Backing up a Sumac database on a Macintosh MySQL server

2011-04-29

Background

Sumac Gold is the multi-user version of Sumac. It stores its data in a multi-user shared database. This database is managed by software named MySQL.

It is very important that you back up the data being managed by MySQL. However, you cannot just copy some files to a backup device. This is because, on a Macintosh, MySQL puts its data into folders that are accessible only to user accounts which are private to the operating system. In other words, you cannot access the files directly.

Fortunately, MySQL comes with utility programs that allow you to quickly copy the contents of a complete database into a file, which you can then back up. Even better, this file is quite small compared to the size of the database. So you can use the utility program to create this file on a regular basis, and make sure that the file is being backed up correctly.

Introduction

This document explains how to manually run a command to back up your database. Then it shows you how to automate this, so that the command will be run automatically on a regular basis.

Manual Backup

Overview

The program you need to run is named mysqldump. This program has no user interface – no buttons, menus, or windows. Instead, you use this program by running Terminal, and typing a command into the Terminal window. The command you need to perform will look something like this:

```
/usr/local/mysqlFolder/bin/mysqldump -h localhost --user=username --password=password
databaseName > ./Documents/SumacBackup.sql
```

Note that the above should be typed as a *single line*, even though it may wrap to multiple lines depending on your window size.

Here is what the different parts of the above statement do:

`/usr/local/mysqlFolder/bin/mysqldump`	This indicates where the mysqldump program resides and tells the operating system to run it.
`-h localhost`	Connect to the database manager on the computer where the mysqldump program is running. Note: you need to run this command on the Macintosh that is the server.
`--user=username`	Use this user ID to connect to the database server.
`--password=password`	Use this password to connect to the database server.
`databaseName`	This is the name of the database to be backed up.
`> ./Documents/SumacBackup.sql`	This says to put the backup data into a file named SumacBackup.sql, in the Documents folder. You can specify another location if it would be more convenient.

Our Mission

The problem is that the items that are red are unknown. We need to figure them out. We must remove all the red things in the above statement, so that we will be able to actually use the command.

Step 1 – Find `databaseName`

You can determine the name of the database by running Sumac and choosing the Show Configuration command from the Utilities menu. It tells you the name of the database you are using.

Let's pretend that the name of your database is CharityDB. We can put this into our command string to get the following:

```
/usr/local/mysqlFolder/bin/mysqldump -h localhost --user=username --password=password
CharityDB > ./Documents/SumacBackup.sql
```

Step 2 – Find `mysqlFolder`

Before we can run mysqldump, we need to know where MySQL was installed on your computer. The easiest way to determine this is to run Terminal and ask it to list the contents of the directory where MySQL was (most likely) installed. "ls" is the command for listing a directory's contents. Here is what to do:

Run Terminal and key in:

```
ls /usr/local
```

then press Return or Enter.

The command will show what is in the /usr/local directory. There will probably be a folder named "mysql". It is possible that instead of mysql, the folder will have a longer name which incorporates a version number. For convenience, let's pretend that what you saw was `mysql`. Enter a new ls command as follows:

```
ls /usr/local/mysql/bin
```

When you enter this command, an alphabetical list of file names appears. It should contain mysqldump. If so, then we have found the right mysqlFolder (the one named *mysql*). At this point, we can replaced mysqlFolder with the name of the folder we found (*mysql*), so the backup command is as follows:

```
/usr/local/mysql/bin/mysqldump -h localhost --user=username --password=password CharityDB
> ./Documents/SumacBackup.sql
```

Step 3 – Find `username`

If you have high security needs, your database may have been installed with other user IDs and passwords, which your administrator will know.

If you did a standard Sumac installation of the database, then `username` should be *cfrUser*.

```
/usr/local/mysql/bin/mysqldump -h localhost --user=cfrUser --password=password CharityDB >
./Documents/SumacBackup.sql
```

Step 4 – Find `password`

If you did a standard Sumac installation of the database, then `password` should be *cfrPassword*.

```
/usr/local/mysql/bin/mysqldump -h localhost --user=cfrUser --password=cfrPassword CharityDB
> ./Documents/SumacBackup.sql
```

Step 5 – Try it

If you enter the following command (perhaps modified as described in the above steps) into a Terminal window:

```
/usr/local/mysql/bin/mysqldump -h localhost --user=cfrUser --password=cfrPassword CharityDB
> ./Documents/SumacBackup.sql
```

it should create a file named SumacBackup.sql in the Documents folder. Make sure that the backup file is there, and reasonably sized (perhaps 5 to 20 MB depending on your database size).

Automating The Backup

Define the task using Automator

Now use Automator to create an application which performs this backup command.

Run Automator. It asks what type of script you want to create. Select the "Application" template and click Choose.

Scroll to find *Run Shell Script* in the list of available actions, then drag it to the empty list of actions on the right hand side. This creates a shell script with the command *cat*. Delete the word *cat* and replace it as follows:

Now, this script is going to run every day. There is one problem with repeatedly running the command that was tested in Step 5: it tries to put the backup file in the same place every time, but if the file already exists, then the mysqldump program will not run. So before you can run the backup command, you need to delete the backup file, so a new one can be created. In a shell script, the command for deleting a file is *rm*. So enter the rm command to remove the file, then press Return and enter the command that was tested in Step 5 above. Your entire script will look like this (two lines, the *rm* line followed by the *mysqldump* line, though it may wrap to more lines depending on your window width):

```
rm ./Documents/SumacBackup.sql
```

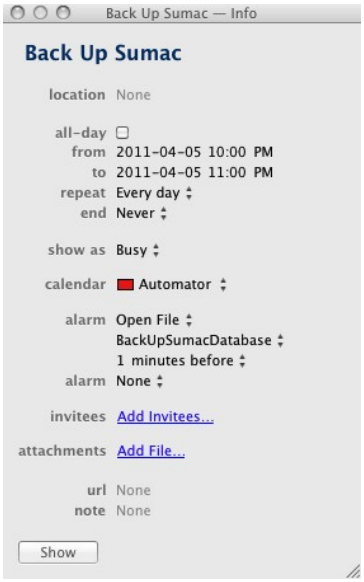
```
/usr/local/mysql/bin/mysqldump -h localhost --user=cfrUser --password=cfrPassword CharityDB
> ./Documents/SumacBackup.sql
```

Choose File > Save As. Save this Automator task with file format Application, giving it an appropriate name (e.g. *BackUpSumacDatabase*) and putting it in a place where it can stay indefinitely (since it will be used every day).

Make the task run every day using iCal

Now you need to tell iCal to automate this task so that it will run every day. Here are the steps to follow:

- Run iCal. Click to select the current date.
- Choose the New command from the File menu, to create a new event.
- Give the new task a meaningful name, e.g. *Back Up Sumac*.
- Set the *from* time to a time when no one is using the server, perhaps late at night or very early in the morning.
- Choose *Every day* from the *repeat* menu.
- Under *alarm*, choose *Open File*, then choose the name of the file (the one created using Automator as described above), then indicate it should be opened *1 minute before*.
- The finished iCal task should look something like this:



- Close and save the task.

Finally

Make Sure it Works

The above steps cause a new file named *SumacBackup.sql* to be created at 10 p.m. each day. For a couple of days, when you come to work in the morning, make sure that the file is being created and updated every night.

Make Sure The File is Being Backed Up

Make sure that the *SumacBackup.sql* file is being backed up to an external removable medium every day.