy-back进行还原，但这种方式还原而来的数据多数会产生数据不一致的问题，因此，无论如何不推荐使用这种方式。

(1)创建部分备份

创建部分备份的方式有三种：正则表达式(--include), 枚举表文件(--tables-file)和列出要备份的数据库(--databases)。

(a)使用--include

使用--include时，要求为其指定要备份的表的完整名称，即形如databasename.tablename，如：

# innobackupex --include='^mageedu[.]tb1' /path/to/backup

(b)使用--tables-file

此选项的参数需要是一个文件名，此文件中每行包含一个要备份的表的完整名称；如：

# echo -e 'mageedu.tb1\nmageedu.tb2' > /tmp/tables.txt

# innobackupex --tables-file=/tmp/tables.txt /path/to/backup

(c)使用--databases

此选项接受的参数为数据名，如果要指定多个数据库，彼此间需要以空格隔开；同时，在指定某数据库时，也可以只指定其中的某张表。此外，此选项也可以接受一个文件为参数，文件中每一行为一个要备份的对象。如：

# innobackupex --databases='mageedu.tb1 testdb" /path/to/backup

(2)整理(preparing)部分备份

prepare部分备份的过程类似于导出表的过程，要使用--export选项进行：

# innobackupex --apply-log --export /pat/to/partial/backup

此命令执行过程中，innobackupex会调用xtrabackup命令从数据字典中移除缺失的表，因此，会显示出许多关于“表不存在”类的警告信息。同时，也会显示出为备份文件中存在的表创建.exp文件的相关信息。

(3)还原部分备份

还原部分备份的过程跟导入表的过程相同。当然，也可以通过直接复制prepared状态的备份直接至数据目录中实现还原，不要此时要求数据目录处于一致状态。

**相关原理**

**完整备份的原理：**

对于InnoDB，XtraBackup基于InnoDB的crash-recovery功能进行备份。

crash-recovery是这样的：InnoDB维护了一个redo log，又称为 transaction log，也叫事务日志，它包含了InnoDB数据的所有改动情况。InnoDB启动的时候先去检查datafile和transaction log，然后应用所有已提交的事务并回滚所有未提交的事务。

XtraBackup在备份的时候并不锁定表，而是一页一页地复制InnoDB的数据，与此同时，XtraBackup还有另外一个线程监视着transactions log，一旦log发生变化，就把变化过的log pages复制走(因为transactions log文件大小有限，写满之后，就会从头再开始写，新数据可能会覆盖到旧的数据，所以一旦变化就要立刻复制走）。在全部数据文件复制完成之后，停止复制logfile。

XtraBackup采用了其内置的InnoDB库以read-write模式打开InnoDB的数据文件，然后每次读写1MB(1MB/16KB=64page)的数据，一页一页地遍历，同时用InnoDB的buf\_page\_is\_corrupted()函数检查此页的数据是否正常，如果正常则进行复制，如不正常则重新读取，最多重读10次，如果还是失败，则备份失败退出。复制transactions log的原理也是一样的，只不过每次读写512KB(512KB/16KB=32page)的数据。

由于XtraBackup其内置的InnoDB库打开文件的时候是rw的，所以运行XtraBackup的用户，必须对InnoDB的数据文件具有读写权限。

由于XtraBackup要从文件系统中复制大量的数据，所以它尽可能地使用posix\_fadvise()，来告诉OS不要缓存读取到的数据(因为这些数据不会重用到了)，从而提升性能。如果要缓存的话，大量的数据会对OS的虚拟内存造成很大的压力，其它进程（如mysqld）很有可能会被swap出去，这样就出问题了。同时，XtraBackup在读取数据的时候还尽可能地预读。

由于不锁表，所以复制出来的数据是不一致的，数据的一致性是在恢复的时候使用crash-recovery进行实现的。

对于MyISAM,XtraBackup还是首先锁定所有的表，然后复制所有文件。

**增量备份的原理：**

在完整备份和增量备份文件中都有一个文件xtrabackup\_checkpoints会记录备份完成时检查点的LSN。在进行新的增量备份时，XtraBackup会比较表空间中每页的LSN是否大于上次备份完成的LSN，如果是，则备份该页，并记录当前检查点的LSN。