

GNU/Linux Administration - Support #609

Migrating a MySQL Database To a PostgreSQL Database

04/30/2015 02:38 PM - Daniel Curtis

Status:	Closed	Start date:	04/30/2015
Priority:	Normal	Due date:	
Assignee:	Daniel Curtis	% Done:	100%
Category:	Database Server	Estimated time:	1.00 hour
Target version:	Debian	Spent time:	5.00 hours

Description

This is a guide for migrating a MySQL database to a PostgreSQL database using Debian 8, Ubuntu 14.04, or FreeBSD 9. This process was done using a VM and access granted to the VM for the MySQL database and the PostgreSQL database, this may not be ideal for all use cases.

Prepare the Environment

Debian/Ubuntu

- Make sure the system is up to date:

```
apt-get update && apt-get upgrade -y
```

- Install a few dependencies:

```
apt-get install virtualenv python-dev
```

- Setup the python virtual environment:

```
virtualenv --no-site-packages ~/envs/py-mysql2pgsql
source ~/envs/py-mysql2pgsql/bin/activate
```

- Install py-mysql2pgsql

```
pip install py-mysql2pgsql
```

FreeBSD

- Make sure the system is up to date:

```
portsnap fetch extract && portmaster -a
```

- Install py-mysql2pgsql:

```
portmaster databases/py-mysql2pgsql
```

Migrate The Database

- Generate a new config file:

```
py-mysql2pgsql
```

- *Example output:*

```
No configuration file found.  
A new file has been initialized at: mysql2pgsql.yml  
Please review the configuration and retry...
```

- Edit the mysql2pgsql config file:

```
nano mysql2pgsql.yml
```

- And modify the config file to connect to both the mysql and postgresql databases:

```
mysql:  
  hostname: mysql.example.com  
  port: 3306  
  username: exampleuser  
  password: SuperSecretPassword  
  database: exampledb  
  compress: false  
  
destination:  
  file:  
  postgres:  
    hostname: pgsq.example.com  
    port: 5432  
    username: exampleuser  
    password: SuperSecretPassword  
    database: exampledb  
  
# if supress_data is true, only the schema definition will be exported/migrated, and not t  
he data  
supress_data: false  
  
# if supress_ddl is true, only the data will be exported/imported, and not the schema  
supress_ddl: false  
  
# if force_truncate is true, forces a table truncate before table loading  
force_truncate: false
```

- Now run the migration:

```
py-mysql2pgsql -v
```

Resources

- <https://github.com/philipsoutham/py-mysql2pgsql>

History

#1 - 05/01/2015 04:21 PM - Daniel Curtis

- Description updated

#2 - 05/04/2015 03:28 PM - Daniel Curtis

- Description updated

- Status changed from New to In Progress

- % Done changed from 0 to 50

#3 - 05/04/2015 04:44 PM - Daniel Curtis

- *Description updated*
- *% Done changed from 50 to 70*

#4 - 05/06/2015 12:09 PM - Daniel Curtis

- *Status changed from In Progress to Resolved*
- *% Done changed from 70 to 100*

#5 - 05/06/2015 12:10 PM - Daniel Curtis

- *Description updated*

#6 - 05/10/2015 09:20 AM - Daniel Curtis

- *Status changed from Resolved to Closed*