

Readme

There are 2 scripts in this repository. They are

1. test-postgres.sh
2. update-postgres.sh

test-postgres.sh

This tests the postgres database on compose by a node.js application on bluemix.

1. Create a Postgres database using compose (<https://compose.io>)
2. Register the Postgres service in Bluemix.
 - a. This can be achieved using bluemix console
 - b. This also can be done by using create-service-postgres.sh. The usage is

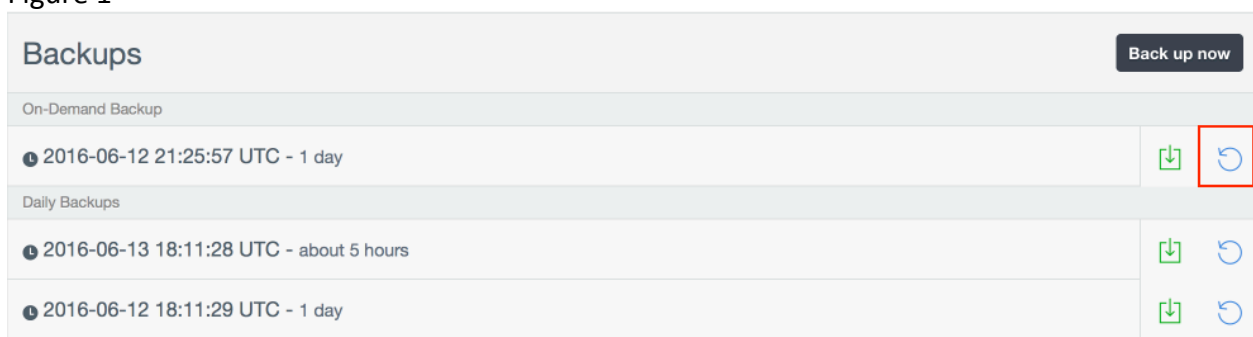
```
./create-service-postgres.sh <hostname> <port no> <dbname>  
<userid> <password>.
```

3. Run test-postgres.sh. This will show whether the connection to the Postgres database is successful or not.

Back up Postgres database in <https://compose.io>

To backup the existing database, you need to log in to <https://compose.io> website. By default the database will be backed up everyday. Either the scheduled backup can be used or new backup can be created on demand. Once the database is backed up, it looks like this:

Figure 1



Restore a backed-up database

To restore a backed-up database, select the backup version that needs to be restored and select the restore icon (marked in red in the above picture). Complete it by giving a new name to the Deployment.

Once the database is restored, get the Deployment details. These details are needed to update the service in Bluemix. For example, after restoration the database connection details look like

```
postgres://[username]:[password]@aws-us-east-1-portal.19.dblayer.com:10371/compose
```

update-postgres.sh

This script updates the Postgres service that was bound to the NodeJS application by test-postgres.sh. This script was restages the application and checks whether it's running successfully. The usage of the script is

```
./update-postgres.sh <hostname> <port no> <dbname> <userid> <password>
```