

Postgres Plus Cloud Database

Presented by Dave Page

22nd March 2013

Introduction

▶ Community

- pgAdmin lead developer
- PostgreSQL core team member
- Installer maintainer
- Postgres Europe and Postgres Canada board member

▶ EnterpriseDB

- Chief Architect, Tools & Installers
- Infrastructure lead

What is a Cloud Database?

▶ Database taking advantage of a cloud environment

- Point and click deployment
- Point and click management
- Auto and manual scaling
- Auto healing

▶ Not merely "installed" in the cloud

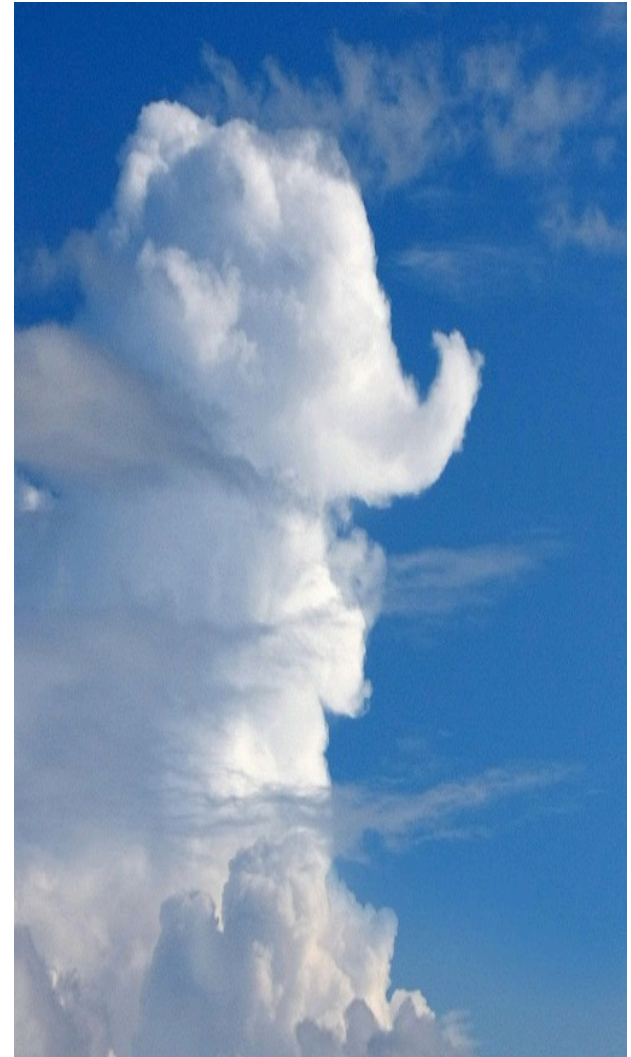
- Which would require similar administration to a physical machine.



What is PPCD?

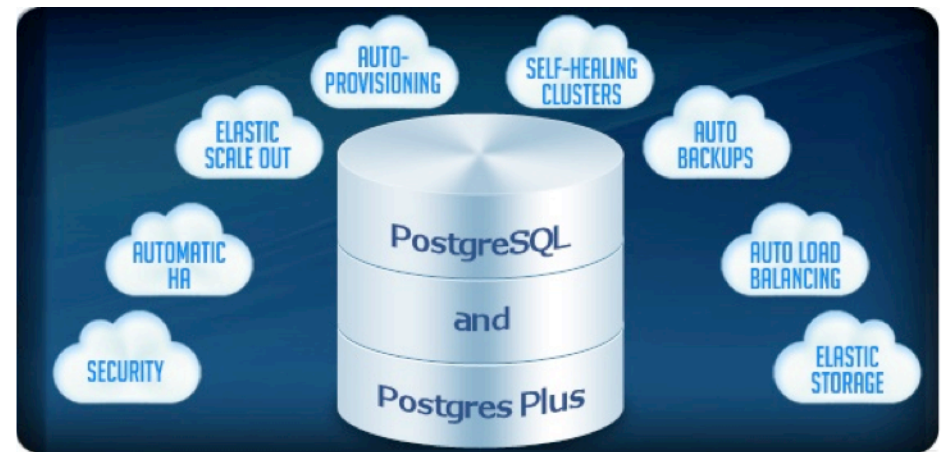
▶ Cloud database:

- Multi-platform:
 - *Amazon AWS*
 - *HP Cloud (OpenStack)*
 - *Citrix (CloudStack)*
 - *More coming soon...*
- Multi-server:
 - *PostgreSQL 9.1/9.2*
 - *Postgres Plus Advanced Server 9.1/9.2*

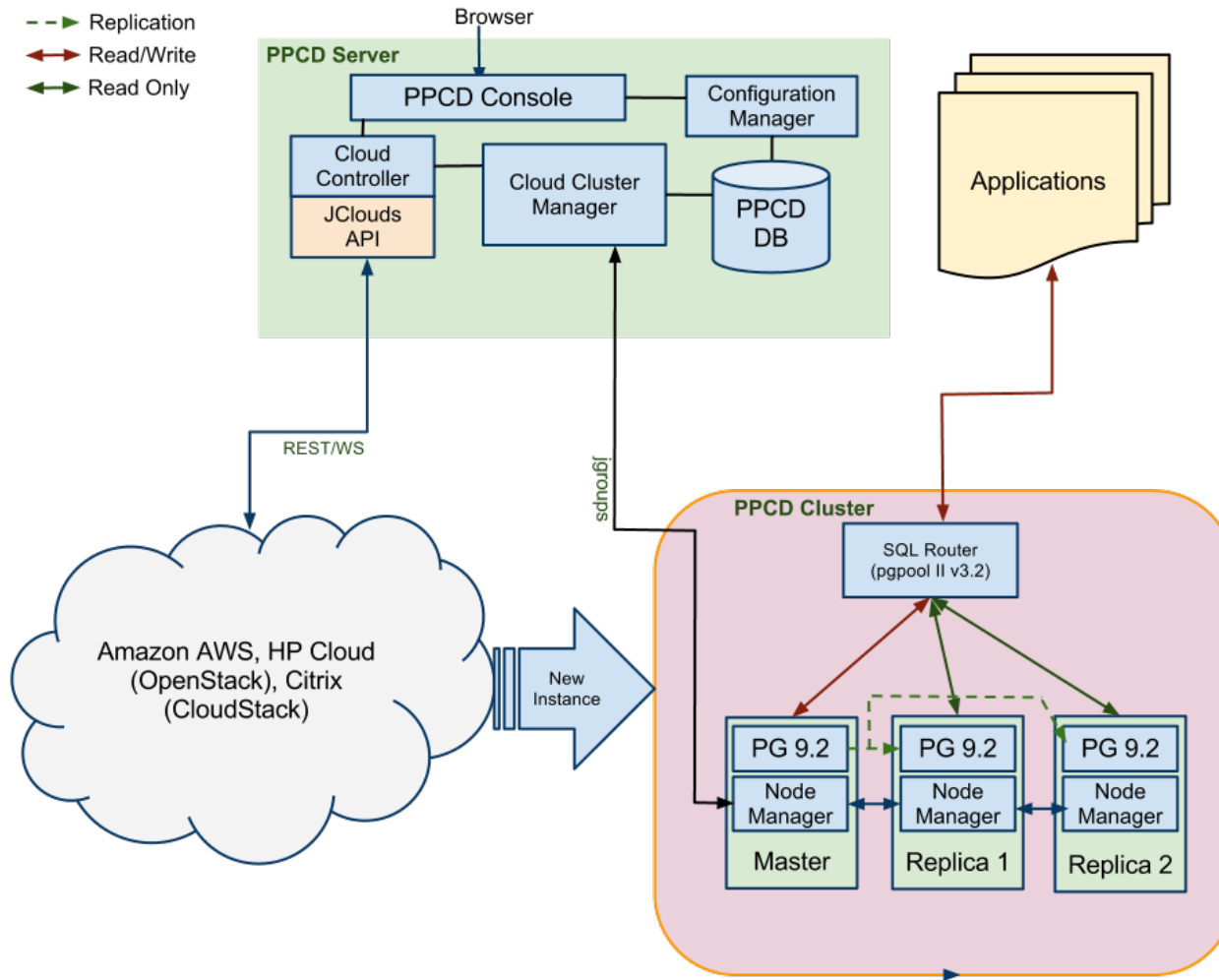


Features

- ▶ Web based management
- ▶ Data security
- ▶ Horizontal and vertical scaling
- ▶ Storage scaling
- ▶ Self-healing
- ▶ Extensible



Architecture



Web based management

- ▶ Deploy new database clusters with a simple wizard
- ▶ Perform manual maintenance with a few clicks:

- Add replicas and clones
- Perform backups
- Upgrade clusters
- Performance monitoring

The screenshot displays the EnterpriseDB PostgreSQL Plus Cloud Database web management interface. The browser address bar shows the URL <https://aws-us-east-1a.theclouddatabase.com>. The page title is "EnterpriseDB PostgreSQL Plus Cloud Database".

The main dashboard features a table with the following columns: NAME, STATUS, DATA SPACE %, LOAD, NETW, CLUST, and SERVI. The table contains one entry for "cluster1" with a status of "PENDING", 8% data space usage, and a load of 0.0. All status indicators (NETW, CLUST, SERVI) are green.

The "Details" section for "Cluster: cluster1" provides the following information:

- Creation Date: Tue Mar 05 09:04:01 EST 2013
- Username: postgres
- Size: 2gb
- Region: us-east-1
- Hardware: m1.small
- Engine Version: PostgreSQL 9.2.2 64bit
- Configuration: default
- Cluster healing mode:
 - Replace failed master with a new master
 - Replace failed master with existing replica
- Auto-Scaling Thresholds:
 - % of Storage Size used: 65
 - # of Server Connections: 10

The "Backup Settings" section includes:

- Backup Window: 12:00am - 2:00am
- Backup Retention: 1

The interface also includes a sidebar with navigation icons and a footer with "Configurations", "Monitoring", and "Events" links.

Live Demo – Create a new cluster

The screenshot shows the EnterpriseDB PostgreSQL Plus Cloud Database console. A modal dialog titled "Create a new Server Cluster" is open, showing the configuration for a new cluster named "cluster2".

Cluster: cluster1
Creation Date: Tue Mar 05 08:12:15 EST 2013
Username: postgres
Size: 2gb
Region: us-east-1
Hardware: m1.small
Engine Version: PostgreSQL 9.2.2 64bit
Configuration: default
Cluster healing mode:
 Replace failed master with a new master
 Replace failed master with existing replica
 Auto-Scaling Thresholds
% of Storage Size used: 65
of Server Connections: 95
Backup Window: 12:00am - 2:00am
Backup Retention: 1

Create a new Server Cluster

Step 1 | Step 2

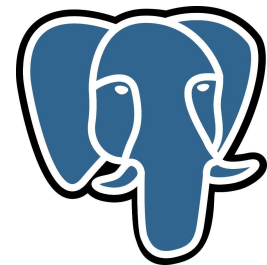
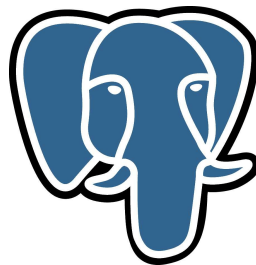
Provide the details for your cluster

Cluster Name: cluster2
Engine Version: PostgreSQL 9.2.2 64bit
Server Class: m1.small
Number of nodes: 2
Storage GB: 20
Master User: postgres
Master Password: Passw0rd

Next

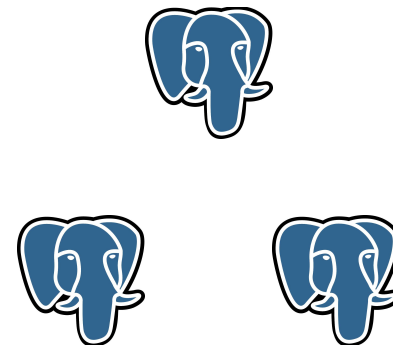
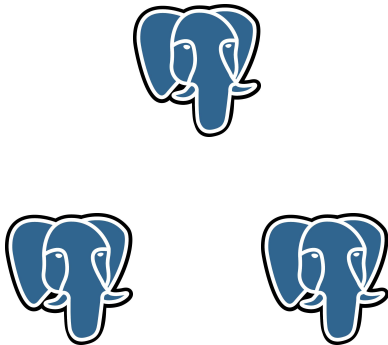
Data security - replicas

- ▶ Clusters contain one or more replicas of your data
- ▶ Uses streaming replication
- ▶ Replicas are used for read load balancing
- ▶ Replicas can be promoted to the master role at any time



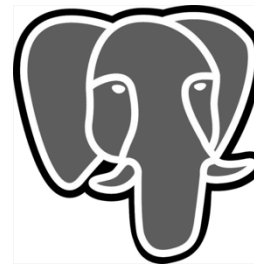
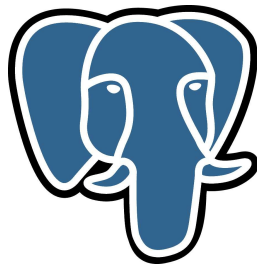
Data security - clones

- ▶ Clone a cluster at any time
- ▶ Useful for:
 - "online" backups
 - Reporting clusters
 - Development clusters



Data security - backups

- ▶ Automated backups to storage such as S3
- ▶ Manual on-demand backups
- ▶ Restore to a new cluster on demand



Live Demo – Run a backup

The screenshot shows the EnterpriseDB PostgreSQL Plus Cloud Database management console. A modal dialog box titled "Backup Data?" is open, prompting the user to create a backup. The dialog contains the following text:

Creates a backup of the data which can be recovered at any time.

Optional notes about this backup:

Below the text is a blue graphic with a white question mark and a white cloud. At the bottom of the dialog are "Backup" and "Cancel" buttons.

The background console shows a table of clusters:

NAME	PENDING	DATA SPACE %	LOAD	NETW	CLUST	SERVE
cluster1				✓	⊙	⊙
cluster2	Initializing...			⊙	?	?

Below the table is a "Details" section for "Cluster: cluster1" with the following information:

- Creation Date: Tue Mar 05 08:12:15 EST
- Username: postgres
- Size: 2gb
- Region: us-east-1
- Hardware: m1.small
- Engine Version: PostgreSQL 9.2.2 64bit
- Configuration: default
- Cluster healing mode:
 - Replace failed master with a new master
 - Replace failed master with existing master

Below the details are "Auto-Scaling Thresholds" and "Backup Settings".

Auto-Scaling Thresholds

- Auto-Scaling Thresholds
- % of Storage Size used: 65
- # of Server Connections: 95

Backup Settings

- Backup Window: 12:00am - 2:00am
- Backup Retention: 1

Horizontal scaling

- ▶ Scales out by adding more nodes to the load balanced cluster
- ▶ Manual scaling on demand
- ▶ Auto-scaling based on "read" connection numbers
- ▶ Can be performed live

Cluster: cluster1

Creation Date: Tue Mar 05 08:12:15 EST 2013

Username: postgres

Size: 2gb

Region: us-east-1

Hardware: m1.small

Engine Version: PostgreSQL 9.2.2 64bit

Configuration: default

Cluster healing mode:

- Replace failed master with a new master
- Replace failed master with existing replica

Auto-Scaling Thresholds

% of Storage Size used 65



of Server Connections 95



Horizontal scaling – how?

- ▶ Snapshot the data directory
- ▶ Initialise a new server instance
- ▶ Create a data directory from the snapshot
- ▶ Add the new instance to the cluster and "catch up"
- ▶ Add the new instance to the load balancer

Live Demo – Add a replica

The screenshot shows the EnterpriseDB PostgreSQL Plus Cloud Database management console. A 'Scale Up' dialog box is open, allowing the user to add replicas to a cluster named 'dev'. The dialog has two steps: 'Step 1' and 'Step 2'. Under 'Step 1', there are two sections: 'Add Replicas' and 'Add Storage'. The 'Add Replicas' section has a 'Number of Replicas' dropdown menu set to '1'. The 'Add Storage' section has a 'Number of Gigabytes' input field set to '0'. A 'Next' button is visible at the bottom right of the dialog.

The background shows a table of clusters with the following data:

NAME	PENDING	DATA SPACE %	LOAD	NETW	CLUS1	SERVE
dev		45%	0.0	✓	✓	✓
hr	Saving Snapshots of attached volumes	10%	0.01	✓	✓	✓

Below the dialog, there is a table showing details for the 'dev' cluster:

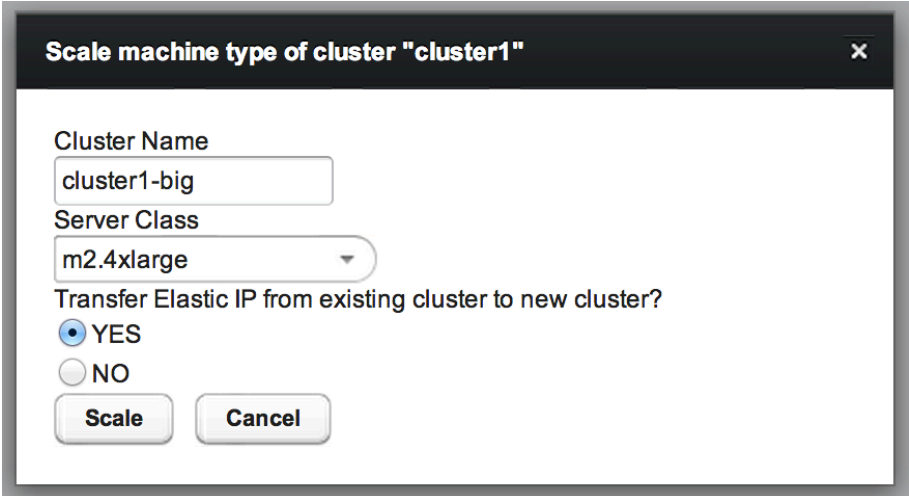
DBPORT	CONNEC	NETW	CLUS1	SERVE
5432	1	✓	✓	✓
5432	1	✓	✓	✓

The 'Cluster: dev' details section shows the following information:

- Creation Date: Tue M
- Username: postgres
- Size: 3gb
- Region: us-east-1
- Hardware: m1.small
- Engine Version: PostgreSQL 9.2.2-0401
- Configuration: default
- Cluster healing mode:
 - Replace failed master with a new master
 - Replace failed master with existing replica

Vertical scaling

- ▶ Scales up (or down) by changing the machine type
- ▶ Manually initiated only
- ▶ Currently requires a maintenance window



The screenshot shows a dialog box titled "Scale machine type of cluster 'cluster1'". It contains the following fields and options:

- Cluster Name:** A text input field containing "cluster1-big".
- Server Class:** A dropdown menu showing "m2.4xlarge".
- Transfer Elastic IP from existing cluster to new cluster?:** Radio buttons for "YES" (selected) and "NO".
- Buttons:** "Scale" and "Cancel".

Vertical scaling – how?

- ▶ Snapshot the current cluster
- ▶ Duplicates cluster servers onto new machine types
- ▶ Restores backup taken in step 1 to all nodes
- ▶ Starts the databases and replication
- ▶ Optionally switches the elastic IP over to the new cluster

Storage scaling

- ▶ Manual scaling of storage
- ▶ Automatic scaling when at N% used
- ▶ Can be performed live

Cluster: cluster1

Creation Date: Tue Mar 05 08:12:15 EST 2013

Username: postgres

Size: 2gb

Region: us-east-1

Hardware: m1.small

Engine Version: PostgreSQL 9.2.2 64bit

Configuration: default

Cluster healing mode:

- Replace failed master with a new master
- Replace failed master with existing replica

Auto-Scaling Thresholds

% of Storage Size used 65



of Server Connections 95



Storage scaling – how?

- ▶ New EBS volumes are created for each node
- ▶ New volumes are mounted and added to the data volume group
- ▶ Logical volume expanded to utilise additional space

Live Demo – Auto scaling

EnterpriseDB Postgres Plus Cloud Database

ec2- .compute-1.amazonaws.com:8181

EnterpriseDB® The Enterprise PostgreSQL Company

Postgres Plus Cloud Database

Log Out

Dashboard Clusters Backups User

NAME	PENDING	DATA SPACE %	LOAD	NETW	CLUS1	SERVI
hr	Scaling Up +1gb	41%	0.71	✓	✓	✓
dev		9%	0.0	✓	✓	✓

Details

Configurations

Monitoring

Time Range: Last Hour

Data Space

Connections

Load

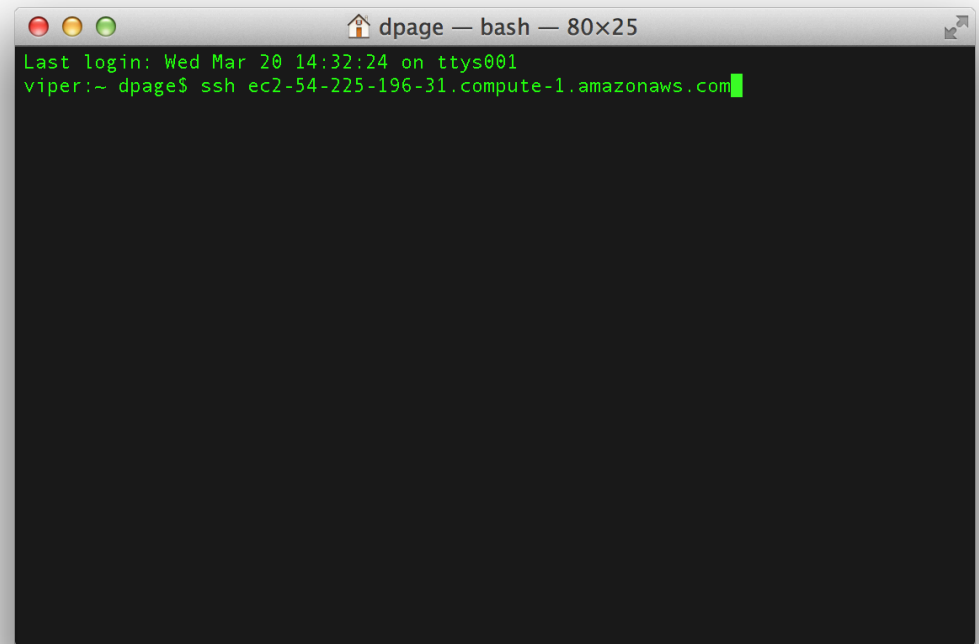
Events

Self healing

- ▶ Automatic replacement of failed replicas
- ▶ Automatic replacement of failed masters:
 - Promotes then replaces an existing replica OR
 - Creates a new master and brings it online

Extensible

- ▶ Shell access to cluster nodes
- ▶ Allows installation of extensions like PostGIS
- ▶ Push configuration changes to the pooler and database servers from the console



```
dpage — bash — 80x25
Last login: Wed Mar 20 14:32:24 on ttys001
viper:~ dpage$ ssh ec2-54-225-196-31.compute-1.amazonaws.com
```

Questions?

www.enterprisedb.com/cloud-database