Яндекс

Как справляться с ростом инфраструктуры сервиса

Игорь Андреев, системный администратор Я.Субботник, Новосибирск, 18 октября 2014

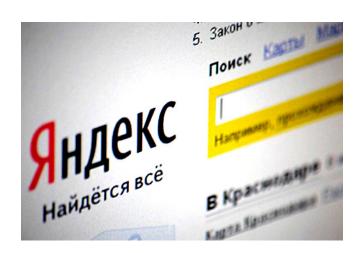
План доклада

Pазвитие инфраструктуры сервиса Salt Reclass Интеграция с другими системами Hadoop

Развитие инфраструктуры сервиса

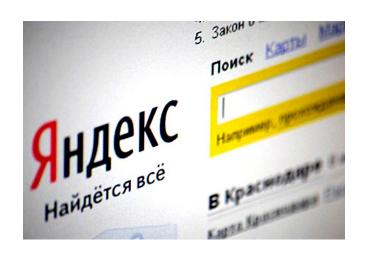


Как люди воспринимают сервис



Пользователи

Как люди воспринимают сервис

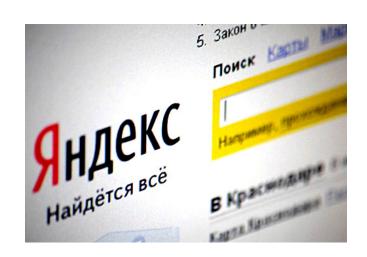




Пользователи

Разработчики

Как люди воспринимают сервис







Пользователи

Разработчики

Администраторы

Какой путь проходит сервис



Proof of concept

Какой путь проходит сервис

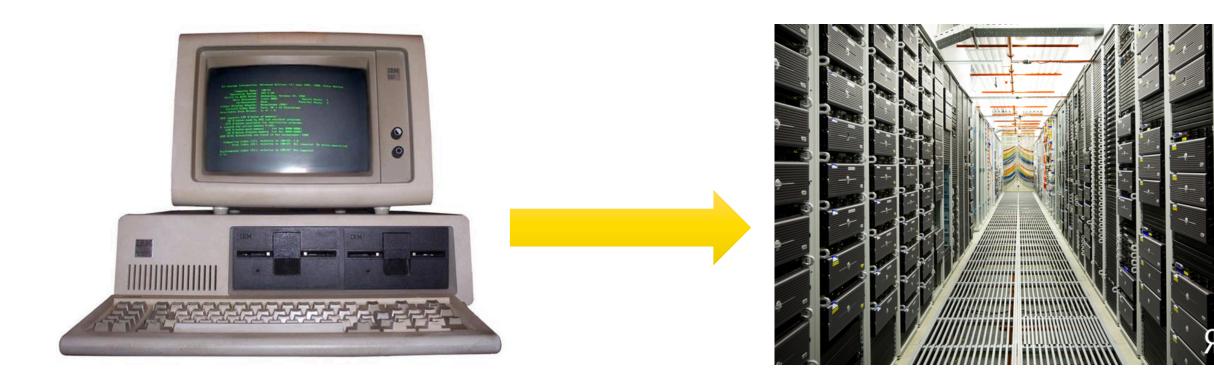


Proof of concept



Стабильная система

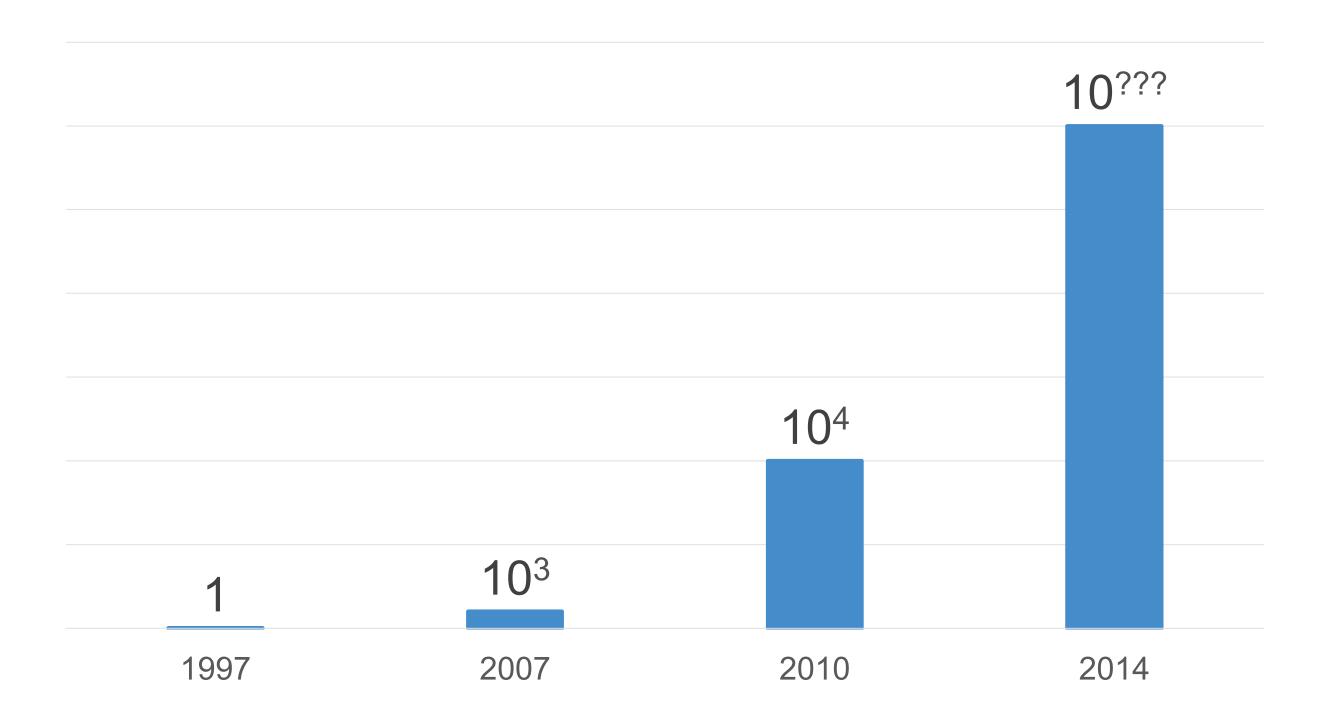
Какой путь проходит сервис



Proof of concept

Стабильная система

Масштабы роста



Задачи



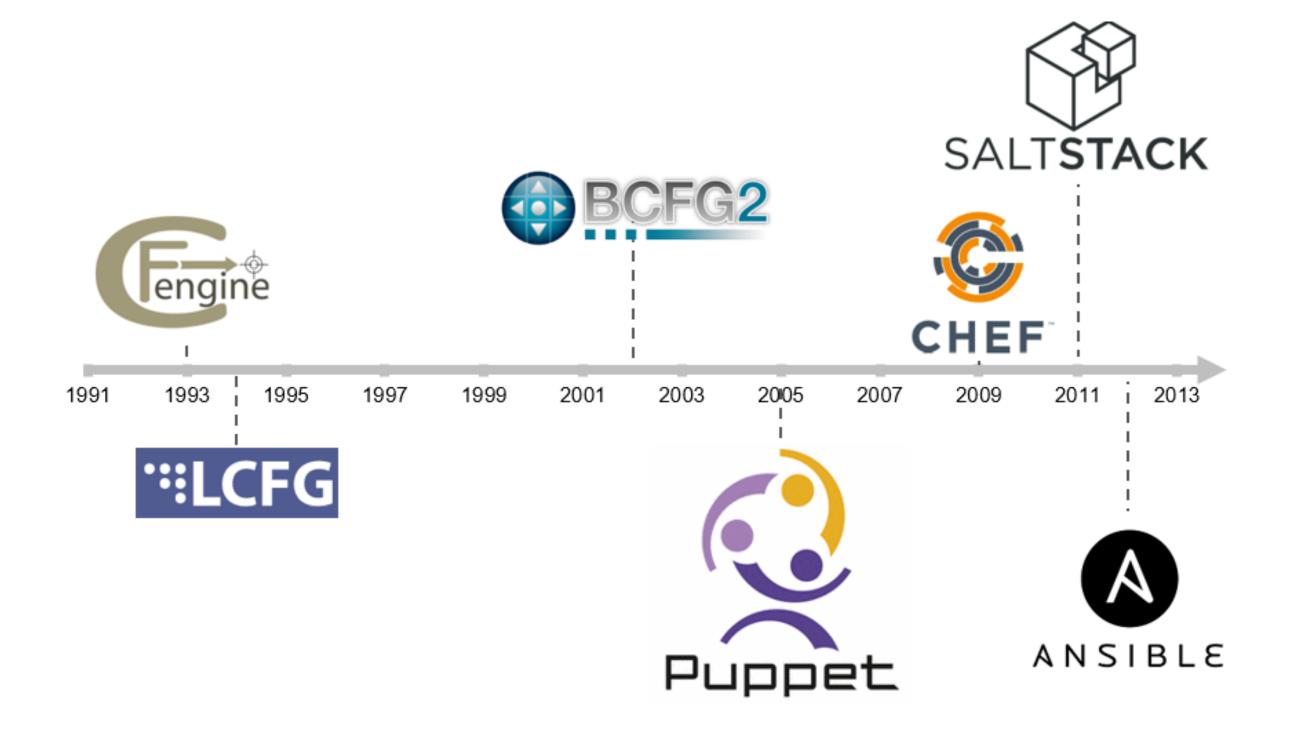
Мониторинг

- <u>Juggler</u>
- <u>Graphite</u>

Управление изменениями конфигурации

Зафиксировать изменения у разработчика Скопировать эти изменения Применить изменения на тестовую среду Перенести на production

Configuration management systems



Salt





Python

Yaml

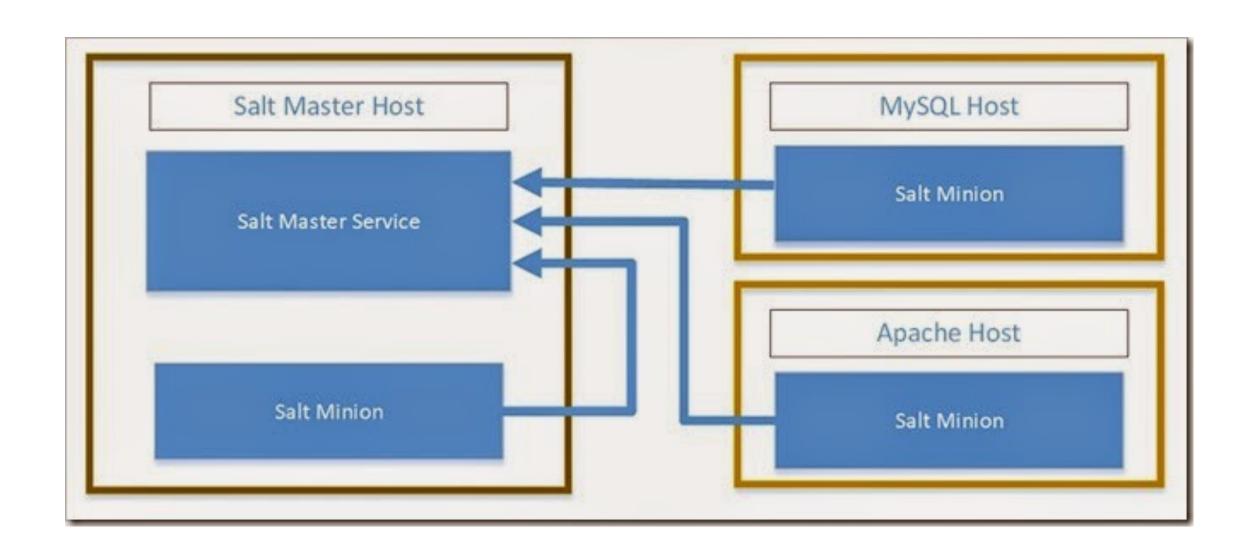
ZeroMQ

File server

Parallel execution

Top 10 on GitHub

Архитектура



Описание конфигурации

```
1 #top.sls
2 base:
3 'host':
      - timezone
1 #timezone.sls
2 tzdata:
      pkg:

    installed

5
  "echo Europe/Moscow > /etc/timezone":
      cmd.run:
8
         - require:

    pkg: tzdata
```

Файловая структура

```
pillar
    packages.sls
    top.sls
    salt
    top.sls
    web-server.sls
```

Grains

```
packages.sls
   top.sls
salt
   top.sls
   web-server.sls
server:
    "os:Debian":
       - match: grain
       - debian-servers
```

Grains

```
zmqversion
osfinger osfinger lsb_distrib_id num_gpus domain virtual oscodename ipv4 fqdn_ip6
              cpuarch fqdn_ip4 fqdn ps localhost
      osfullname kernel master shell
                                         cpu_model
        gpus osarch id server_id nodename
       cpu_flags saltversion saltpath
      mem_total hwaddr_interfaces path osrelease
     num_cpus ipv6 host
                                          serialnumber
        biosversion lsb_distrib_codename
                   lsb_distrib_release
     os_family
               saltversioninfo os
              biosreleasedate defaultencoding
                kernelrelease
                 lsb_distrib_description
                     pythonversion
                    productname
                    defaultlanguage
```

Pillar

```
packages.sls
top.sls
    salt
    top.sls
     — web-server.sls
1 #/srv/pillar/top.sls
2 {% if grains["os"] == "Debian" %}
3 web-server: apache2
4 {% elif %}
5 web-server: httpd
6 {% endif %}
```

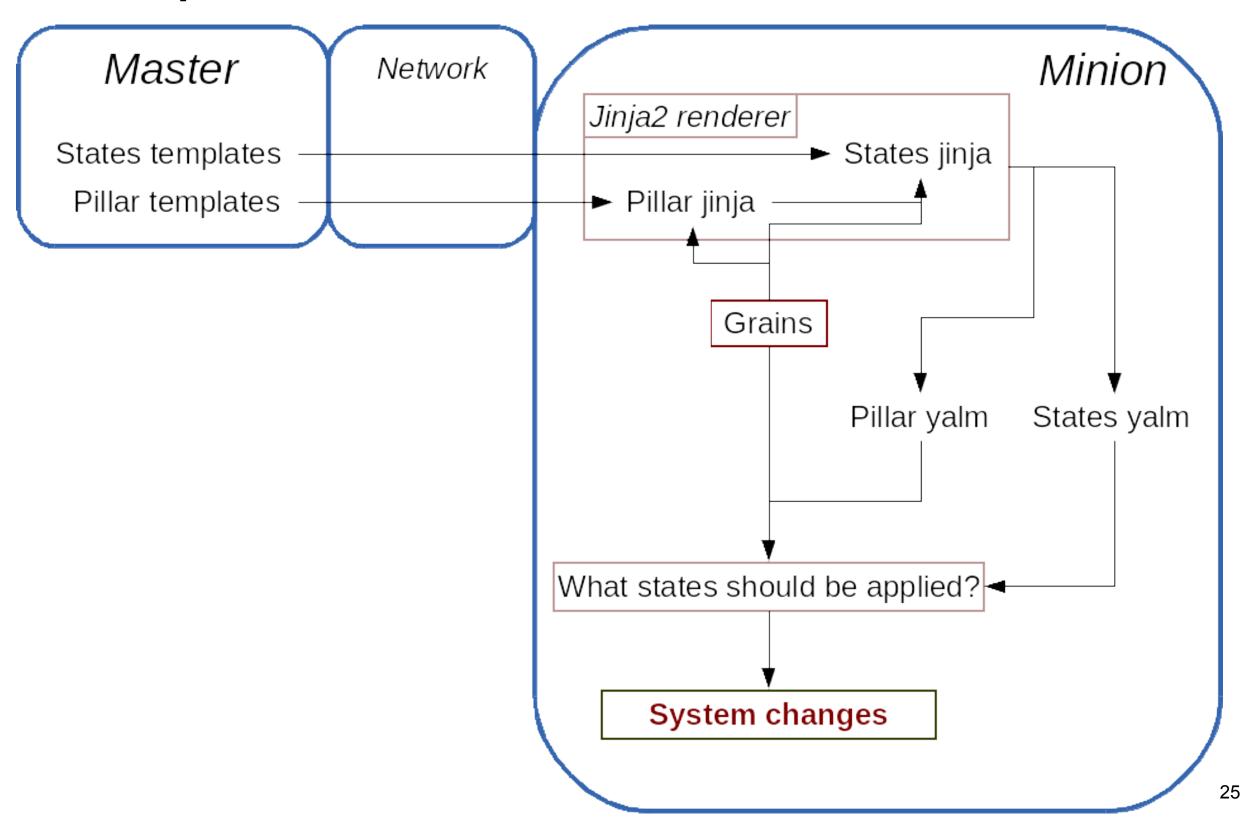
Pillar

```
1 #/srv/pillar/top.sls
2 {% if grains["os"] == "Debian" %}
3 web-server: apache2
4 {% elif %}
5 web-server: httpd
6 {% endif %}
1 #/srv/states/web-server.sls
2 web-server:
      pkg:

    installed

        - name: {{ pillar['web-server'] }}
```

Как применяются изменения



Недостатки

```
1 # top.sls
 2 base:
        'host1':
          system
 5
          - apache
        'host2':
          - system
          - nginx
 8
        'host3':

    frontend

10
        'host4':
11
12

    frontend
```

Reclass



Recursive external node classification

- http://reclass.pantsfullofunix.net
 - Node
 - > Class
 - > Application
 - > Parameters

Использование reclass c salt

```
inventory
    classes
        base.yml
        master.yml
       service.yaml -> ../../modules/service/_classes/service.yaml
   nodes
        master.hosts
        master.yml
        worker.hosts
     — worker.yml
modules
   service
         classes
         — service.yaml
        service
            └── service.cfg
            init.sls
states
└─ up.sh
```

Использование reclass c salt

```
inventory
    classes
       base.yml
       master.yml
       service.yaml -> ../../modules/service/_classes/service.yaml
   nodes
       master.hosts
                                 1 #worker.yml
       master.yml
       worker.hosts
                                 3 classes:
     — worker.yml
modules

    service.yaml

   service
                                 5 applications:
        classes
       └─ service.yaml
                                      worker
       service
                                 7 parameters:
                                      - test: true
           └── service.cfg
           init.sls
states
└─ up.sh
```

Использование reclass c salt

```
inventory
    classes
       base.yml
       master.yml
       service.yaml -> ../../modules/service/_classes/service.yaml
   nodes
       master.hosts
                                 1 #worker.hosts
       master.yml
                                 2 host1.example.com
       worker.hosts
       worker.yml
                                 3 host2.example.com
modules
   service
         — service.yaml
       service
             - service.cfg
           init.sls
states
└─ up.sh
```

Top.sls

```
1 # top.sls
2 base:
        'host1':
          - system
          - apache
        'host2':
          - system
          - nginx
        'host3':

    frontend

10
        'host4':
11
12

    frontend
```

Top.sls vs class.yml & host.yml

```
1 # top.sls
                                1 #frontend.yaml
   base:
                                3 classes:
         'host1':
                                    system
           - system
                                    - frontend

    apache

                                6 parameters:
                                    system:
 6
         'host2':
                                      os: debian
           - system
 8
             nginx
                                1 #frontend.hosts
         'host3':
                                2 host1
                                3 host2

    frontend

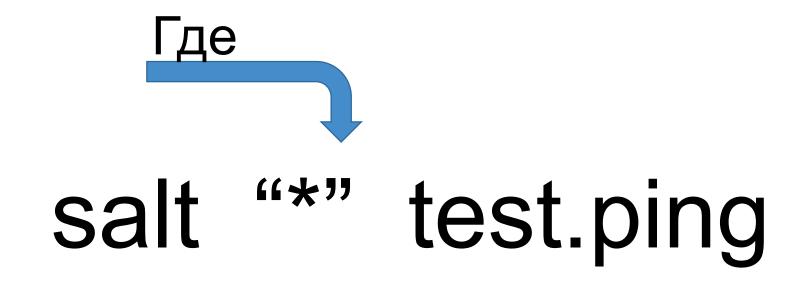
10
11
         'host4':
12

    frontend
```

Command Line

salt "*" test.ping

Command Line



Command Line

salt "*" test.ping

Command line

```
freebsd sysctl augeas of locale mod boto iam
              haproxyconn freebsdkmod alternatives glance ipset grains darwin sysctl boto secgroup.
      boto route 53 Idapmod ebuild linux acl eix
   event brew freebsdports cassandra chef layman bridge kmod gem logadm aliases junos key at debian ip linux sysctl disk deb apache daemontools launchetl environ dnsutil git logrotate cmdmod httpssswd aws sqs linux lym cron hosts data
   keystone chocolatey locate config dpkg boto elb djangomod archive hashutil ddns hadoop dig hashutil ddns hadoop groupadd glusterfs dnsmasq bluez apache freebsdservice extfs bsd shadow eselect aptpkg boto asg gentoolkitmod keyboard introspect hashutil ddns hadoop dig hadoop dig hadoop shadow eselect aptpkg blockdev influx etcd mod guestfs compact dockerio incron keyboard introspect
boto sqs img
                                      defaults gentoo service grub legacy
                             genesis cloud boto cloudwatch
```

Демонстрация



Интеграция с другими системами

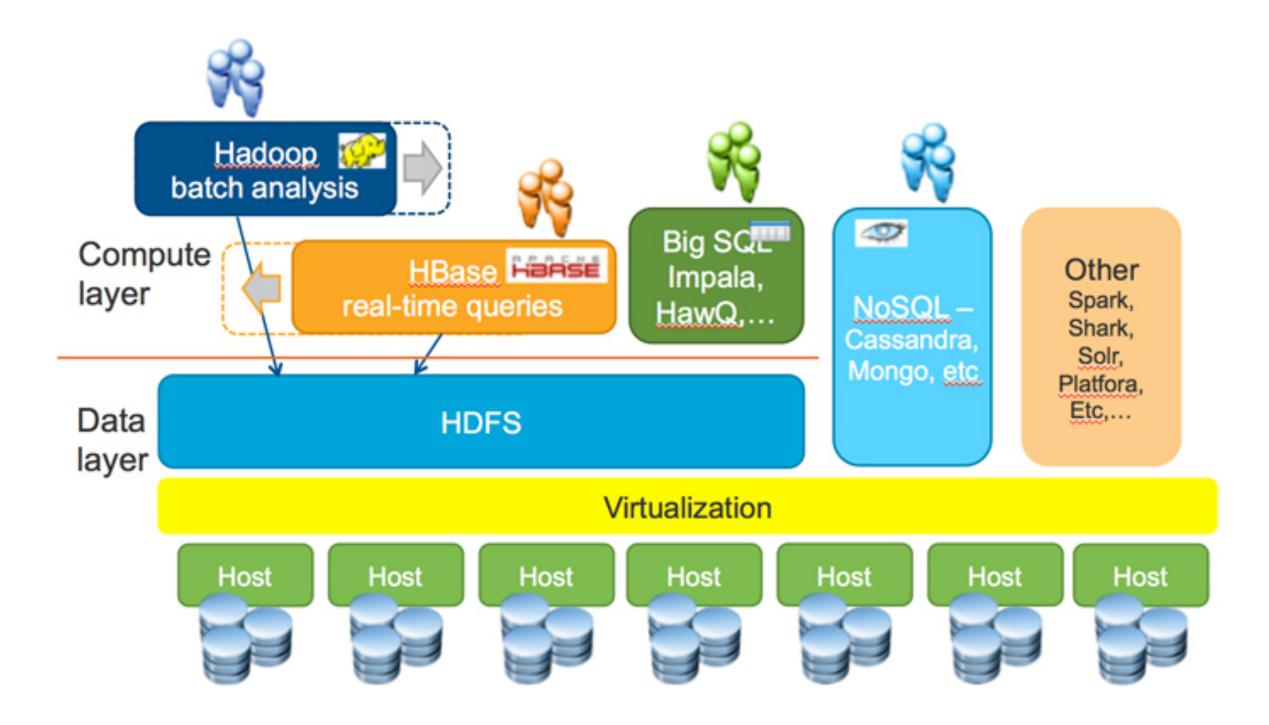
Возможности

Вся конфигурация в одном месте Экспорт в другие системы Запросы к reclass OpenStack





Hadoop



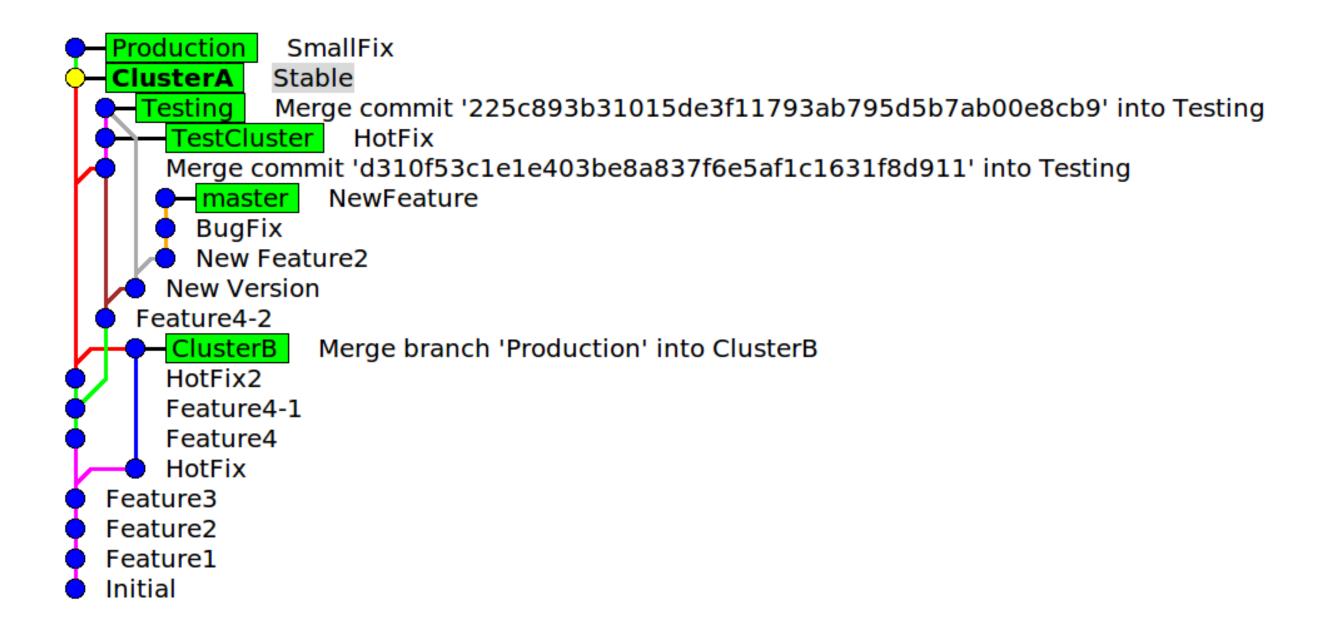
Сложность управления

Большое количество сервисов Сервисы зависят друг от друга Строгий порядок запуска и выключения

Управление кластерами

Вооtstrap
Несколько копий окружения
Переиспользование кода
Защита от выкладки старой версии
Open-source
Интеграция с инфраструктурой

Управление кластерами



Вопросы?

Контакты

Игорь Андреев системный администратор E-mail: iandreyev@yandex-team.ru