



# Database on Demand Update

HEPiX, Amsterdam

`ignacio.coterillo@cern.ch`

17 Oct 2019

# Table of Contents

**Introduction**

**Service evolution**

**Future plans**

**Conclusions**

# We are here!

**Introduction**

Service evolution

Future plans

Conclusions

# The Database on Demand service

- ▶ Started in 2011, initially only targeting MySQL databases
- ▶ Self service portal with access to database monitoring, backup and PITR recovery operations and scheduling and configuration and upgrade management
- ▶ Currently over **820** databases for services from all CERN communities: OpenStack, Indico, Gitlab, LHCb compuring, COSMOS, eFILES, IT/WLCG Monitoring ...
  - ▶ MySQL (+500): **v5.6, v5.7, v8**
  - ▶ PostgreSQL (+180): **v9.6, v11**
  - ▶ InfluxDB (+130): **v1.7.7**

# The Database on Demand service

Actions

- 
- 
- 
- 
- 
- 
-

## Information for instance testinstance

Refresh instance

**State:** Running

**DB Type:** MySQL

**Creation Date:** 24/11/2011

**Category:** Test

**NO Connections:** 100

**Version:** 5.6.10

**Description:** Test instance until February edited.

**Username:** icoteril

**e-Group:** dbondemand-admin

**Expiry Date:** -

**Project:** DBOD

**DB Size:** 500 GB

**Shared instance:**

Job logs    Attributes log

Information for job

**State:** OK

**Creation Date:** 22/02/2013 15:24:51

**Requester:** dgomezbl

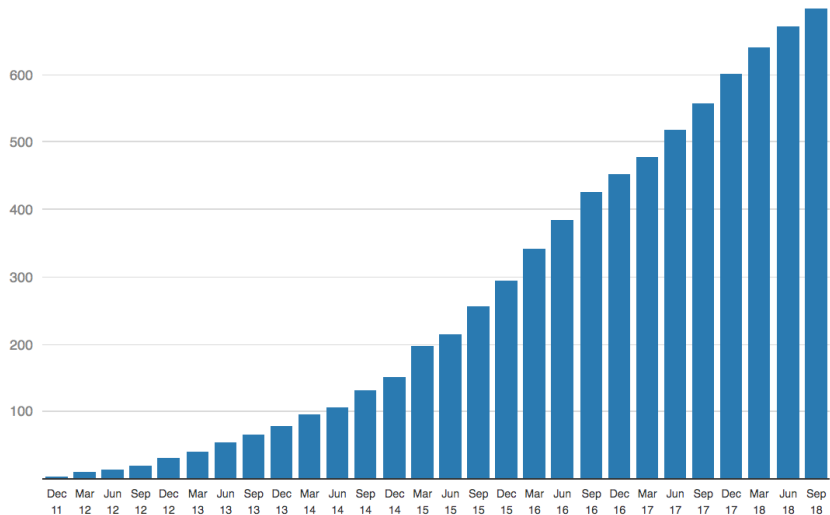
**Completion Date:** 22/02/2013 15:24:59

**Log:**

```
Fri Feb 22 15:24:55 CET 2013 : Main: Starting
Fri Feb 22 15:24:55 CET 2013 : Main: Loading entity variables
Fri Feb 22 15:24:55 CET 2013 : RunTime.RunStr running hostname
Fri Feb 22 15:24:55 CET 2013 : GeneralMysql.GetGeneralVars: host <[redacted]>
Fri Feb 22 15:24:55 CET 2013 : GeneralMysql.GetGeneralVars: port <[redacted]>
Fri Feb 22 15:24:55 CET 2013 : GeneralMysql.GetGeneralVars: socket <[redacted]>
Fri Feb 22 15:24:55 CET 2013 : RunTime.RetrievePasswordForUser: password found for <password_user_nastorag>
Fri Feb 22 15:24:55 CET 2013 : RunTime.RetrievePasswordForUser: password found for <password_db_dod_mysql>
Fri Feb 22 15:24:55 CET 2013 : RunTime.RetrievePasswordForUser: password found for <password_db_dod_dbmon>
Fri Feb 22 15:24:55 CET 2013 : GeneralMysql.GetGeneralVars: Base directory </usr/local/mysql/mysql-5.6.10>
Fri Feb 22 15:24:55 CET 2013 : GeneralMysql.GetGeneralVars: Binaries directory </usr/local/mysql/mysql-5.6.10/bin>
Fri Feb 22 15:24:55 CET 2013 : GeneralMysql.GetGeneralVars: Data directory </ORA/dbs03/TESTINSTANCE/mysql>
Fri Feb 22 15:24:55 CET 2013 : GeneralMysql.GetGeneralVars: Binary logs directory </ORA/dbs02/TESTINSTANCE/mysql>
Fri Feb 22 15:24:55 CET 2013 : Main: Instance is not CRS or local flag was specified.
Fri Feb 22 15:24:55 CET 2013 : RunTime.RunStr executing /usr/local/mysql-5.6.10/bin/mysqldadmin -u dod_mysql -pXXXXXXXXXX -socket <[redacted]>
```

# Database on Demand instances

Evolution of the amount of MySQL, PostgreSQL, and InfluxDB instances in the DBOD service



# Database on Demand platform

## Introduction

Set of tools and support systems for database infrastructure and operations automation

- ▶ DBOD API<sup>1</sup> (REST, PostgreSQL, Python (Tornado))
- ▶ **DBOD Daemon**<sup>2</sup>/Rundeck: Job scheduling, orchestration
- ▶ Puppet
- ▶ dbod-core: database and NetApp system interaction libraries

**Examples:** Instance creation in under 60 minutes. Database migration between servers in ~ 4 minutes

---

<sup>1</sup><https://github.com/orgs/cerndb>

<sup>2</sup>Deprecated



# We are here!

Introduction

**Service evolution**

Future plans

Conclusions

# Computing Infrastructure

## Service evolution

- ▶ Geneva (GPN)
  - ▶ 1270 VCPUs, 2.6TB RAM on VM's
  - ▶ 160 VCPUs, 300GB RAM on Critical VM's
  - ▶ 128 cores HT, 2TB RAM on Physical hosts
  - ▶ 448 cores HT, 2TB RAM on Physical hosts<sup>3</sup>
  - ▶ 64 cores HT, 3TB RAM on Physical hosts<sup>4</sup>
- ▶ Preveessin (TN): 128 cores HT, 4TB RAM
- ▶ 2nd Network Hub (GPN): 64 cores HT, 2TB RAM
- ▶ Wigner: 64 cores HT, 256 GB RAM

---

<sup>3</sup>de-commissioning

<sup>4</sup>commissioning

# Storage Infrastructure

## Service evolution

- ▶ Using **53TB** of **NFS** Volumes (NetApp)
- ▶ ~ **50TB EOS**. 2nd level backup, extended PITR support, will host full database backup solution

### New storage system

- ▶ All new instances using a new NetApp cluster since last week
- ▶ Older volumes will be migrated between now and somewhere 2020 Q1
- ▶ Volumes will be mounted using NFSv4
- ▶ Evaluating viability and benchmarking impact of **sec=krb5p**

# Time Series Databases

## Service evolution

### InfluxDB

- ▶ Introduced in 2016
- ▶ Today we host over 130 instances, (eg. IT/WLCG monitoring, COSMOS)

### TimescaleDB<sup>5</sup>

- ▶ A PostgreSQL extension
- ▶ Compatible with all PostgreSQL features: **replication**

---

<sup>5</sup>Work in progress

# Monitoring

## Service evolution

- ▶ During the first iteration of the platform used AppDynamics (\$\$)
- ▶ In 2017 moved to a custom solution based on:
  - ▶ **CollectD**, **Telegraf** for metrics collection
  - ▶ **Filebeat**, **Logstash** and **ElasticSearch** for logs
  - ▶ **InfluxDB** for metrics storage
  - ▶ **Grafana** for Dashboards and visualization

# Monitoring

## Service evolution



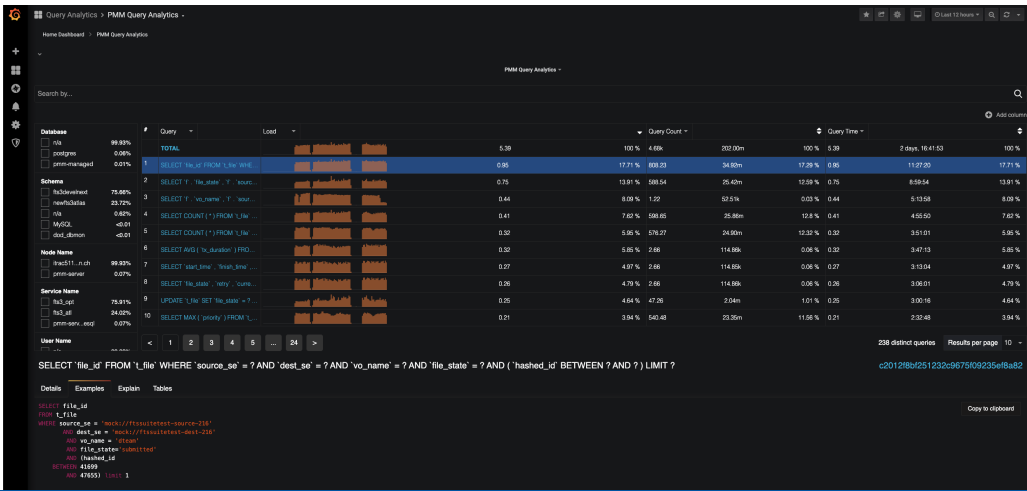
# Monitoring

## Service evolution

- ▶ Adding a performance focused layer based on **Percona Monitoring & Management (PMM)**
  - ▶ Agent runs locally on database hosts on user demand
  - ▶ PMM server running on Kubernetes/Nomad
  - ▶ Visualization layer available as Grafana App plugin

# Monitoring

## Service evolution





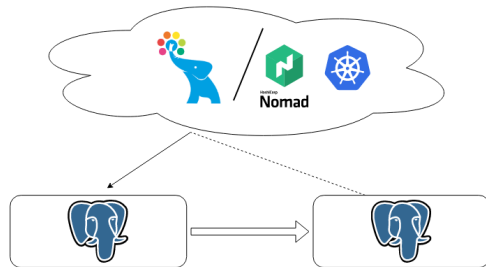
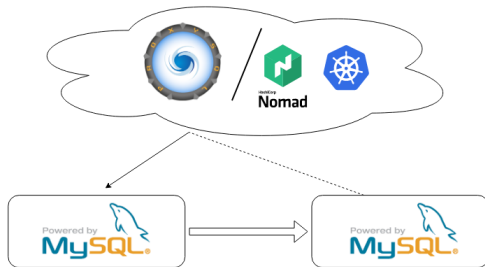
# High Availability

## Service evolution

- ▶ Replication supported for MySQL and PostgreSQL since 2015
- ▶ Implementing a HA layer with SQL layer proxies (**ProxySQL**, **pgpool-ii**) running on container clusters (Kubernetes/Nomad) with capabilities for connection routing, load balancing, caching and node switch-over capabilities
- ▶ Testing **MySQL Group Replication** (MySQL v8) clusters for highly critical services

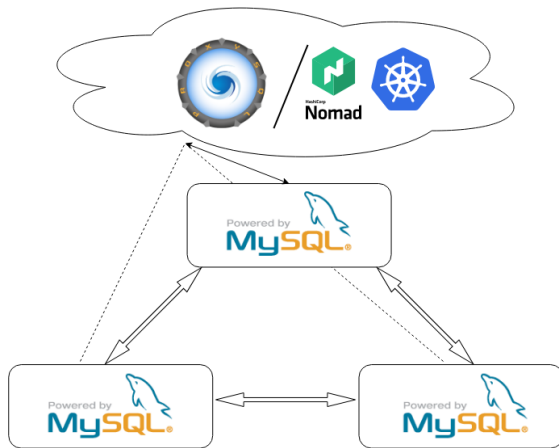
# High Availability

## Service evolution



# High Availability

## Service evolution



# We are here!

Introduction

Service evolution

**Future plans**

Conclusions

# Web Interface


## Future plans

- ▶ Implementation based on highly used technologies: NodeJS, Angular
- ▶ Consolidating software stack and with other group projects
- ▶ Reducing complexity of Web application
- ▶ Further information: <http://db-blog.web.cern.ch/blog/baptiste-legouix/2018-09-new-web-application-dbod-service>

# Web Interface

## Future plans

**RUNNING**

 **pinocho**  
IT-DB Ignacio Coterillo Coz

Owner <b>icoteril</b>	Egroup <b>dbondemand-admin</b>	Project <b>DBOD2</b>	Description <b>MySQL puppet test</b>
Creation date <b>2014-08-26T00:00:00</b>	Expiry date <b>2018-09-28T00:00:00</b>	Type <b>MYSQL</b>	Version <b>1.0.1</b>
Category <b>REF</b>	<input checked="" type="checkbox"/> <b>Notifications</b>		

**Logs** | [Jobs](#) | [Snapshots](#) | [File Editor](#) | [Metadata Editor](#)

If enabled, logs are provided in real time

↓

« « 1 2 3 4 5 6 7 8 9 10 ... 12878 » »

# Web Interface

## Future plans

The screenshot displays a web interface with the following elements:

- Navigation Tabs:** Logs (active), Jobs, Snapshots, File Editor, Metadata Editor.
- Logs Toggle:** A grey box with the text "If enabled, logs are provided in real time" and a toggle switch.
- Download Icon:** A downward arrow icon.
- Pagination:** A bar with buttons for ««, «, 1 (highlighted), 2, 3, 4, 5, 6, 7, 8, 9, 10, ..., 12878, », and »».
- Filters:** A text input field labeled "Filters".
- Items per page:** A dropdown menu labeled "Items per page".
- Statistics:** A blue button labeled "Statistics".
- Log Entry:** A single log entry: "Aug 29, 2018, 11:21:31 AM 2018-08-29T09:21:30.366770Z mysqld\_safe mysqld from pid file /ORA/dbs03/PINOCHO/mysql/db" with a dropdown arrow.

### Storage

Interest in investigating CEPHFS (performance, snapshot functionality) as an alternative to NetApp for the platform primary storage layer

### Configuration/Deployment

Re-explore use of Kubernetes for database server orchestration (after NFSv4/CEPHFS)



# We are here!

Introduction

Service evolution

Future plans

**Conclusions**

# Conclusions

- ▶ Next half year will be focus on infrastructure changes
- ▶ Will support HA solutions during the next months/year:
  1. Group Replication + Proxy layer
  2. MySQL replication + Proxy layer
  3. PostgreSQL Replication + Proxy layer
- ▶ Future work to reduce complexity and reduce costs while maintaining or improving level of service



[www.cern.ch](http://www.cern.ch)