

# ElephantSQL

## PostgreSQL as a Service



Product overview

# Table of Contents

.....	1
ElephantSQL.....	3
PostgreSQL as a Service.....	3
What is PostgreSQL?.....	3
Product overview.....	4
Product Details.....	4
SQL Browser .....	4
Other SQL tools .....	5
Backups .....	6
Database statistics .....	6
Cache Hit Rate .....	6
Index usage .....	6
Index Hit Rate .....	6
Connections .....	7
Slow queries .....	7
Server metrics .....	7
CPU Usage .....	7
Memory Usage .....	8

# ElephantSQL

ElephantSQL is a PostgreSQL hosting service provided by 84codes AB.

## PostgreSQL as a Service

**ElephantSQL is a PostgreSQL database hosting service.** ElephantSQL will manage administrative tasks of PostgreSQL, such as installation, upgrades to latest stable version and backup handling.

ElephantSQL is also integrated to several cloud application platforms (also known as PaaS). With a click of a button your database is provisioned in the same data center as your application is hosted, and is ready to be used immediately.

## What is PostgreSQL?

PostgreSQL is an open source, object-relational database management system.

# Product overview

ElephantSQL offer tools to simplify the usage of the PostgreSQL database. Various monitoring tools will help you to overview server metrics, backups are handled automatically and we make it very easy to do *Point in time recovery*. The instance details, such as connection URL, statistics, open connections and your slow queries can be seen at the details pages in the control panel. Here you will also be able to restore backups and rotate your password. If you are on a dedicated plan (Happy Hippo or larger) you will be able to view server metrics, you can set up followers and you can view information for all your databases.

## Product Details

Immediately after the instance is created it will be provisioned to you. The instance details, such as connection URL, statistics, connections and your slow queries can be seen at the details pages. From here will you also be able to restore backups and rotate your password. If you are on a dedicated plan (Happy Hippo or larger) you will be able to view server metrics, you can set up followers and you can view information for all your databases.

Connection URL format: `postgres://username:password@hostname/databasename`

## ElephantSQL Console test

The screenshot shows the 'Database info' section of the ElephantSQL console. At the top, there are tabs for 'Management', 'Log', and 'Browser'. A 'Reset database' button is located in the top right corner. The main content is a table with the following rows:

URL	postgres://[redacted]@pellefant-01.db.elephantsql.com:5432/[redacted]
Hostname	pellefant-01.db.elephantsql.com
Database name	[redacted] <small>Use this for Maintenance DB in PgAdmin</small>
Username	[redacted]
Password	[redacted] <span>Rotate password ↻</span>
PostgreSQL version	9.4.4
Current database size	8 MB
Max database size	20 MB

Below the table, there are more tabs: 'Statistics', 'Connections', 'Slow queries', 'Followers', 'Server metrics', and 'Backups'.

### Illustration 1: ElephantSQL Details

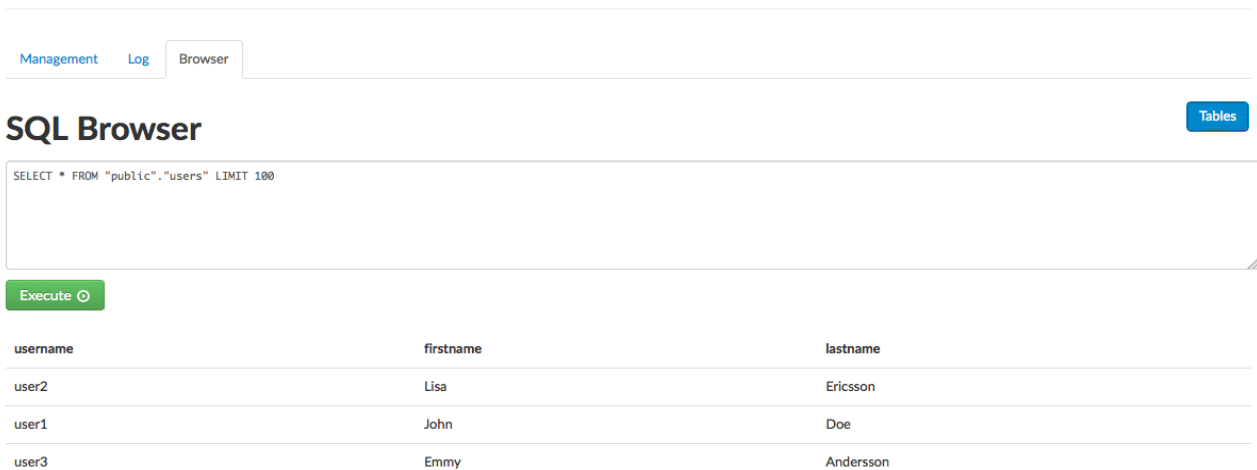
Once you have created your account you can get started using your PostgreSQL database by using any of the guides listed here: <http://www.elephantsql.com/docs/index.html>. These tutorials cover the basics of how to get started with ElephantSQL.

## SQL Browser

ElephantSQL provides a browser tool for sql queries where you can create, read, update and delete data direct from your web browser. A link to the browser can be found on the

console page for your instance. The different available tables for the database can be found under the tables button to the right.

## ElephantSQL Console test



Management Log Browser

### SQL Browser Tables

```
SELECT * FROM "public"."users" LIMIT 100
```

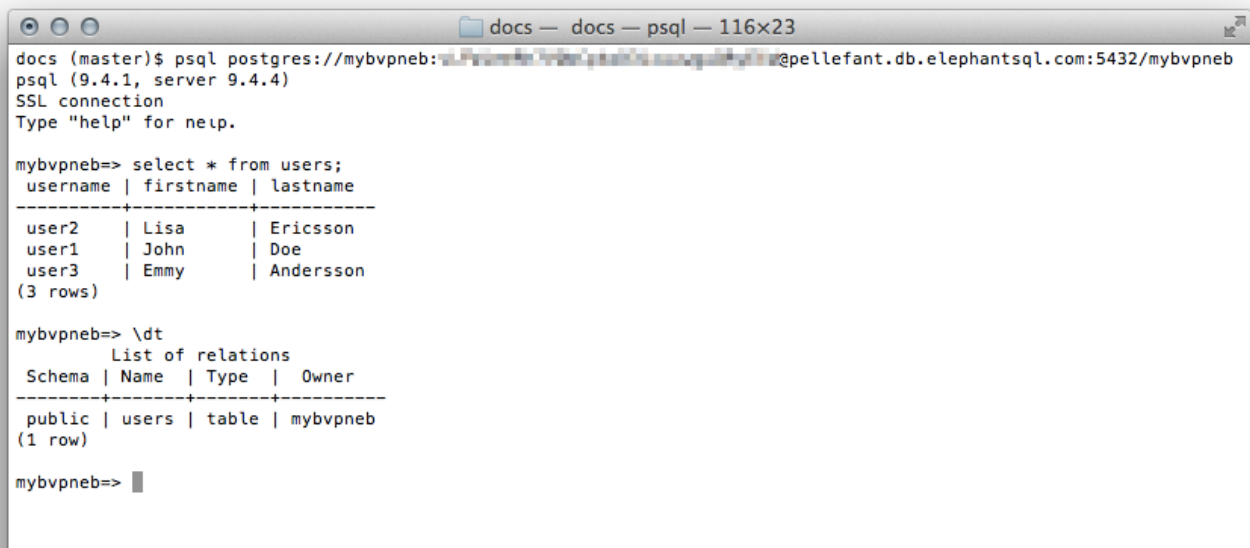
Execute

username	firstname	lastname
user2	Lisa	Ericsson
user1	John	Doe
user3	Emmy	Andersson

*Illustration 2: ElephantSQL SQL Browser*

## Other SQL tools

Other tools that can be used when handling SQL data and your database instance are [pgAdmin](#), and [psql](#). *pgAdmin* is a graphical administration client for PostgreSQL and *psql* is a terminal-based tool for work with PostgreSQL.



```
docs (master)$ psql postgres://mybvpneb:~@pelletfant.db.elephantsql.com:5432/mybvpneb
psql (9.4.1, server 9.4.4)
SSL connection
Type "help" for help.

mybvpneb=> select * from users;
 username | firstname | lastname
-----+-----+-----
 user2    | Lisa     | Ericsson
 user1    | John    | Doe
 user3    | Emmy   | Andersson
(3 rows)

mybvpneb=> \dt
      List of relations
 Schema | Name  | Type  | Owner
-----+-----+-----+-----
 public | users | table | mybvpneb
(1 row)

mybvpneb=> █
```

*Illustration 3: PSQL*

# Backups

Automated backups are performed every day. These are stored in a cloud file storage so they are always accessible to you.

Statistics Connections Slow queries Followers Server metrics **Backups** Database:

The backups are compressed with [lzop](#) to restore the following should work: `lzop -cd "$FILE_NAME" | psql "$DATABASE_NAME"`

Date	Size		
2015-04-29 08:01:22 +0000	35 MB	<a href="#">Download</a>	<a href="#">Restore database</a>
2015-04-28 08:01:22 +0000	35 MB	<a href="#">Download</a>	<a href="#">Restore database</a>

## Illustration 4: ElephantSQL Backups

The backups are compressed with [lzop](#), to restore the data you can usually use the following: `lzop -cd "$FILE_NAME" | psql "$DATABASE_NAME"`

# Database statistics

Statistics Connections Slow queries Followers Server metrics Backups Database:

### Database statistics

Index hit rate	100.00%	Should be 99%, else add indexes
Cache hit rate	99.99%	Should be 99%, else upgrade to a larger instance

**Index usage** Should be 99% for every table with more than 10 000 rows

Relname	Percent of times index used	Rows in table
account_events	0	311777
accounts	98	158228
account_transfers	99	23481

## Cache Hit Rate

Usually (in many applications) only a fraction of the data is regularly accessed. Postgres is tracking access patterns of data and will on its own keep frequently accessed data in cache. Cache hit rate show how many of your indexes that are within your cache. Generally you want your database to have a cache hit rate of about 99%.

## Index usage

If you're not somewhere around 99% on any table over 10,000 rows you may want to consider adding an index. Indexes are most valuable across very large tables. When examining where to add an index you should look at what kind of queries you're running and add indexes to some other id or on values that you're commonly filtering.

## Index Hit Rate

Show how many of your indexes that are within your cache. You should have an index hit rate around 99%, similar to your regular cache hit rate.

# Connections

Open connections can be seen in the tab open connections.

Since	Application name	Client address	State	Query	
2015-04-28 18:22:11 +0000		10.8.162.212	idle	SELECT hostname, username FROM servers	<a href="#">Cancel</a> <a href="#">Terminate</a>
2015-04-29 15:02:02 +0000	ElephantSQL Console	10.166.48.22	idle	SELECT * FROM clusters c WHERE not deleted and ready	<a href="#">Cancel</a> <a href="#">Terminate</a>

Illustration 5: ElephantSQL Connections

# Slow queries

Identify which queries that are running slowly in your system by checking the slow queries tab. Inefficient SQL queries can kill the performance of your application.

Total time	Calls	Rows	Avg. time	Avg. time/row	Cache hit rate	Query	
4073724 ms	40450173	40450173	0 ms	0 ms	100 %	UPDATE `databases` SET `current_size` = ? WHERE (`id` = ?)	<a href="#">Explain</a>
982927 ms	15798	41413213	62 ms	0 ms	100 %	SELECT * FROM `databases` WHERE (`deleted` IS FALSE)	<a href="#">Explain</a>

Illustration 6: ElephantSQL Slow queries

# Server metrics

## For dedicated plans only

ElephantSQL offers various monitoring tools. These tools will address performance issues promptly and automatically, before they impact your business. ElephantSQL monitoring include diagrams for CPU and Memory usage.

# CPU Usage

CPU Usage refers to how much work your processor is doing.

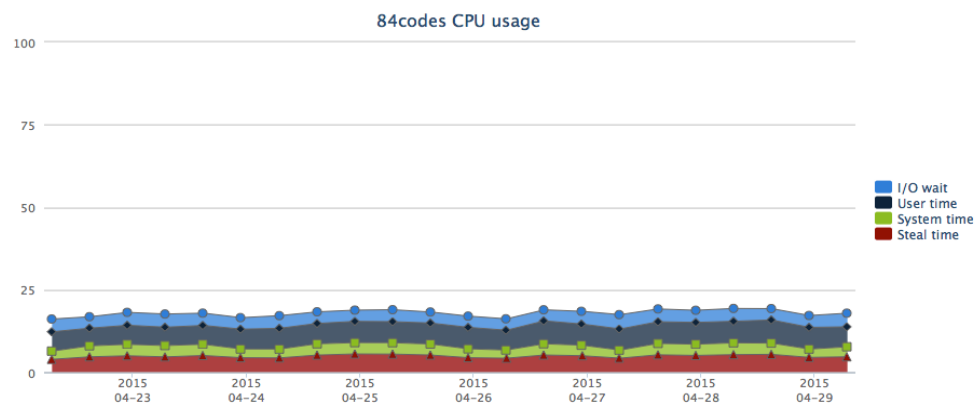


Illustration 7: ElephantSQL CPU Usage

- I/O Wait:**  
 Show percentage of time spent by the CPU waiting for a IO (input/output) operation to complete, the percentage of time the CPU have to wait on the disk.
- User time:**  
 Show percentage of time your program spends executing instructions in the CPU. In this case, the time the CPU spent running PostgreSQL.  
  
 If this is high it probably means you are on the limit of what your server can handle. You should consider upgrading before lack of CPU power becomes an serious issue.
- System time:**  
 Describes percentage of time the CPU spent running OS tasks.
- Steal time:**  
 Percentage of CPU time "stolen" by the virtualization system - time spent when the virtual CPU waits for a real CPU. If this is high it means that you are using too much CPU power. This can seriously impacting the performance of your server. You should probably upgrade to a larger instance.

## Memory Usage

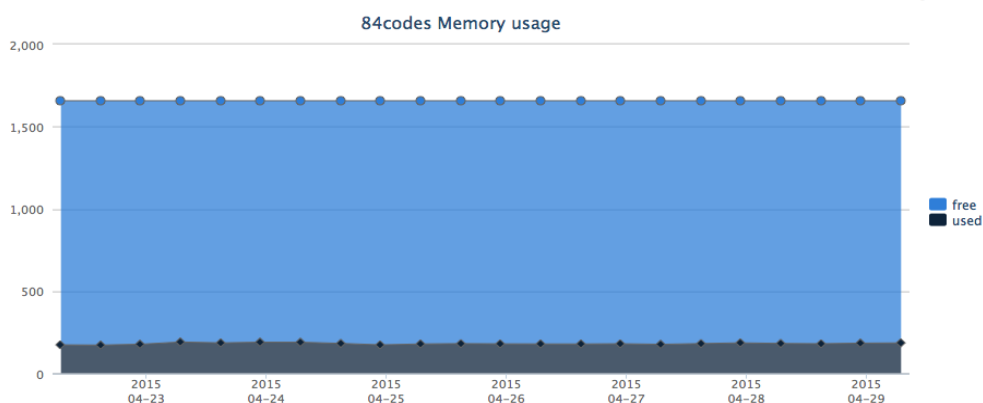


Illustration 8: ElephantSQL Memory Usage



- **Used:** Percentage of used memory.
- **Free:** Percentage of free memory.