



Infrastructure as code на примере Ansible

Месропян Н. А.

nbw.adm+slurm@gmail.com



Pets or cattle?



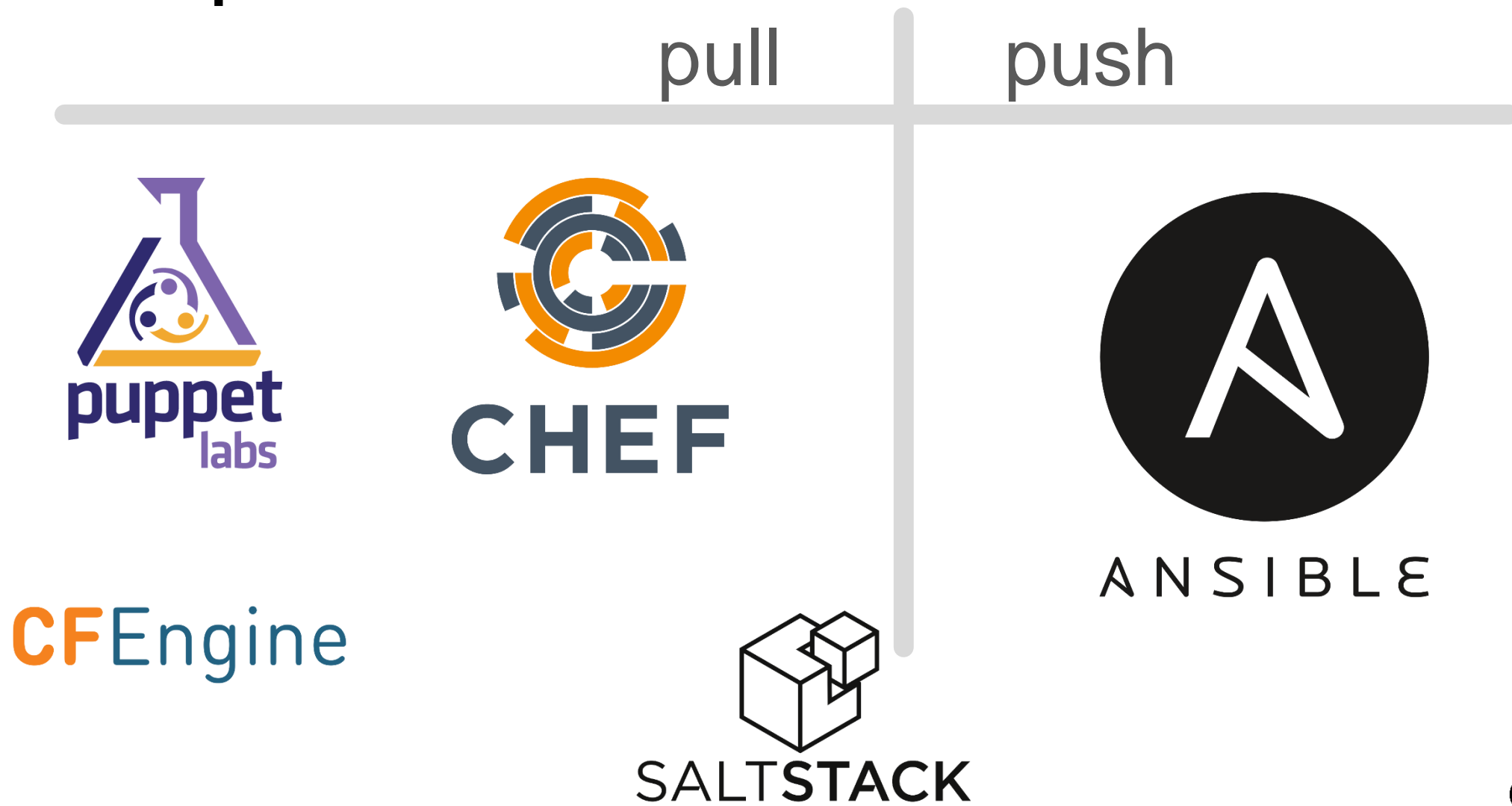


Pets → cattle



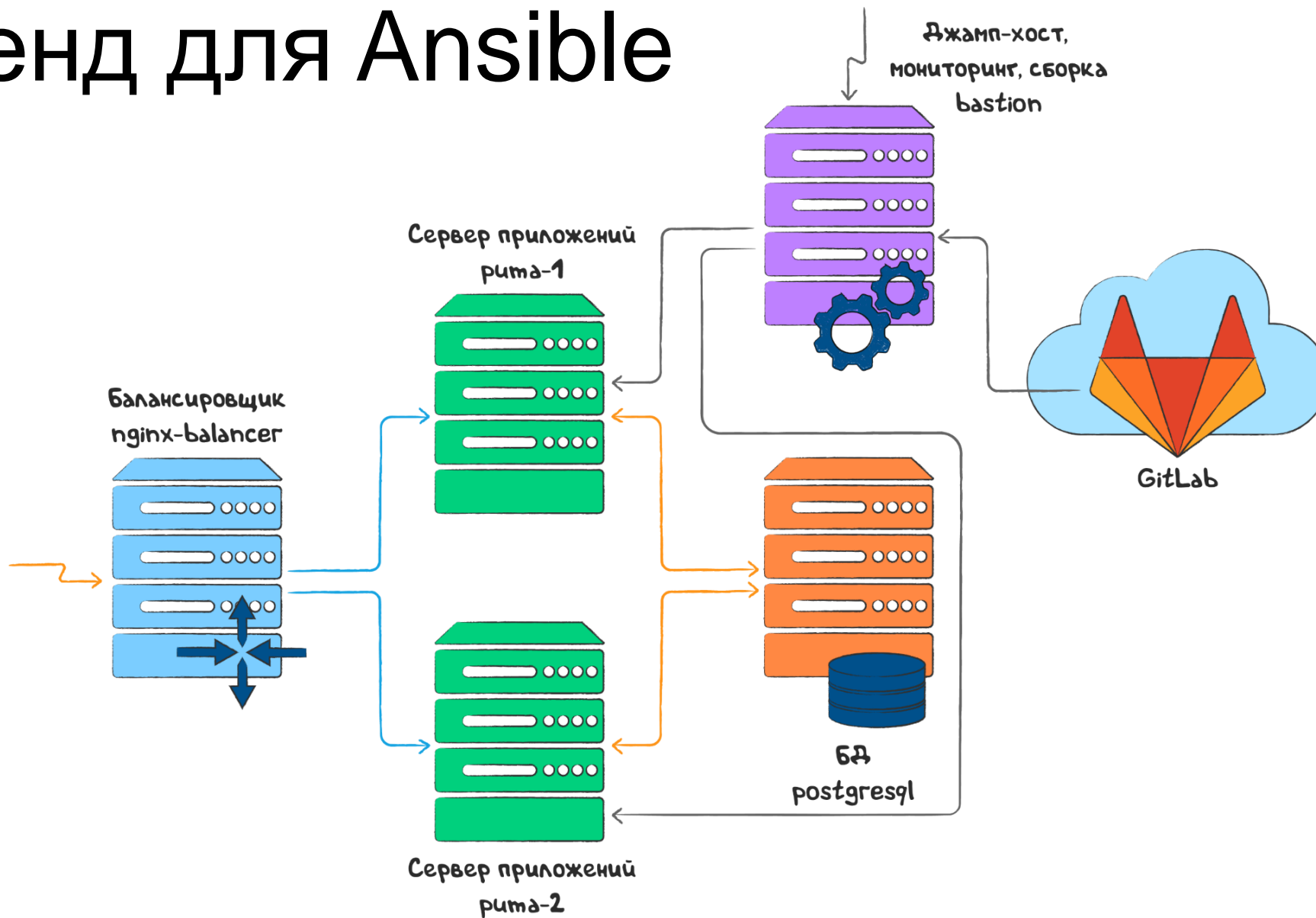


Выбор SCM





Стенд для Ansible





Работа с ansible-vault

Файл `~/.vpaswd`

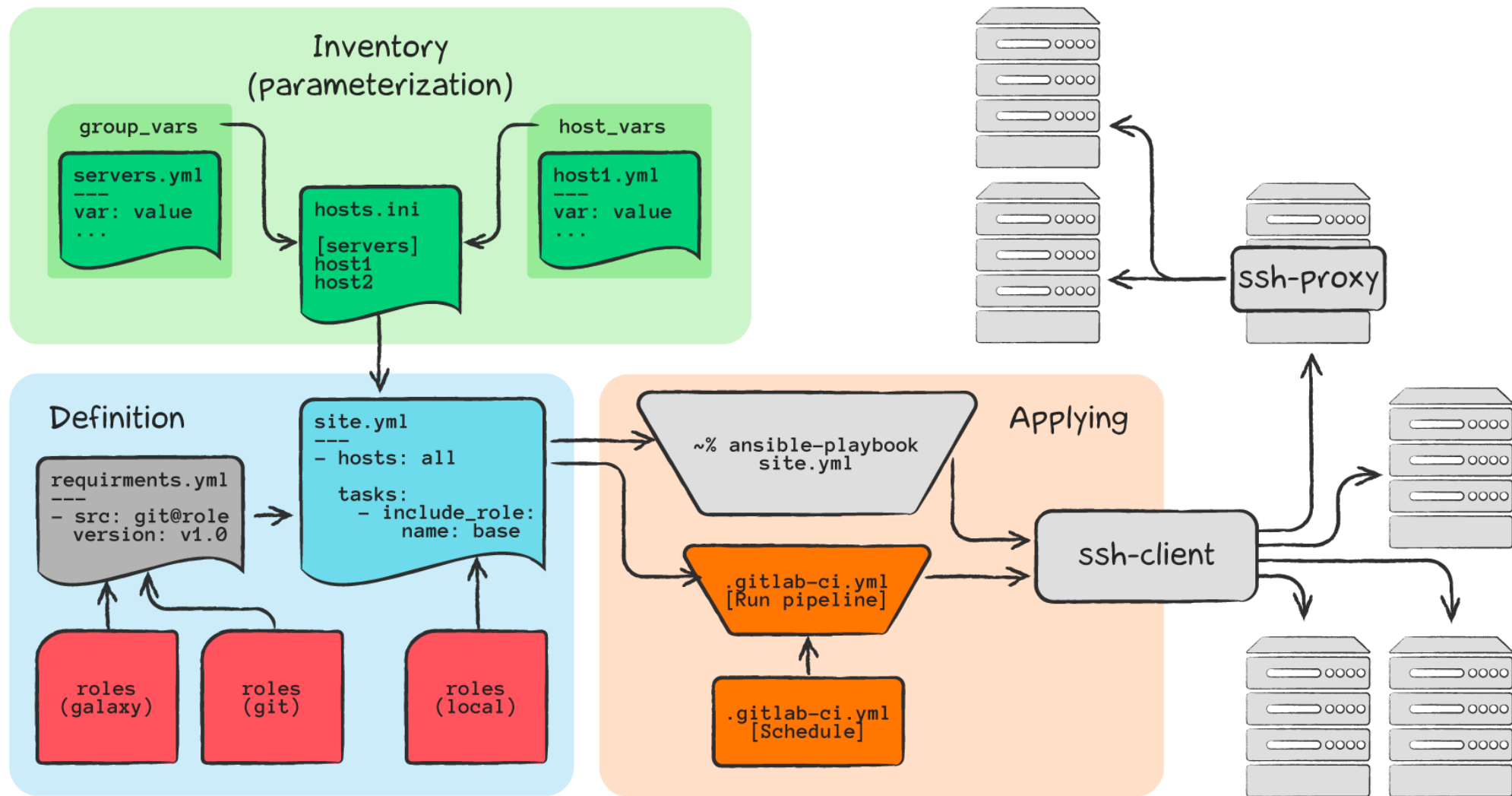
`ansible-vault encrypt`

`ansible-vault encrypt_string`





Ansible data flow





Шаблон → переменные → результат

```
1 {% for key, value in base_sysctl_vars|dictsort %}  
2 {{ key }} = {{ value }}  
3 {% endfor %}
```

```
1 base_sysctl_vars:  
2     net.ipv4.ip_forward: 1  
3     net.netfilter.nf_conntrack_udp_timeout: 0
```

```
1 net.ipv4.ip_forward = 1  
2 net.netfilter.nf_conntrack_udp_timeout = 0
```




Чуть более сложный пример

```
1 # THIS FILE IS ANSIBLE MANAGED.
2 # HANDS OFF!
3
4 # - Connection Settings -
5
6 listen_addresses      = '{{ postgresql_conf['listen_addresses']|default('*') }}'
7 port                  = {{ postgresql_conf['port']|default('5432') }}
8 max_connections        = {{ postgresql_conf['max_connections']|default('100') }}
9
10 # - Memory -
11
12 {% set auto_shared_buffers = ansible_memtotal_mb / 100 * 20 %}
13 shared_buffers          = {{ postgresql_conf['shared_buffers']|default(auto_shared_buffers|round(-1)|int ~ 'MB') }}
14 temp_buffers            = {{ postgresql_conf['temp_buffers']|default('8MB') }}
15 {% set auto_work_mem = ansible_memtotal_mb / 100 * 1 %}
16 work_mem                = {{ postgresql_conf['work_mem']|default(auto_work_mem|round(0)|int ~ 'MB') }}
17 {% set auto_maintenance_work_mem = ansible_memtotal_mb / 100 * 4 %}
18 maintenance_work_mem   = {{ postgresql_conf['maintenance_work_mem']|default(auto_maintenance_work_mem|round(0)|int ~ 'MB') }}
19 {% if postgresql_conf['max_stack_depth'] is defined %}
20 max_stack_depth         = {{ postgresql_conf['max_stack_depth'] }}
21 {% endif %}
22
```

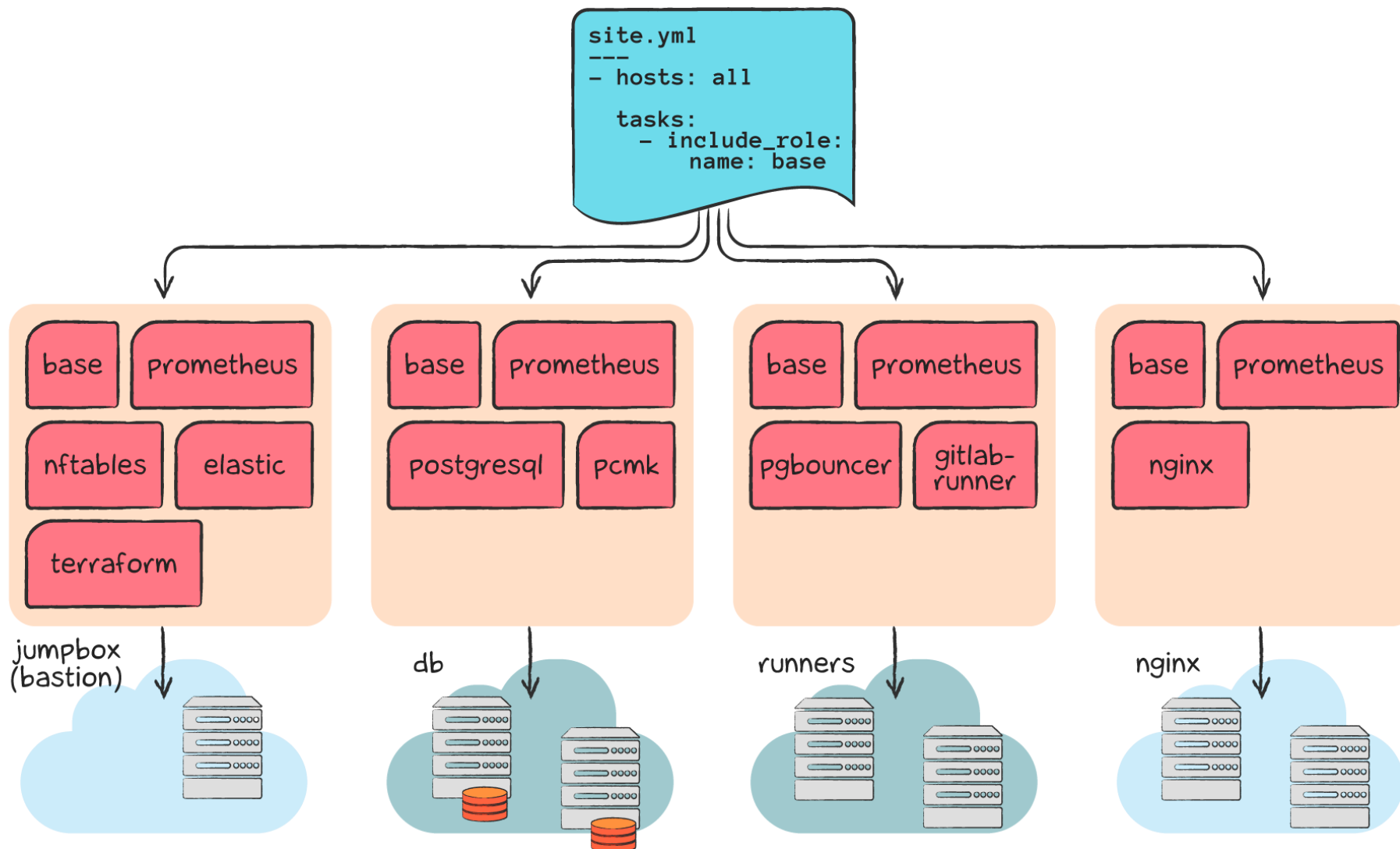
Вычисляемые значения
по умолчанию;

Фильтры;

Условия.



Применим роли к хостам

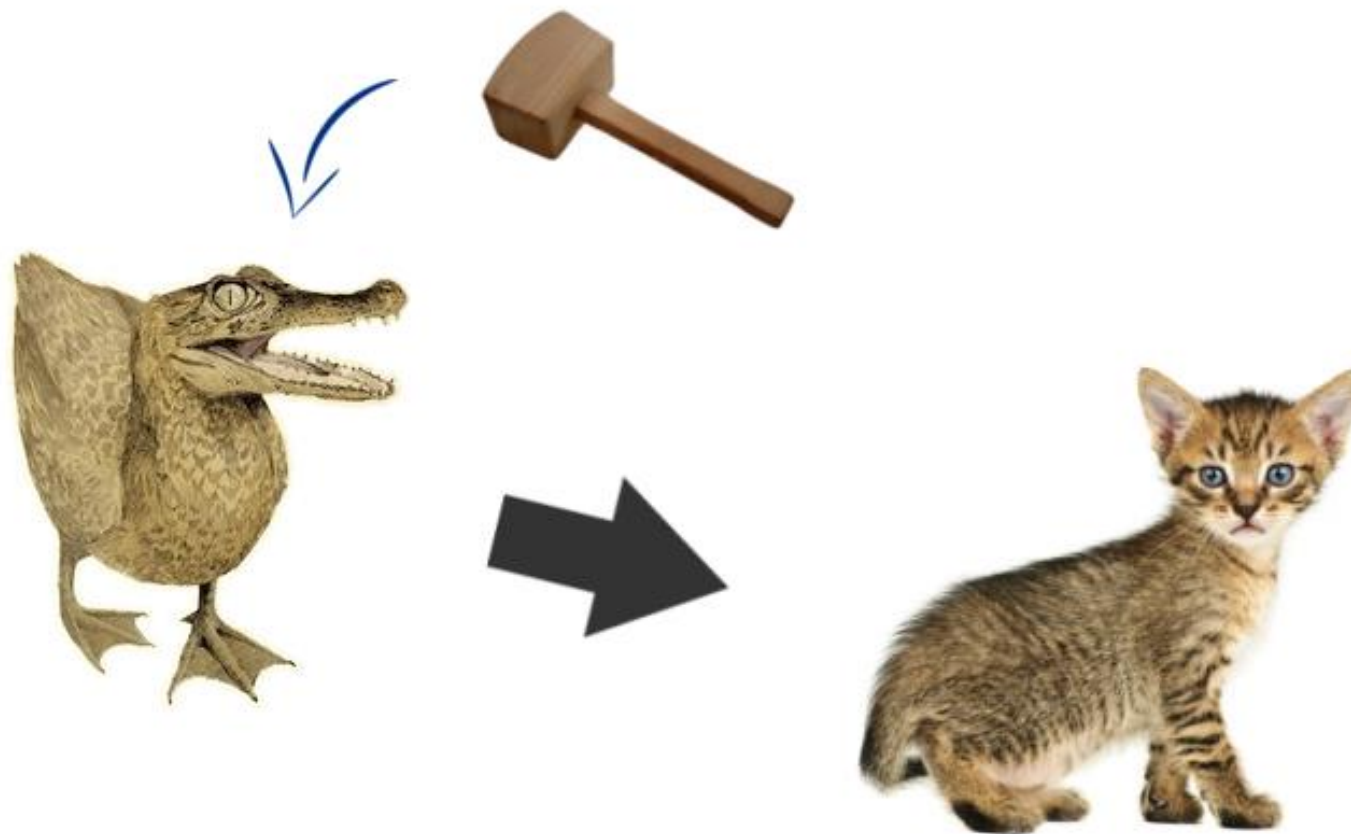


Идемпотентность | Idempotence

Свойство, означающее, что действие может **многokrатно** применяться к объекту **без изменения результата**.

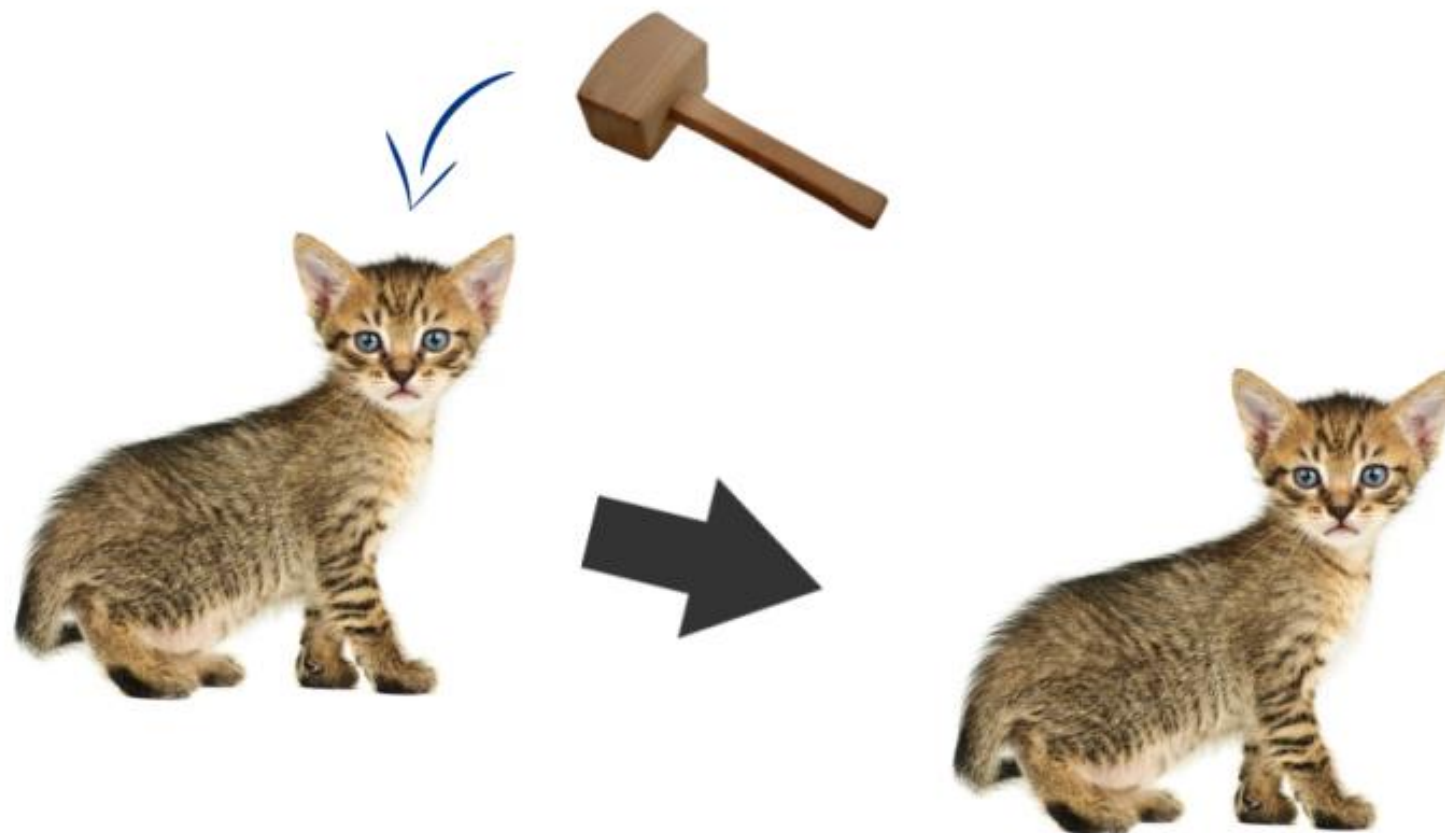


Идемпотентность | Idempotence



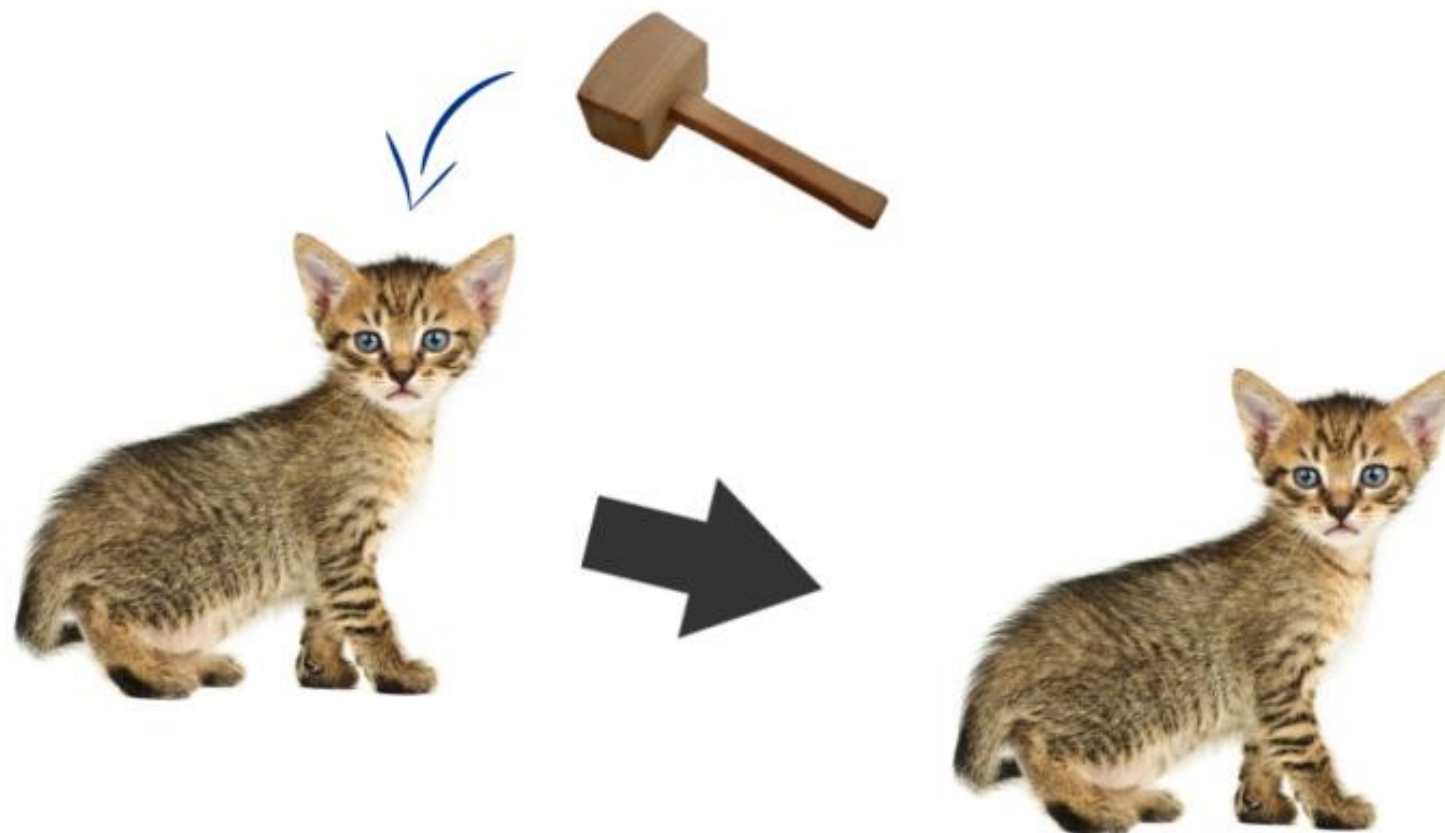


Идемпотентность | Idempotence





Идемпотентность | Idempotence





Проверить регистрацию раннеров

Проверяем регистрацию раннеров в своей группе проектов:

« Settings → CI/CD → Runners »

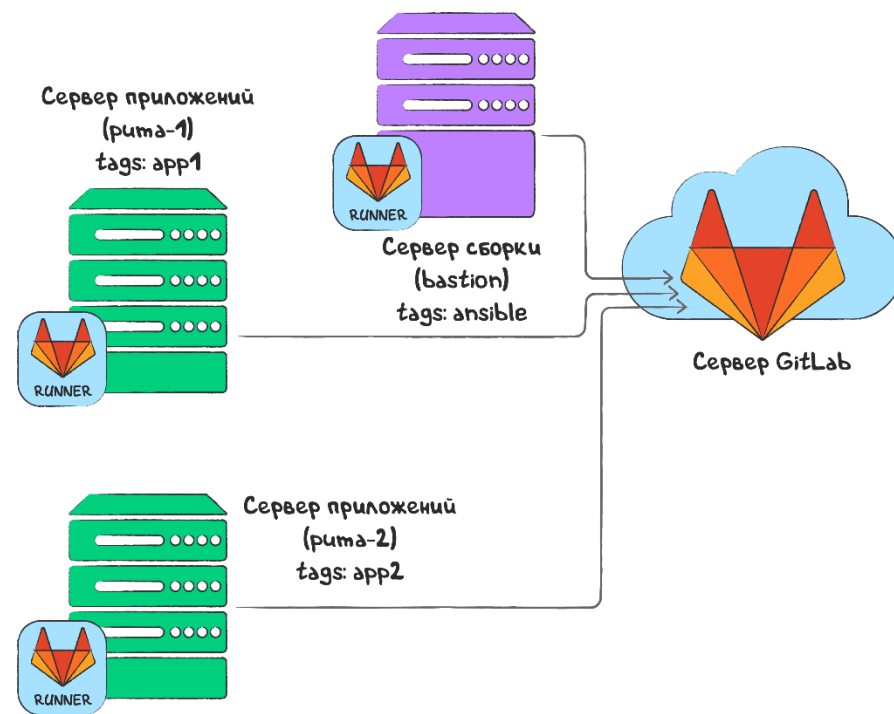
Теги :

`ansible` для сервера сборки

`app1` для первого сервера приложений

`app2` для второго.

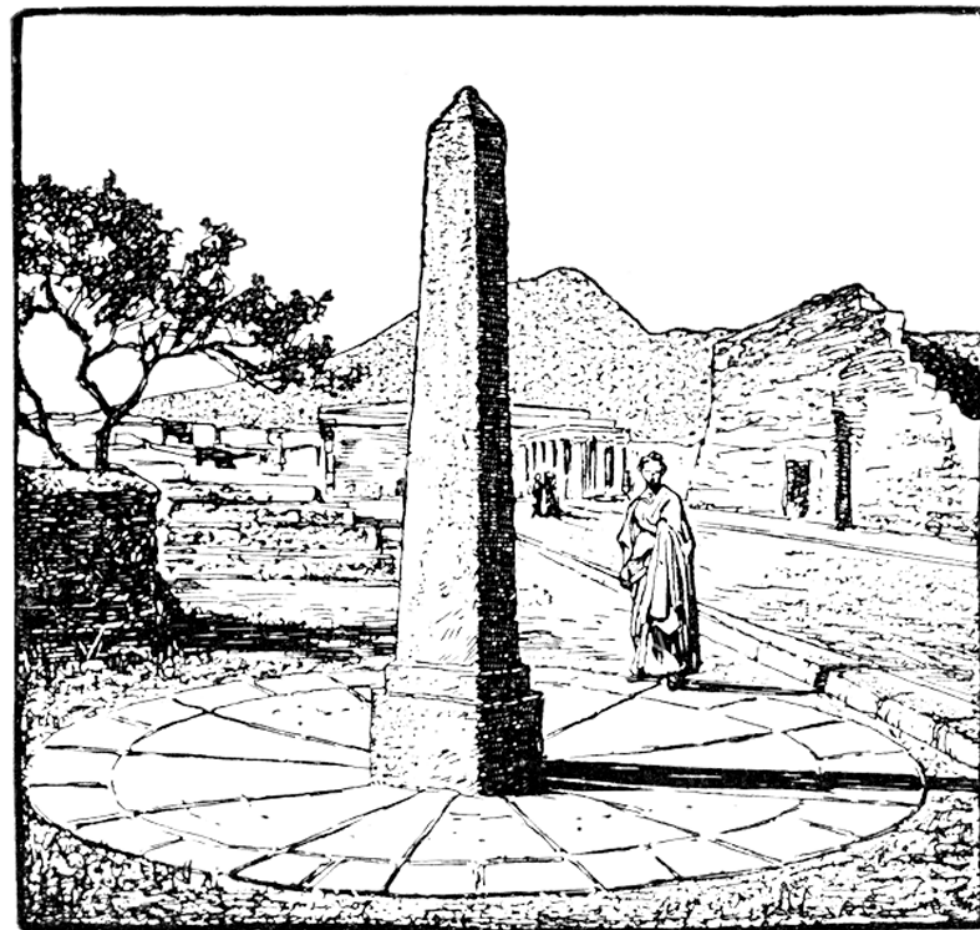
Тип – shell





Запуск по расписанию

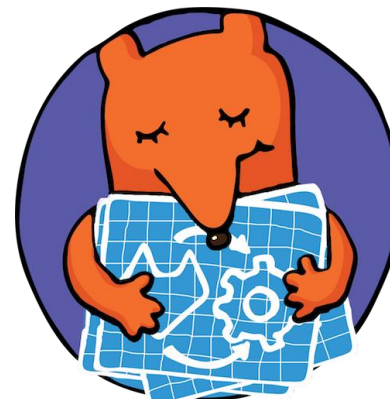
Для поддержания
постоянной конфигурации
можно запускать
site.yml по расписанию





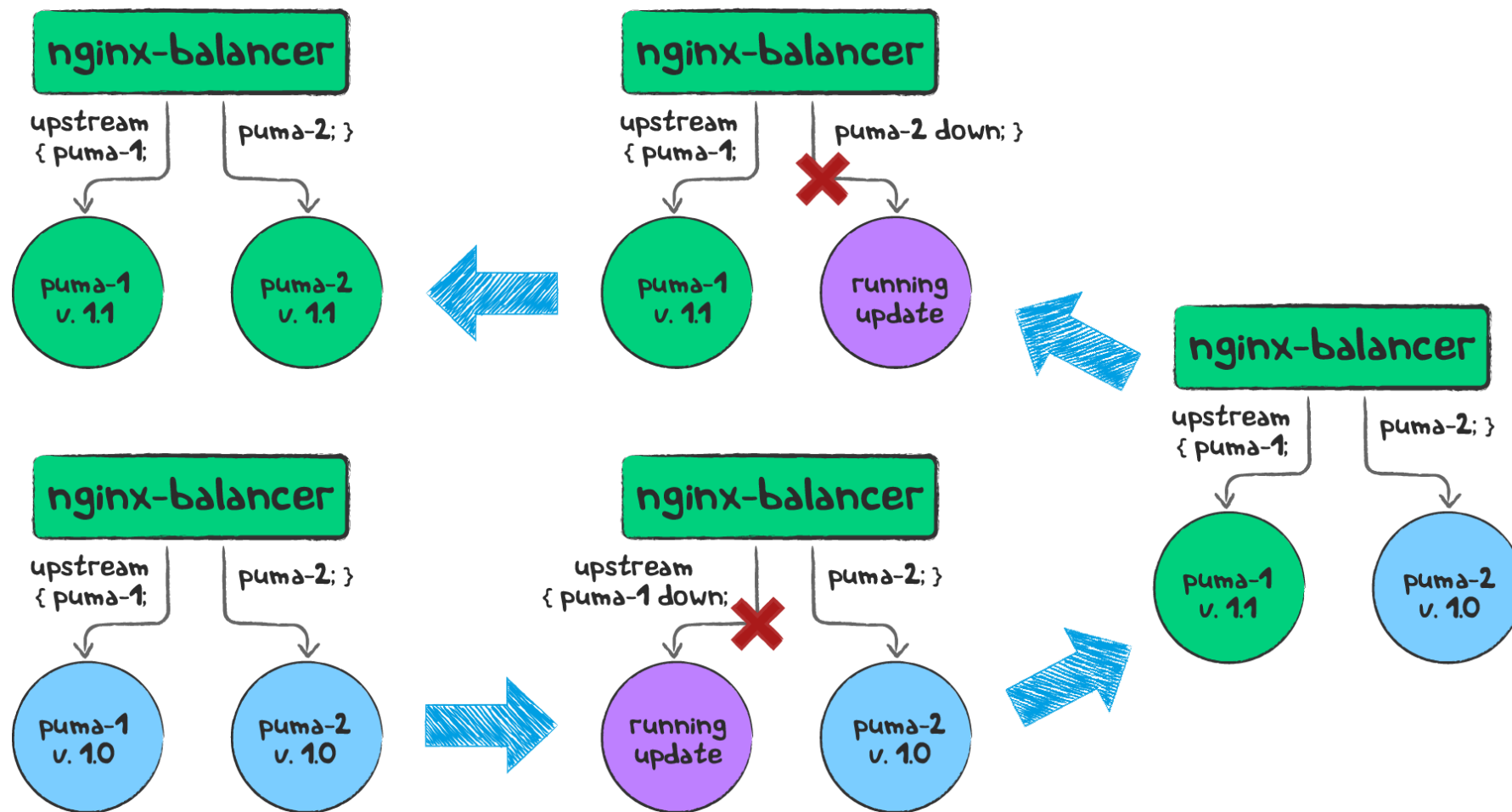
Цели пайплайна (пример)

- скачивание зависимостей, формирование ассетов (CSS, JS);
- тестирование работоспособности сборки;
- обновление версии приложения на площадке без перерыва в обслуживании;
- **отсутствие простоя в случае сбоя обновления.**



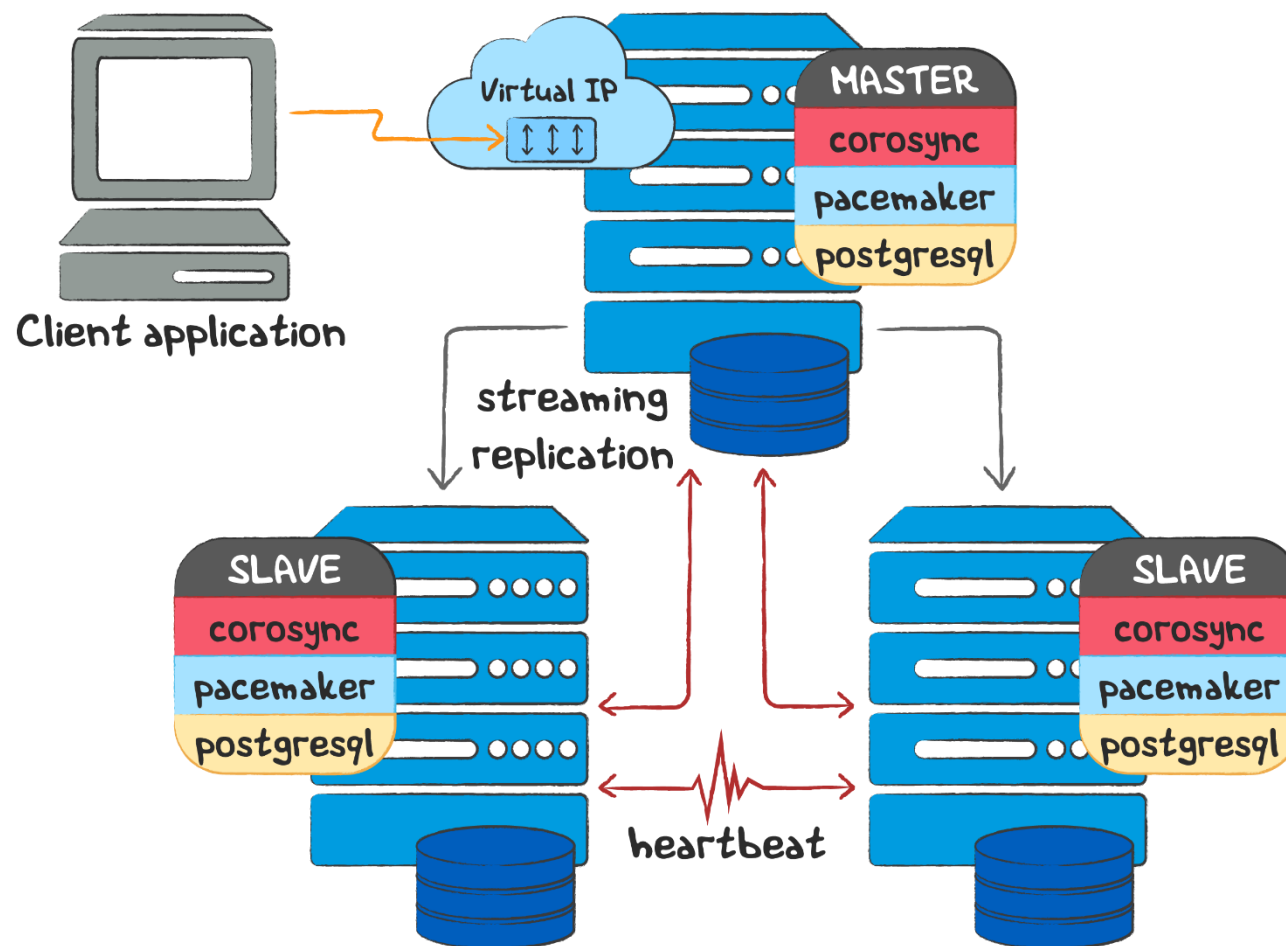


Rolling Update



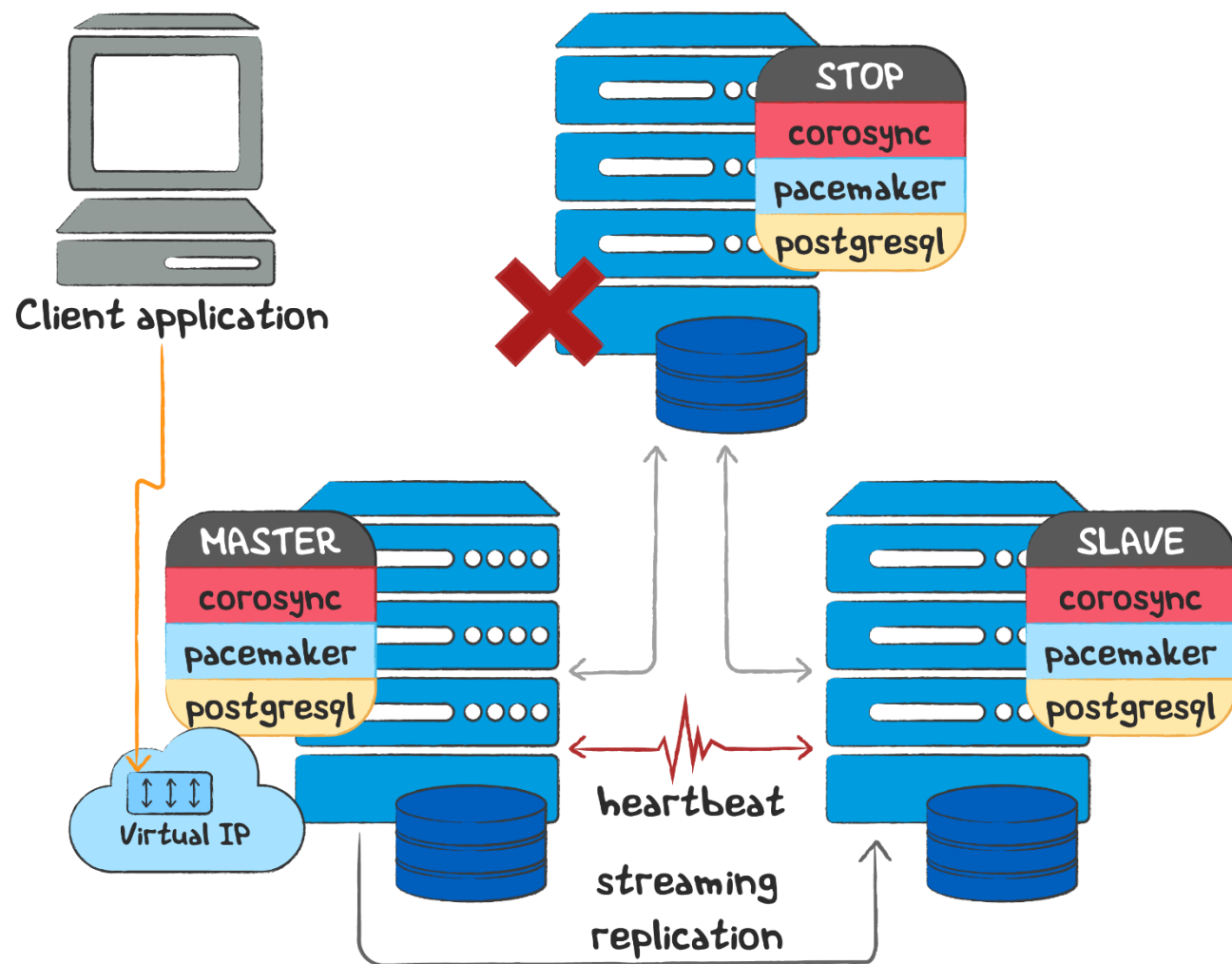


Кластер PostgreSQL-over-Pacemaker





Кластер PostgreSQL: обработка отказа





Вывод

Ansible позволяет реализовывать IaC



ИНФА 100%