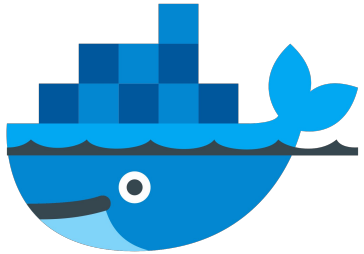
Готовим PostgreSQL в эпоху DevOps. Опыт 2ГИС

Павел Молявин, 2ГИС

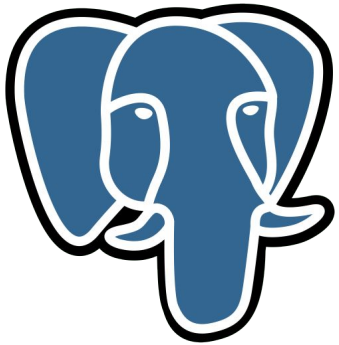
2GIS

- Отдел веб-разработки

- Отдел веб-разработки
- Infrastructure & Operations



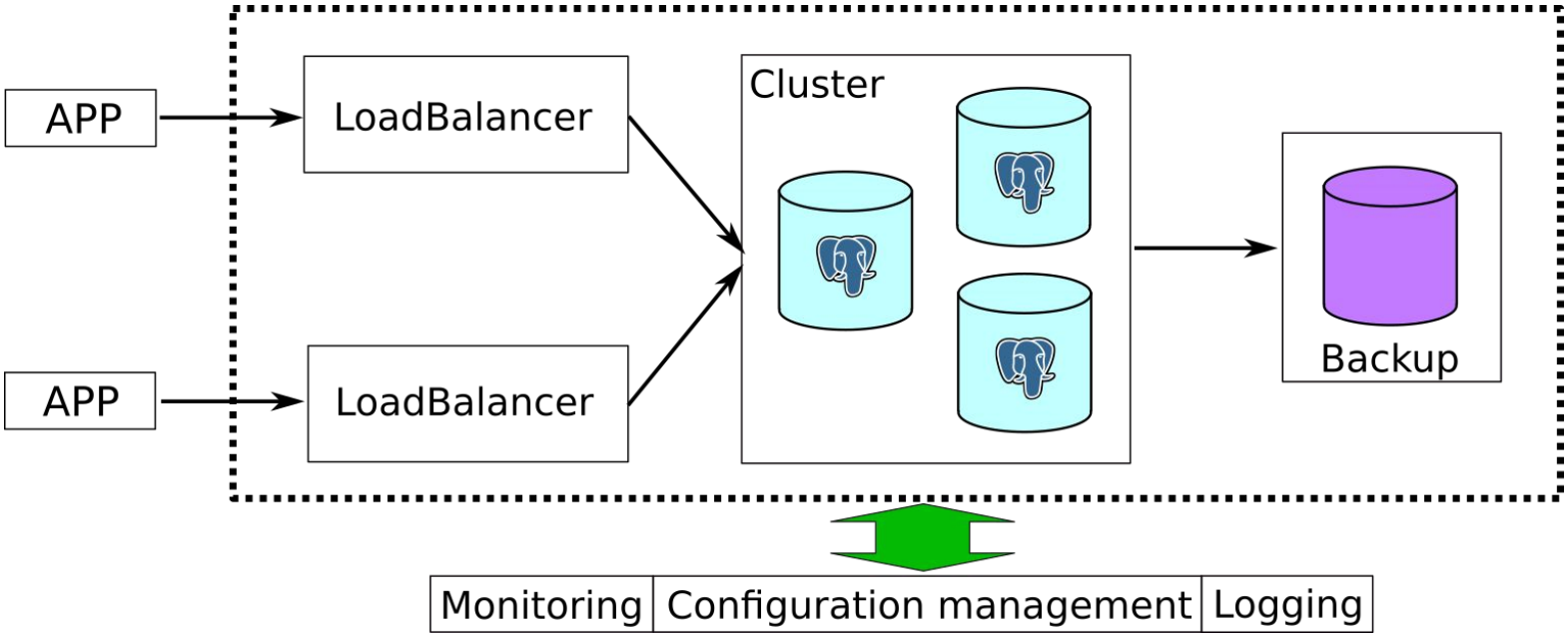

Stack



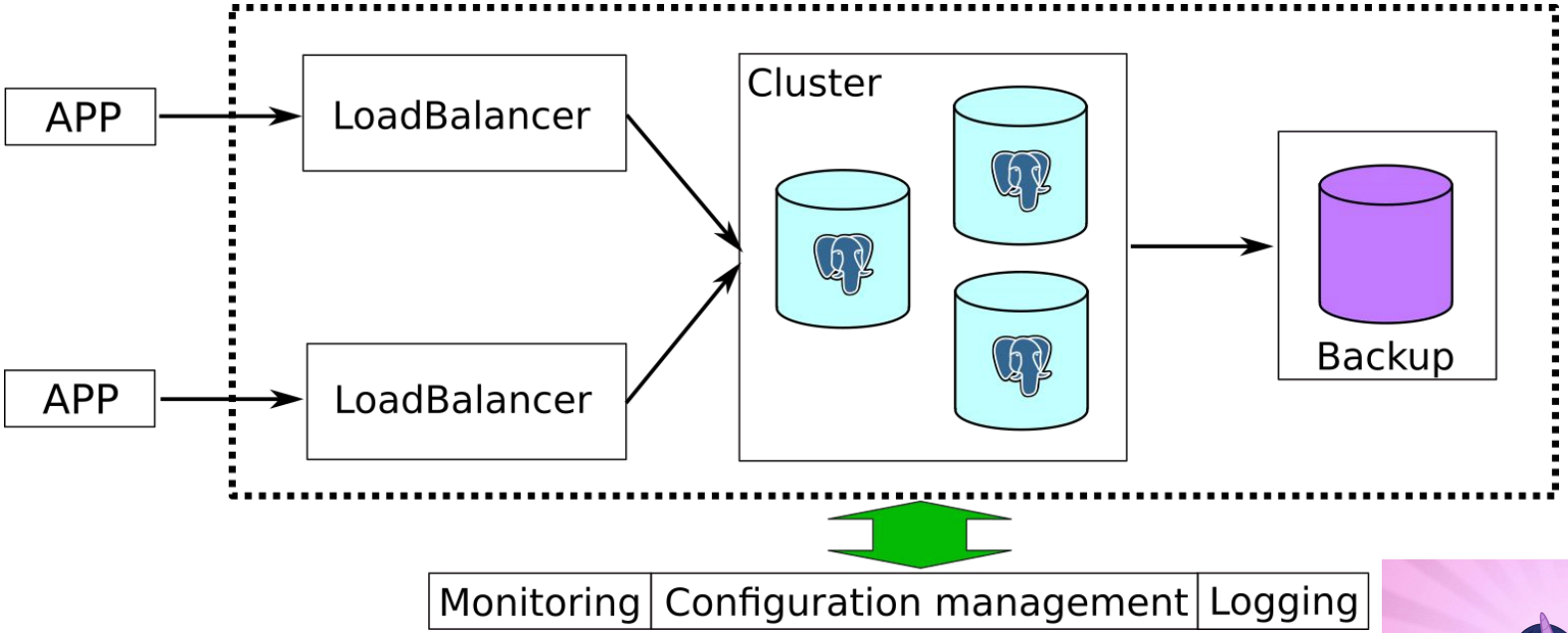
Командное селфи

Командное селфи

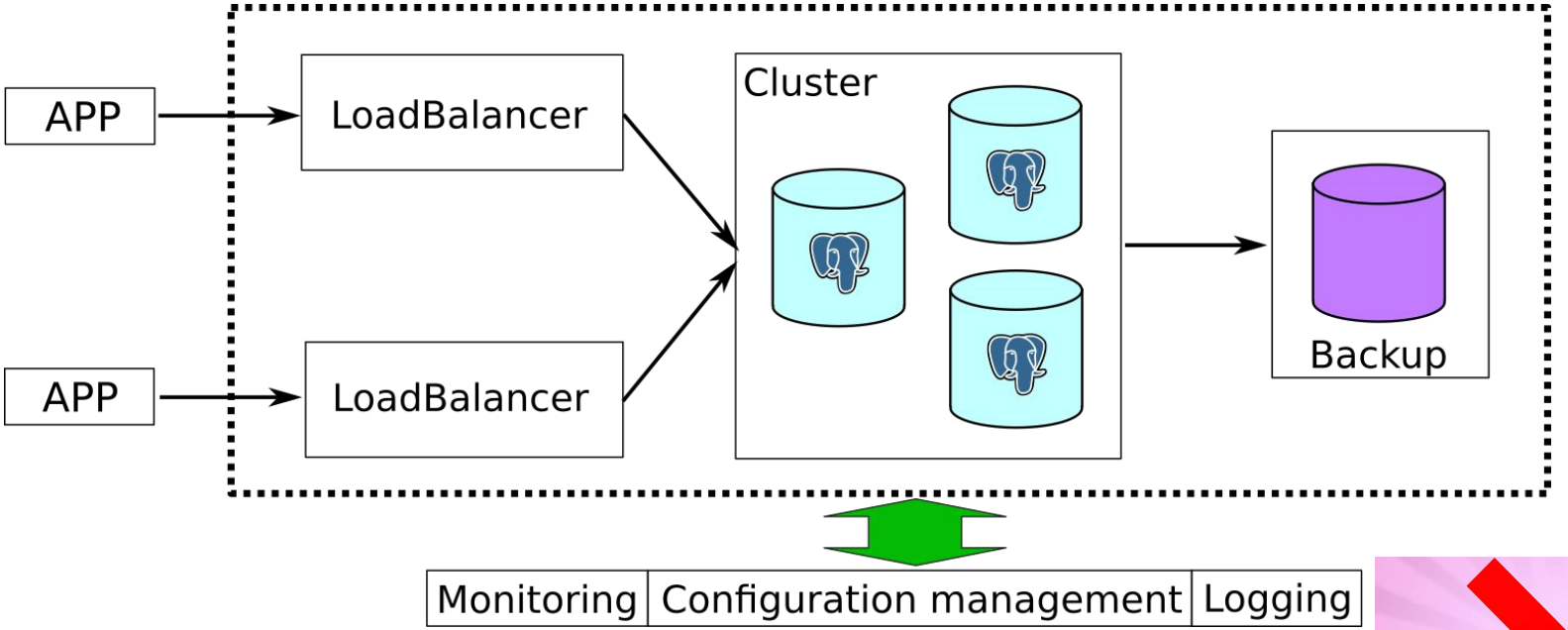
О чем это всё


APP
APP
LoadBalancer
LoadBalancer
Cluster
Backup
Monitoring
Configuration management
Logging

О чем это всё


APP
APP
LoadBalancer
LoadBalancer
Cluster
Backup
Monitoring
Configuration management
Logging

О чем это всё


APP
APP
LoadBalancer
LoadBalancer
Cluster
Backup
Monitoring
Configuration management
Logging

Постановка задачи

Постановка задачи

Постановка задачи

App

App

App

App

Постановка задачи

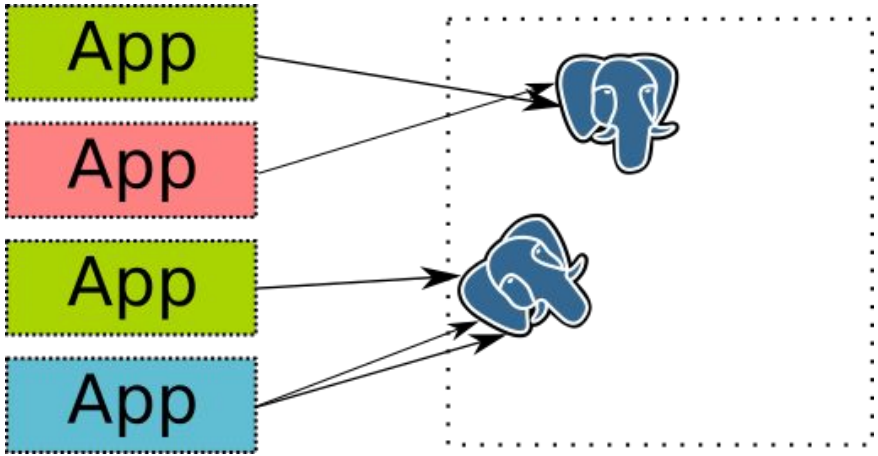
App

App

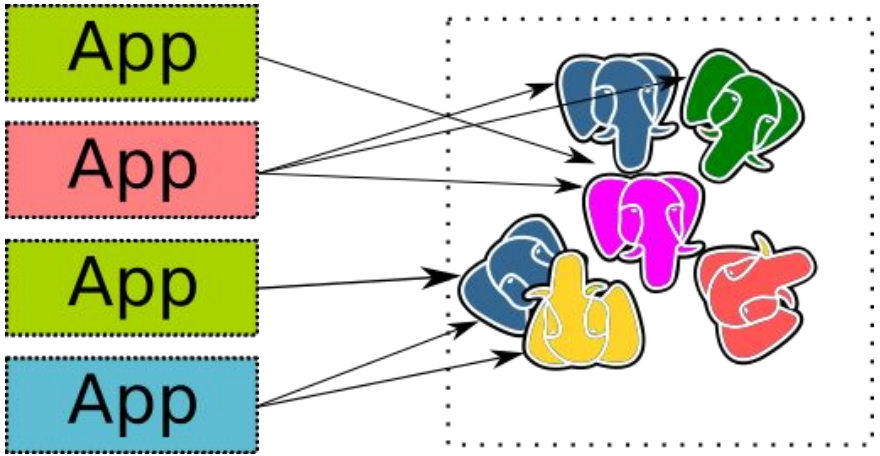
App

App

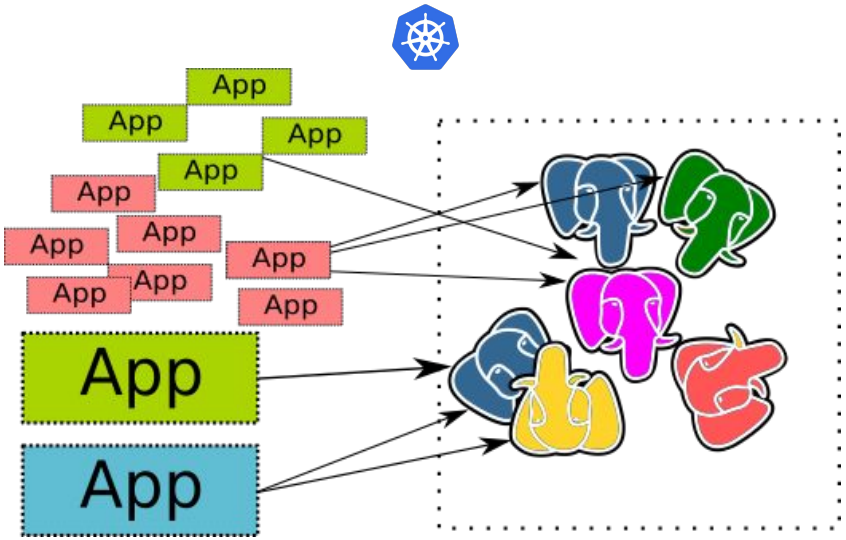
Постановка задачи



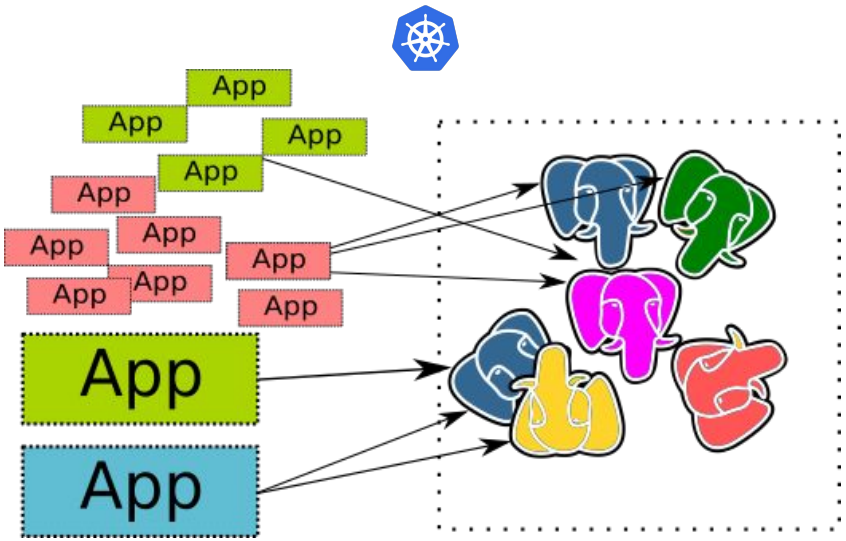
Постановка задачи



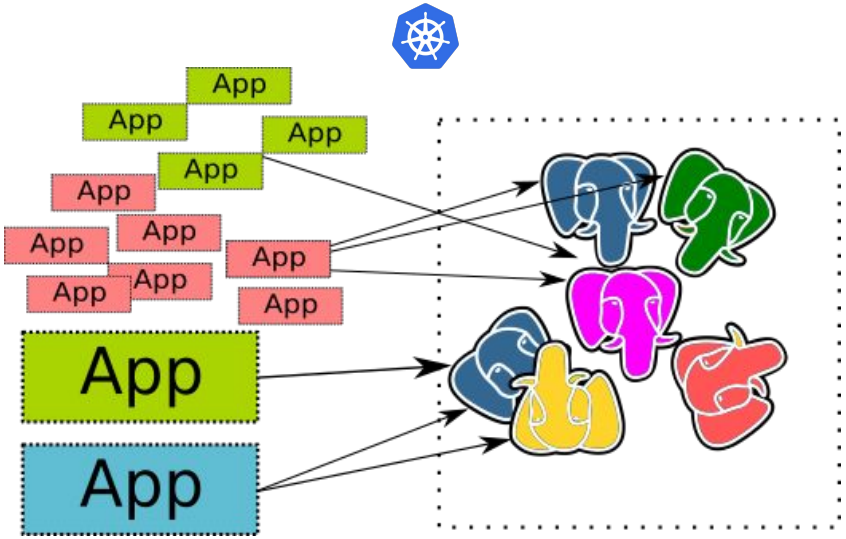
Постановка задачи



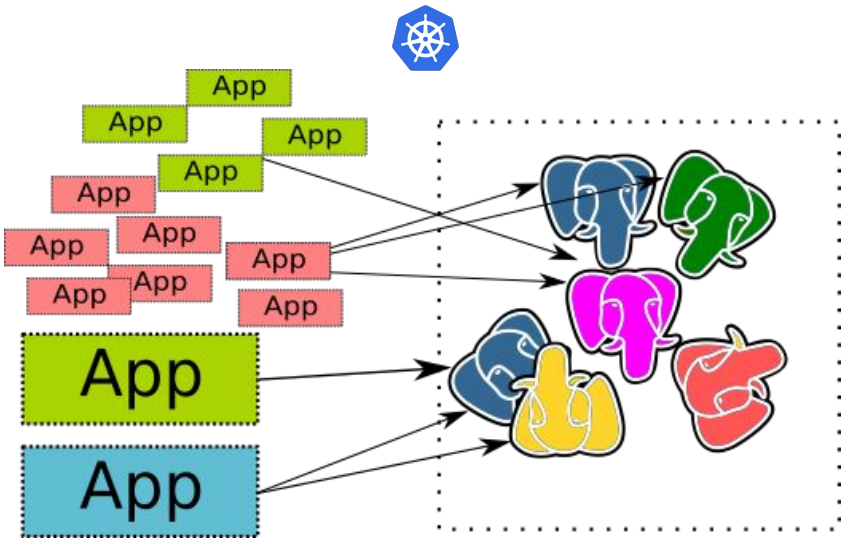
Постановка задачи



Постановка задачи



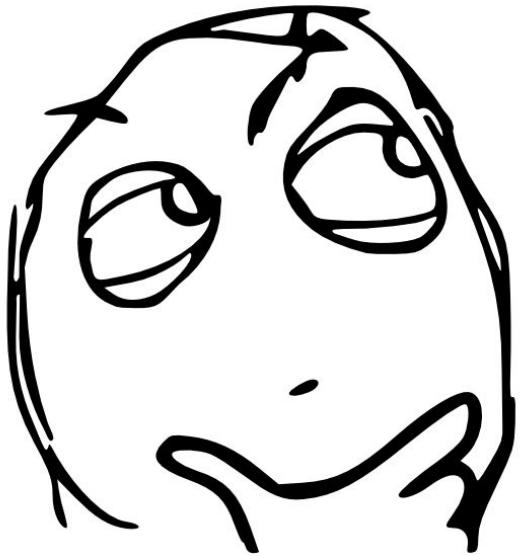
Постановка задачи



ANSIBLE

CHEF

Что мы решили сделать



Кластер

Кластер

Балансировка

Кластер

Балансировка

Резервные копии

Кластер

Балансировка

Резервные копии

Мониторинг и логи

Кластер

Балансировка

Резервные копии

Мониторинг и логи

Деплой

Кластер

Балансировка

Резервные копии

Мониторинг и логи

Деплой

CI-CD

Мы здесь

Кластер

Балансировка

Резервные копии

Мониторинг и логи

Деплой

CI-CD

- PostgreSQL 9.4 (9.6)

- PostgreSQL 9.4 (9.6)
- Потоковая репликация

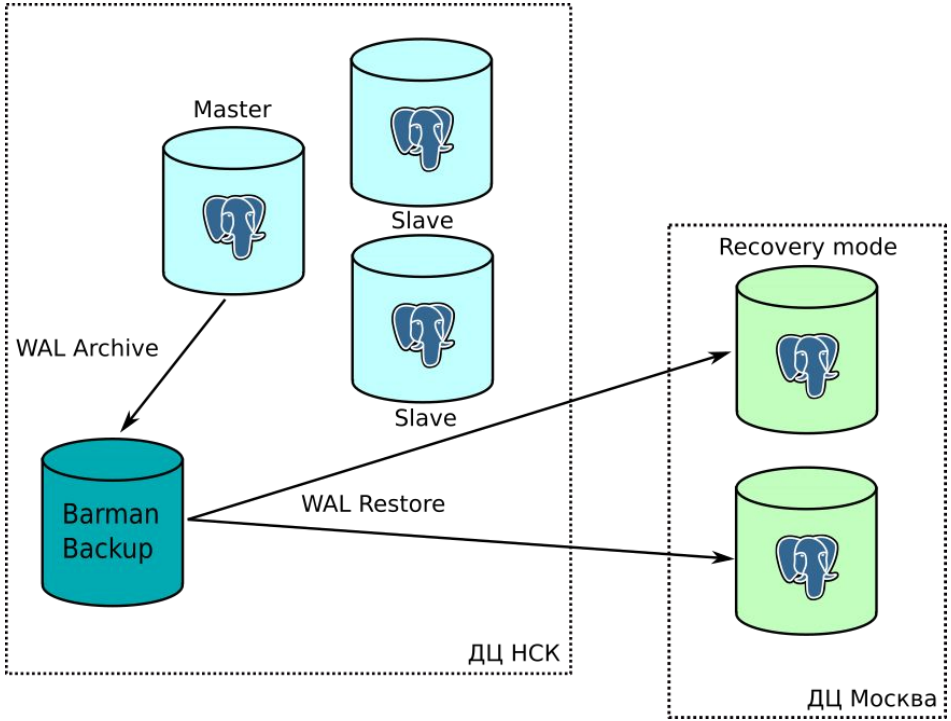
- PostgreSQL 9.4 (9.6)
- Потоковая репликация
- Кластеризация — repmgr от 2ndQuadrant

2ndQuadrant
PostgreSQL

- PostgreSQL 9.4 (9.6)
- Потоковая репликация
- Кластеризация — repmgr от 2ndQuadrant
- Кастомный failover_command

2ndQuadrant
PostgreSQL

- PostgreSQL 9.4 (9.6)
- Потоковая репликация
- Кластеризация — repmgr от 2ndQuadrant
- Кастомный failover_command
- RO реплика в другом ДЦ


Master
Slave
Slave
WAL Archive
Barman
Backup
WAL Restore
Recovery mode
ДЦ НСК
ДЦ Москва

- PostgreSQL 9.4 (9.6)
- Потоковая репликация
- Кластеризация — repmgr от 2ndQuadrant
- Кастомный failover_command
- RO реплика в другом ДЦ
- Проблемы

2ndQuadrant
PostgreSQL

Мы здесь

Кластер

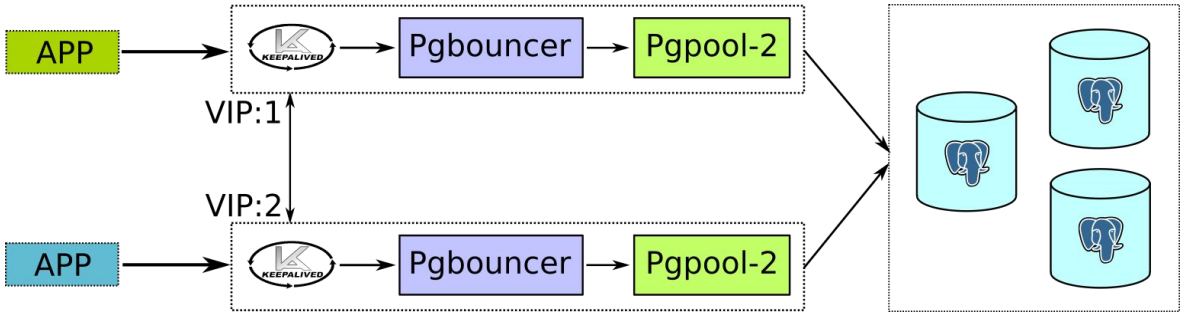
Балансировка

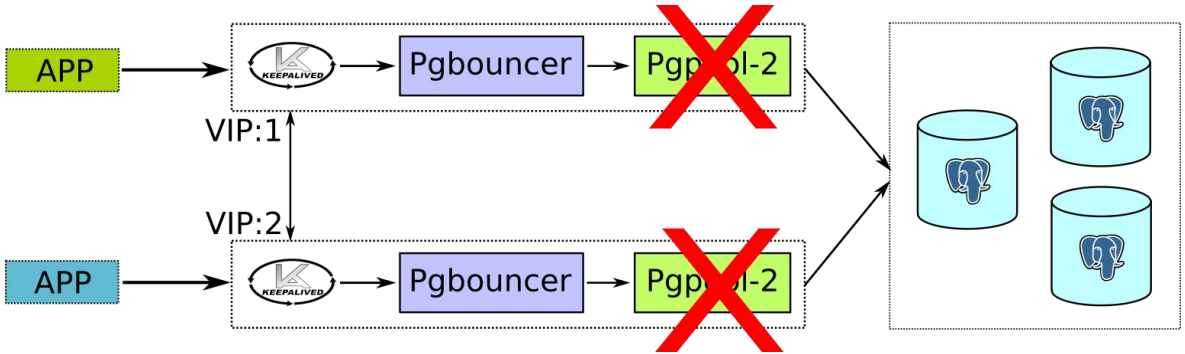
Резервные копии

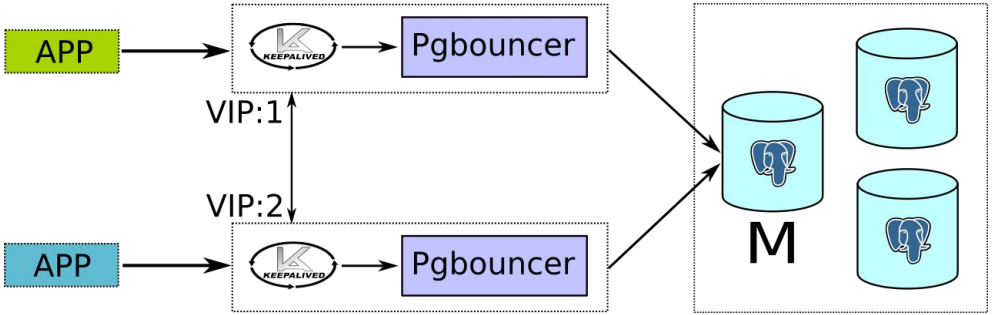
Мониторинг и логи

Деплой

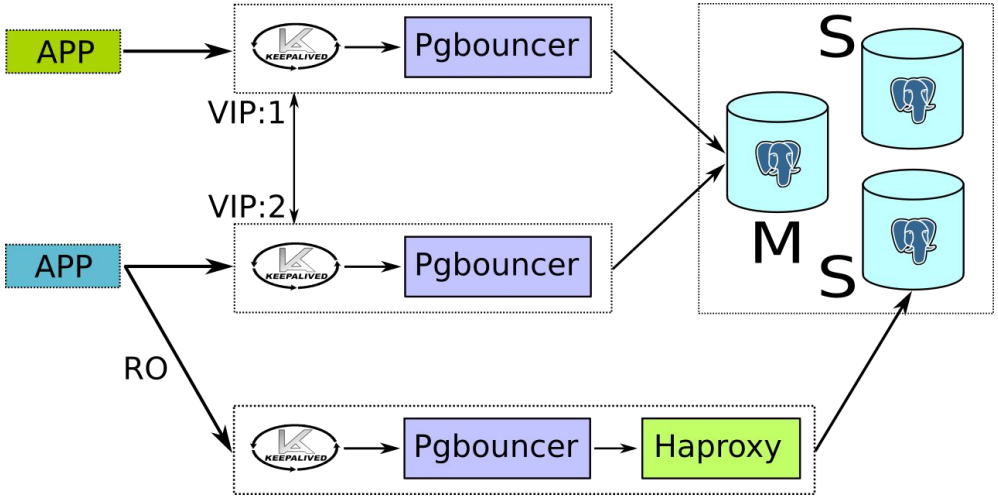
CI-CD


APP
KEEPALIVED
Pgbouncer
Pgpool-2
VIP:1
VIP:2
APP
KEEPALIVED
Pgbouncer
Pgpool-2


APP
KEEPALIVED
Pgbouncer
VIP:1
VIP:2
APP
KEEPALIVED
Pgbouncer



*Failover_command переключает бекенд


APP
KEEPALIVED
Pgbouncer
VIP:1
VIP:2
APP
KEEPALIVED
Pgbouncer
RO
KEEPALIVED
Pgbouncer
Haproxy
S
M
S

Мы здесь

Кластер

Балансировка

Резервные копии

Мониторинг и логи

Деплой

CI-CD

Barman


BaRMan
backup & recovery manager
for PostgreSQL


2ndQuadrant
PostgreSQL

Barman

- Base backup

Barman

- Base backup
- WAL Archiving для возможности PITR

Barman

- Base backup
- WAL Archiving для возможности PITR

Проверка backup-ов

Barman

- Base backup
- WAL Archiving для возможности PITR

Проверка backup-ов

Недостатки

- Медленный archive_command
- Много места

Кластер

Балансировка

Резервные копии

Мы здесь

Мониторинг и логи

Деплой

CI-CD

- Интеграция с Prometheus

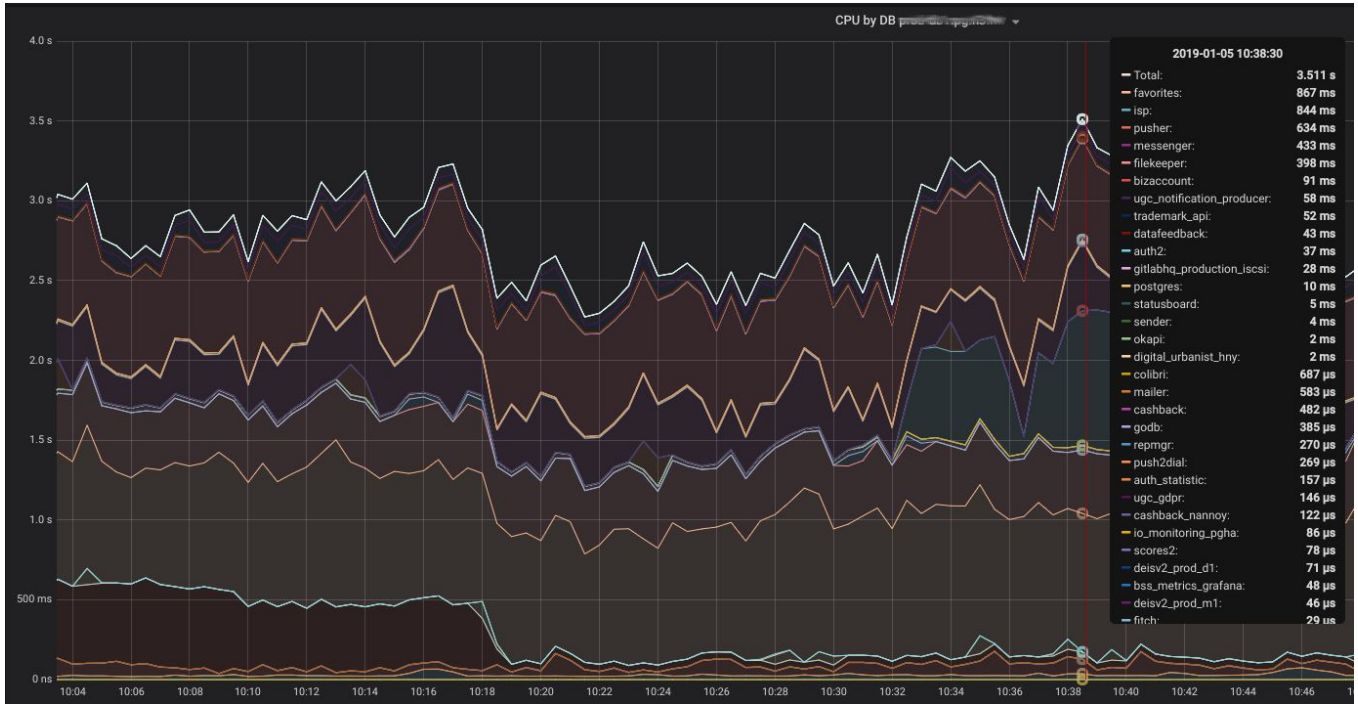
- Интеграция с Prometheus
- Golang Python Bash

- Интеграция с Prometheus
- Golang Python Bash
- Node, Postgres, Pgbouncer, Cgroups exporters

- Интеграция с Prometheus
- Golang Python Bash
- Node, Postgres, Pgbouncer, Cgroups exporters
- Системные метрики (CPU, RAM, IO)

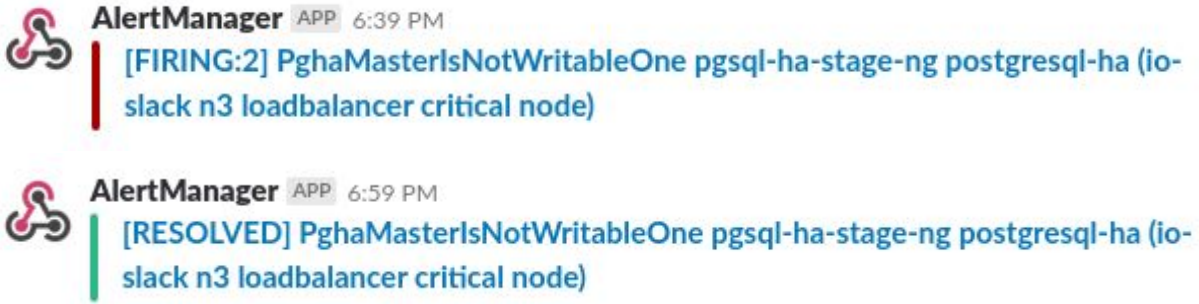
- Интеграция с Prometheus
- Golang Python Bash
- Node, Postgres, Pgbouncer, Cgroups exporters
- Системные метрики (CPU, RAM, IO)
- Метрики всех компонентов

Метрики в Grafana

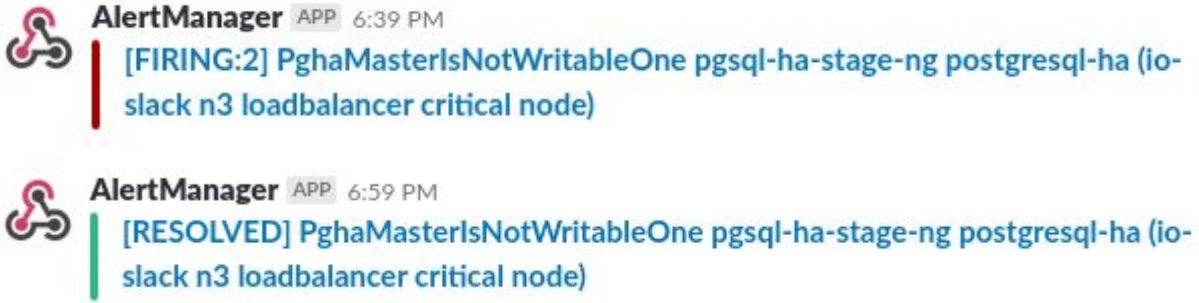


Alerts

Alerts

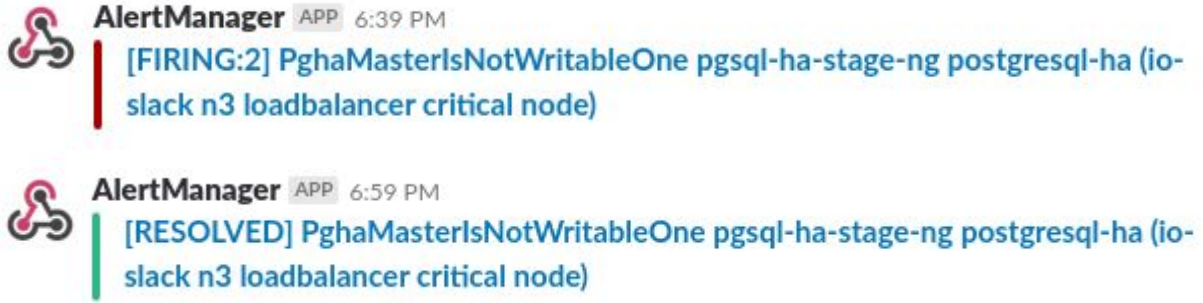

AlertManager APP 6:39 PM
[FIRING:2] PghaMasterIsNotWritableOne pgsql-ha-stage-ng postgresql-ha (io-slack n3 loadbalancer critical node)
AlertManager APP 6:59 PM
[RESOLVED] PghaMasterIsNotWritableOne pgsql-ha-stage-ng postgresql-ha (io-slack n3 loadbalancer critical node)

Alerts



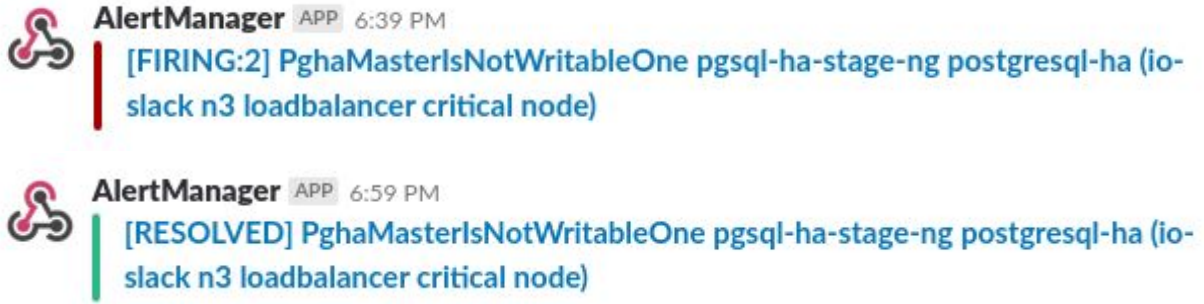
- Failover

Alerts



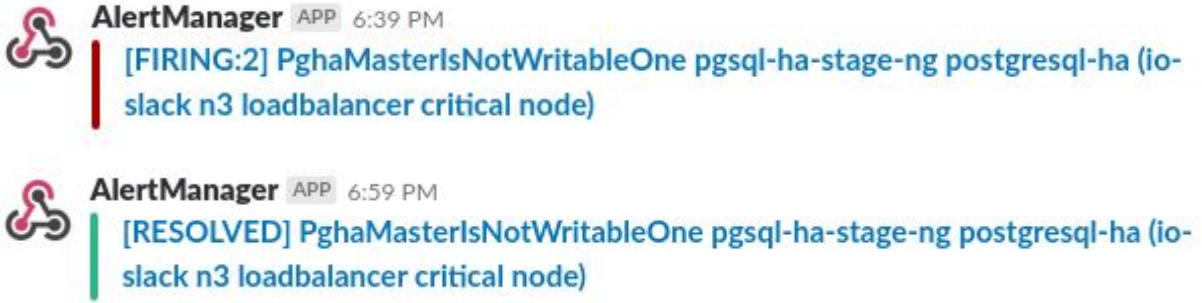
- Failover
- Pg_settings

Alerts



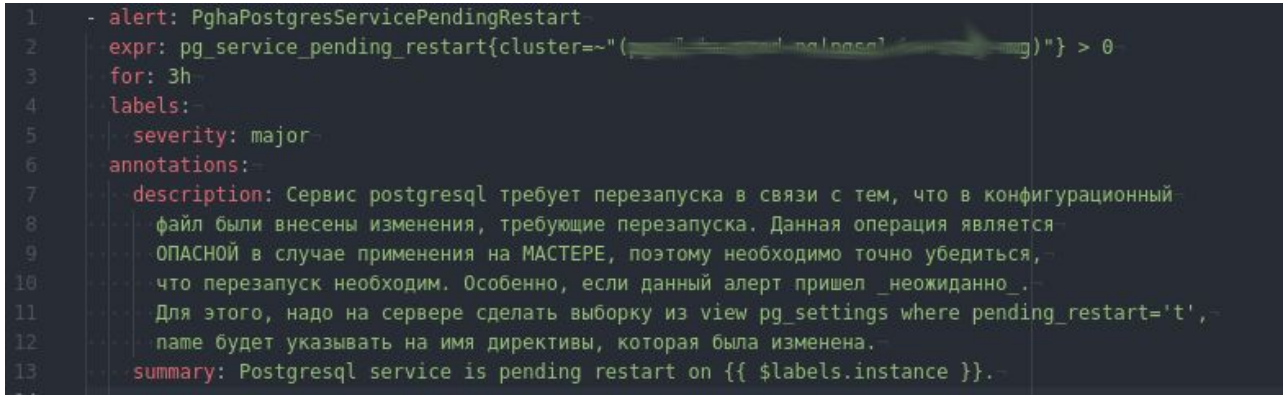
- Failover
- Pg_settings
- Лаг репликации

Alerts



- Failover
- Pg_settings
- Лаг репликации
- Тестовое восстановление

Alerts & Playbooks



```yaml
- alert: PghaPostgresServicePendingRestart
  expr: pg_service_pending_restart{cluster=~"([redacted])"} > 0
  for: 3h
  labels:
    severity: major
  annotations:
    description: Сервис postgresql требует перезапуска в связи с тем, что в конфигурационный
      файл были внесены изменения, требующие перезапуска. Данная операция является
      ОПАСНОЙ в случае применения на МАСТЕРЕ, поэтому необходимо точно убедиться,
      что перезапуск необходим. Особенно, если данный алерт пришел _неожиданно_.
      Для этого, надо на сервере сделать выборку из view pg_settings where pending_restart='t',
      name будет указывать на имя директивы, которая была изменена.
    summary: Postgresql service is pending restart on {{ $labels.instance }}.
```

Кластер

Балансировка

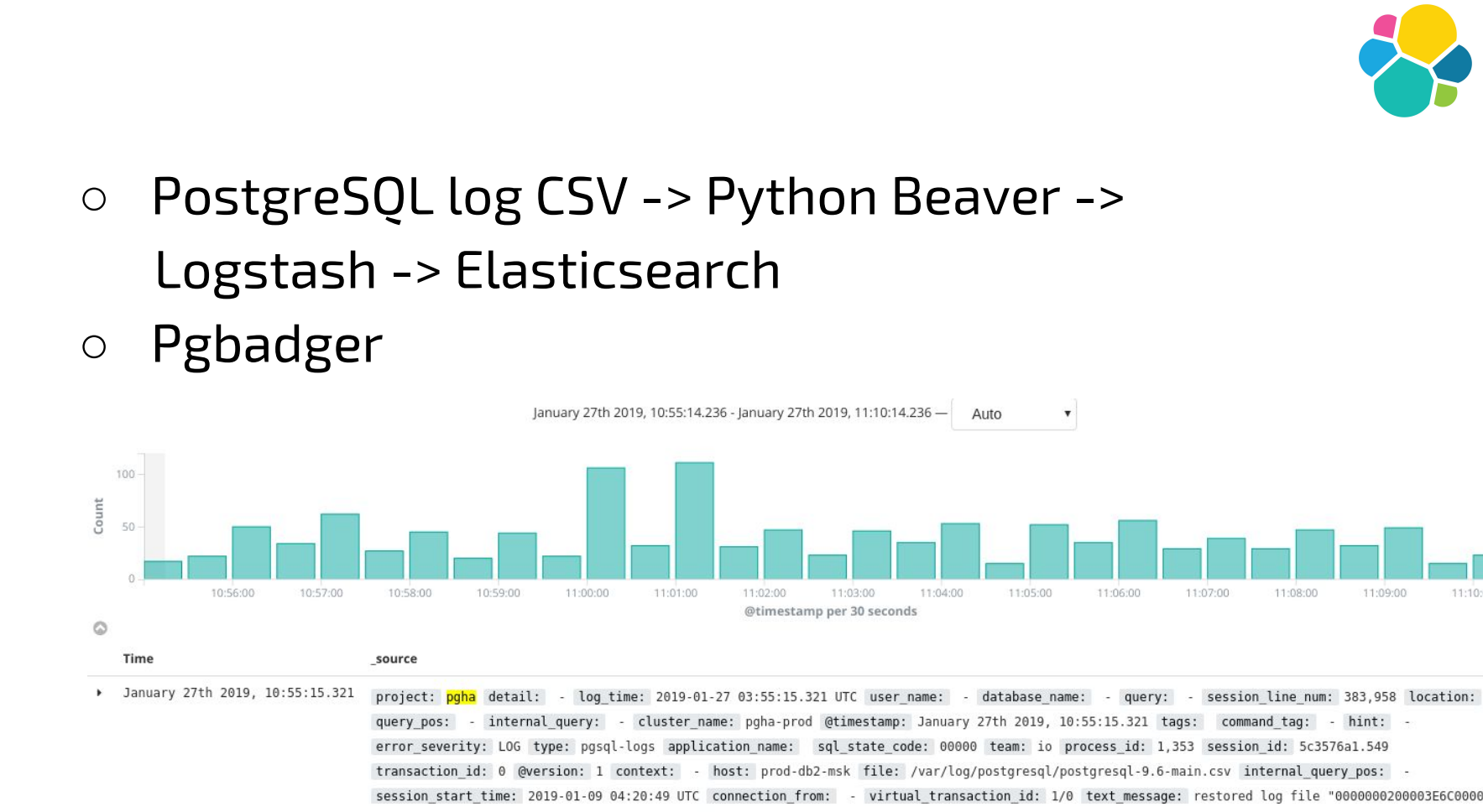
Резервные копии

Мы здесь

Мониторинг и логи

Деплой

CI-CD



- PostgreSQL log CSV -> Python Beaver -> Logstash -> Elasticsearch
- Pgbadger

Кластер

Балансировка

Резервные копии

Мы здесь

Мониторинг и логи

Деплой

CI-CD

- Ansible. 25 ролей. 20 самописных. 12 вырожденных

- Ansible. 25 ролей. 20 самописных. 12 вырожденных
- Environments

- Ansible. 25 ролей. 20 самописных. 12 вырожденных
- Environments
- Секреты

- Ansible. 25 ролей. 20 самописных. 12 вырожденных
- Environments
- Секреты
- Установка из apt

- Ansible. 25 ролей. 20 самописных. 12 вырожденных
- Environments
- Секреты
- Установка из apt
- Рутинные операции

- Ansible. 25 ролей. 20 самописных. 12 вырожденных
- Environments
- Секреты
- Установка из apt
- Рутинные операции
- Git

- Ansible. 25 ролей. 20 самописных. 12 вырожденных
- Environments
- Секреты
- Установка из apt
- Рутинные операции
- Git
- MR

- Ansible. 25 ролей. 20 самописных. 12 вырожденных
- Environments
- Секреты
- Установка из apt
- Рутинные операции
- Git
- MR
- Bootstrap для bare-metal, Openstack

Тесты

- Ansible 

Тесты


- Ansible 

- Запуск на тестовом окружении

Тесты


- Ansible 

- Запуск на тестовом окружении
- Проверяем
  - ⚒ Запуск кластера
  - ⚒ Failover
  - ⚒ Бэкап, восстановление

Кластер

Балансировка

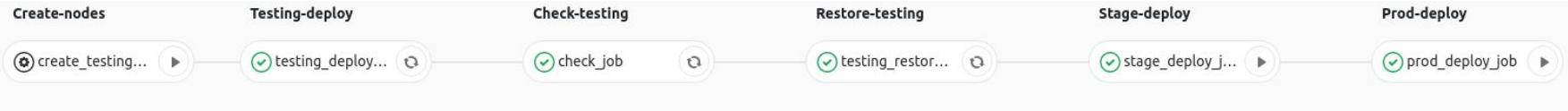
Резервные копии

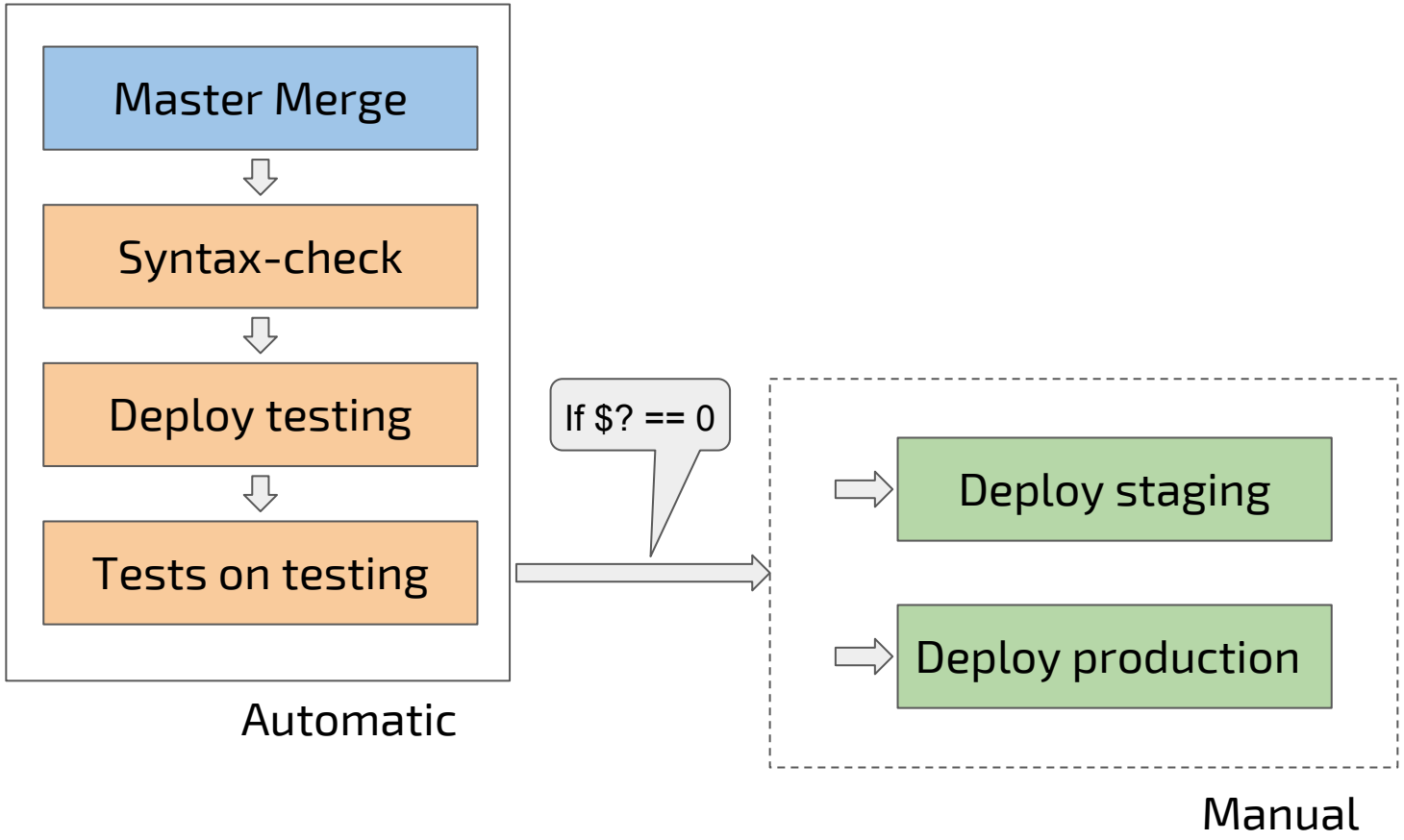
Мы здесь

Мониторинг и логи

Деплой

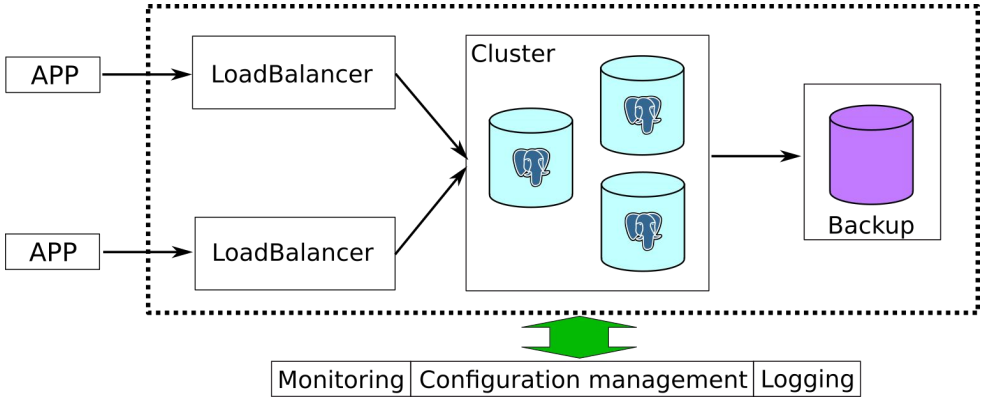
CI-CD


Create-nodes
create_testing...
Testing-deploy
testing_deploy...
Check-testing
check_job
Restore-testing
testing_restor...
Stage-deploy
stage_deploy_j...
Prod-deploy
prod_deploy_job

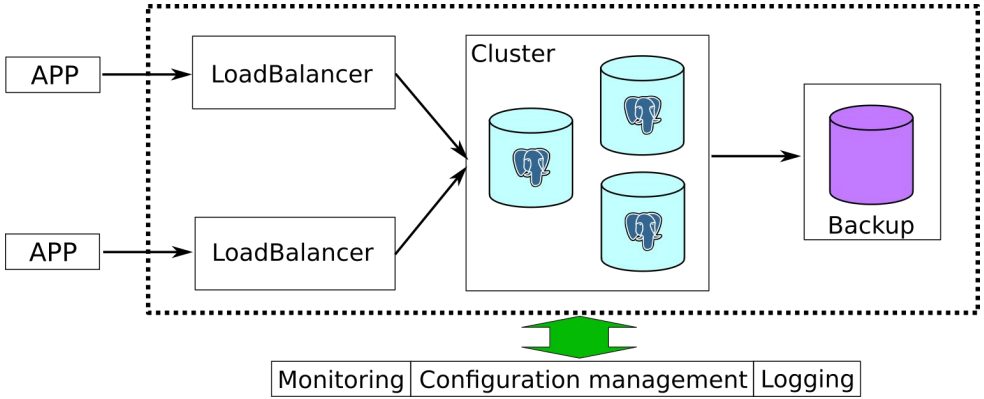

Master Merge
Syntax-check
Deploy testing
Tests on testing
Automatic
If $? == 0
Deploy staging
Deploy production
Manual

Что удалось

Что удалось


APP
APP
LoadBalancer
LoadBalancer
Cluster
Backup
Monitoring
Configuration management
Logging

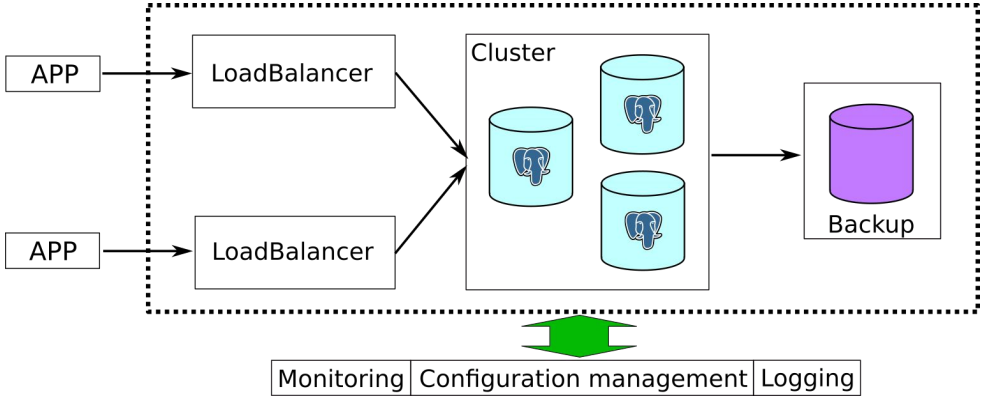
Что удалось


APP
APP
LoadBalancer
LoadBalancer
Cluster
Backup
Monitoring
Configuration management
Logging


COMPLETED
COMPLETED

Что удалось


15 мин.


APP
APP
LoadBalancer
LoadBalancer
Cluster
Backup
Monitoring
Configuration management
Logging


COMPLETED
COMPLETED

Что удалось

✓ Несколько кластеров

Что удалось

- ✓ Несколько кластеров
- ✓ --Поддержка командных СУБД

Что удалось

- ✓ Несколько кластеров
- ✓ --Поддержка командных СУБД
- ✓ XP++

Что удалось

- ✓ Несколько кластеров
- ✓ --Поддержка командных СУБД
- ✓ XP++
- ✓ Failover работает, мы проверяли, и спим спокойно по ночам :-)

Что не удалось

Что не удалось

✗ Сделать решение простым

Что не удалось

✗ Сделать решение простым

Что не удалось

- ✗ Сделать решение простым
- ✗ Приходится дорабатывать

Что не удалось

- ✗ Сделать решение простым
- ✗ Приходится дорабатывать
- ✗ Полностью избавиться от ручных операций

Что не удалось

- ✗ Сделать решение простым
- ✗ Приходится дорабатывать
- ✗ Полностью избавиться от ручных операций
- ✗ Обновление — отдельная боль

Что не удалось

- ✗ Сделать решение простым
- ✗ Приходится дорабатывать
- ✗ Полностью избавиться от ручных операций
- ✗ Обновление — отдельная боль
- ✗ Приходится самим собирать пакеты

Fork 0

Что не удалось

- ✗ Сделать решение простым
- ✗ Приходится дорабатывать
- ✗ Полностью избавиться от ручных операций
- ✗ Обновление — отдельная боль
- ✗ Приходится самим собирать пакеты
- ✗ Cattle vs. pet

Куда дальше?

Куда дальше?

- Stolon, Patroni 

Куда дальше?

- Stolon, Patroni 
- Решение без кластера

Куда дальше?

- Stolon, Patroni 
- Решение без кластера
- Апгрейд на PostgreSQL 10-11

Куда дальше?

- Stolon, Patroni
- Решение без кластера
- Апгрейд на PostgreSQL 10-11

Берегите свой Postgres!

p.molyavin@2gis.ru

Секретный уровень

- 3 кластера
- ~80 баз
- ~1,2 TB данных
- ~3895 xacts/s, 6310 queries/s
- ~3,5 млн аутентифицированных пользователей у сервисов в production
- ~10 млн анонимных пользователей

Типичный конфиг для сервера БД:

- 2 x XEON Gold 6134
- 128 GB RAM
- 4 x 900 GB SSD