

avito.tech

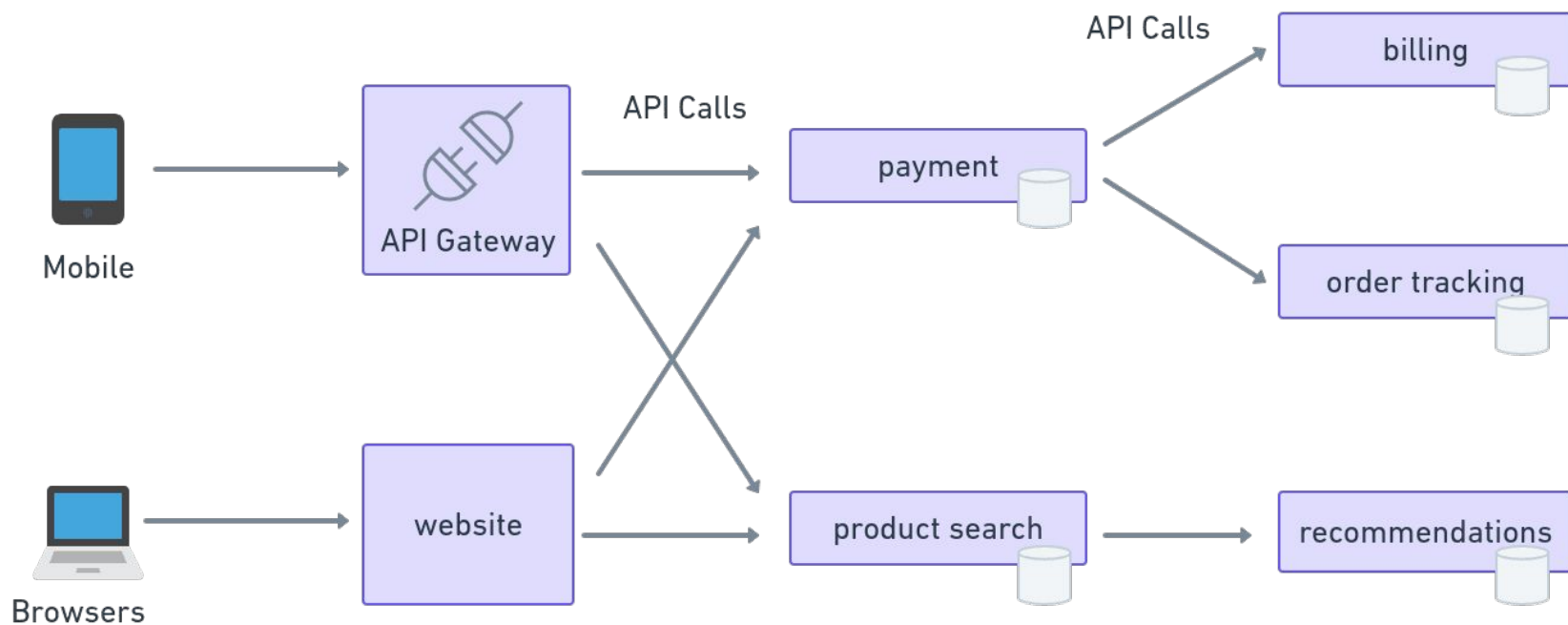
Москва — 2021

Как мы доставляем сервисы в PaaS



Вступление

Вступление



Вступление

Что предоставляем

- ▶ Тестирование – от unit до system
- ▶ Возможность выкатить сервис с новым функционалом с помощью одной команды/кнопки
- ▶ Возможность доставлять через канарейки
- ▶ Уметь все доставить до staging/prod в мультикластерной среде
- ▶ ...

Вступление

Что нужно от разработчика

- ▶ Код сервиса
- ▶ `app.toml` – конфигурационный файл, лежащий в корне репозитория сервиса
- ▶ Конфигурация сервиса из реестра сервисов



Начнем с обычного deploy

Начнем с обычного deploy

```
> avito service deploy -e prod
```

ИЛИ

Подтвердите действие

Деплой ветки [master](#) стратегией [simple](#) сервиса `service-wordcounter` в окружении `prod`!

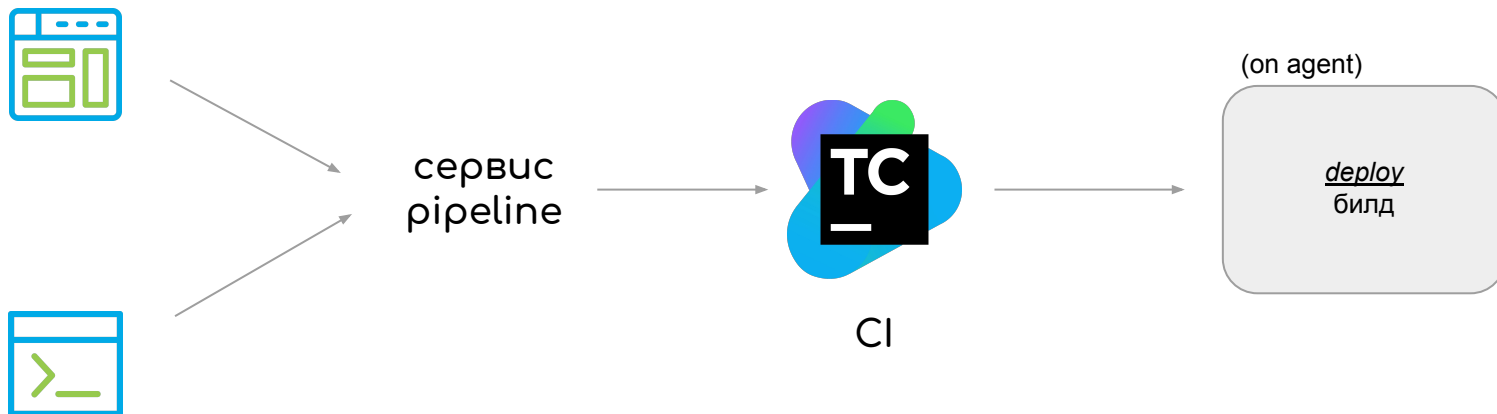
[Проверьте изменения](#) перед выкаткой!

Отмена

Раскатить

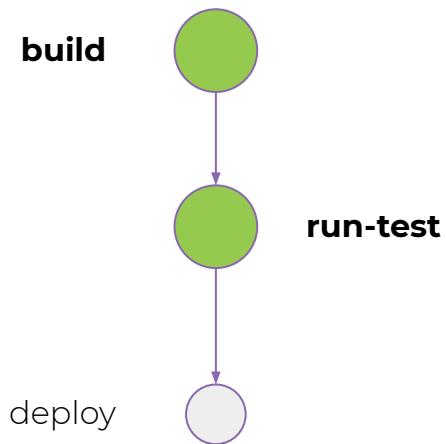
Начнем с обычного deploy

Вкратце об архитектуре



Начнем с обычного deploy

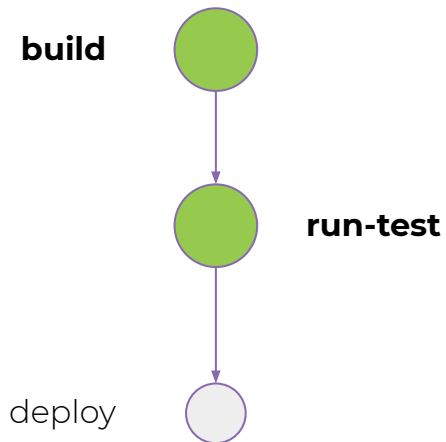
Подготовка кода и его проверка



Начнем с обычного deploy

Подготовка кода и его проверка

- ▶ прогон линтеров
- ▶ валидация схем синхронного и асинхронного api
- ▶ сборка докер образа



- ▶ запуск юнит/интеграционных тестов

Начнем с обычного deploy

Деплой



Начнем с обычного deploy

Создание релиза

- ▶ генерация манифестов во внутреннем сервисе-генераторе
- ▶ сохранение всех настроек, версии образа, сгенеренных манифестов в хранилище релизов

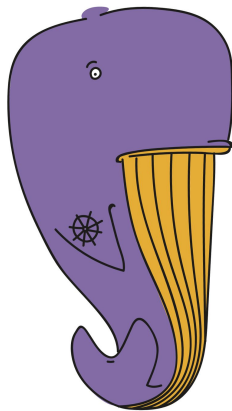
Начнем с обычного deploy

Последние релизы						kappa	theta	zeta
версия	релизер	стратегия		время	▼ связанные задачи			
✓ ver.167		simple	3C	Вчера в 17:03	ARCH-12411	Форсированная выкатка когда сломан ...		
✓ ver.166		simple	3C	18 окт. 16:14	ARCH-9714	AutoPR	+	
✓ ver.165	@naandreev	simple	3C	15 окт. 15:30	ARCH-12184	Добавить в dozer селектор на версию ...	+	
✓ ver.164		simple	3C	14 окт. 12:43	ARCH-12326	Настроить запуск e2e тестов для юнит...		
✓ ver.163		rolling	3C	13 окт. 19:13	ARCH-12452	pipeline: send custom-deploy url for notif...		

Начнем с обычного deploy

Доставка в Kubernetes

- ▶ деплой с помощью Jibe – применение манифестов в Kubernetes



Начнем с обычного deploy

Доставка в Kubernetes



Jibe (англ. кливер – парус)

CLI утилита, написанная на Go, доставляющая манифесты в Kubernetes по определенным стратегиям

Начнем с обычного deploy

Доставка в Kubernetes



Jibe



(generated manifests)

Начнем с обычного deploy

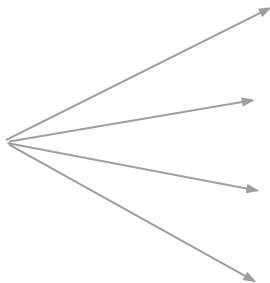
Доставка в Kubernetes



Jibe



(generated manifests)








<i>init</i>	
<i>application</i>	
...	
<i>balancing</i>	

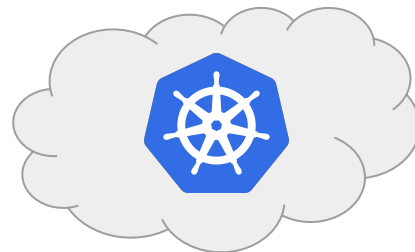
Начнем с обычного deploy

Доставка в Kubernetes



Jibe

<i>init</i>	
<i>application</i>	  
...	 
<i>balancing</i>	








Kubernetes

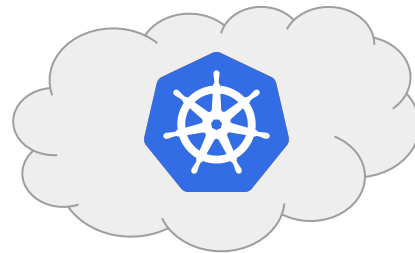
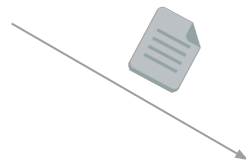
Начнем с обычного deploy

Доставка в Kubernetes



Jibe

<i>init</i>	
<i>application</i>	  
...	 
<i>balancing</i>	





Kubernetes

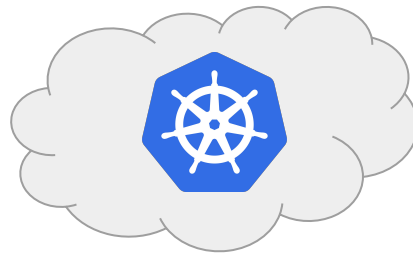
Начнем с обычного deploy

Доставка в Kubernetes



Jibe

<i>init</i>	
<i>application</i>	
...	 
<i>balancing</i>	



Kubernetes

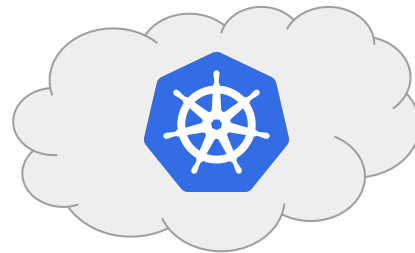
Начнем с обычного deploy

Доставка в Kubernetes



Jibe

<i>init</i>	
<i>application</i>	
...	
<i>balancing</i>	



Kubernetes

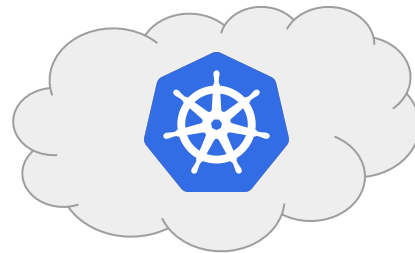
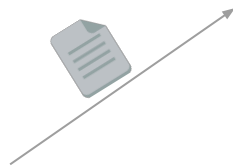
Начнем с обычного deploy

Доставка в Kubernetes



Jibe

<i>init</i>	
<i>application</i>	
...	
<i>balancing</i>	

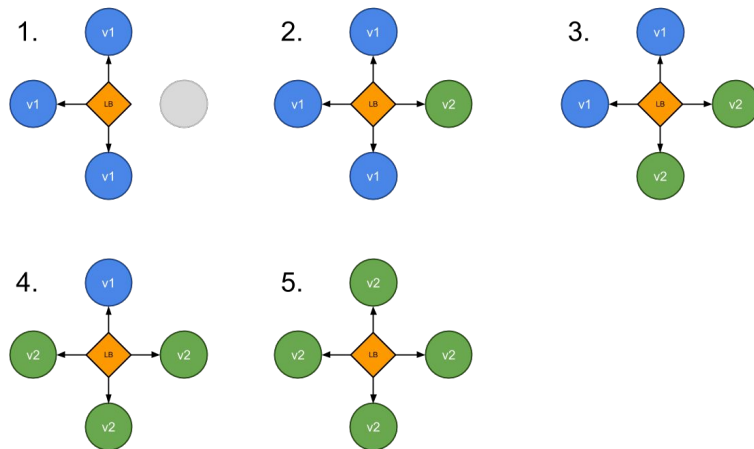


Kubernetes

Начнем с обычного deploy

Как обновляются экземпляры сервиса?

- **rolling update** – плавающее обновление, деплойменты обновляются шаг за шагом, заменяя старые поды на новые

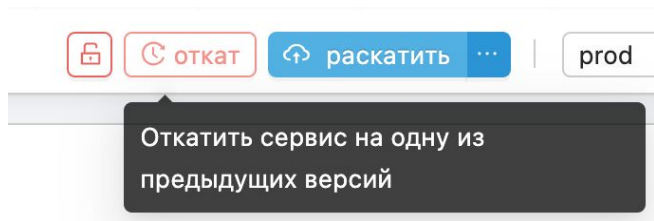


**Нужно срочно
откатиться!**

Нужно срочно откатиться

```
> avito service rollback -e prod
```

ИЛИ



Подтвердите действие

	Версия	Дата	Релизер	Commit
<input type="checkbox"/>	ver.159	27 сент. 22:48		#2786abdd
<input checked="" type="checkbox"/>	ver.158	27 сент. 15:28		#8337412d
<input type="checkbox"/>	ver.157	14 сент. 15:01		#2786abdd

Проверьте изменения перед откатом!

⚠ При оценке изменений помните, что инкремент, который вы видите в стэше будет откатываться, а не деплоиться. А потому в бой поедут изменения выделенные красным цветом, а не зеленым

Нужно срочно откатиться

Подготовка и деплой

- ▶ создание rollback релиза с регенерацией манифестов, используя релиз, на который нужно откатиться
- ▶ деплой с помощью Jibe

Нужно срочно откатиться

Почему генерируем заново?

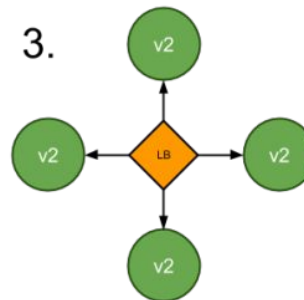
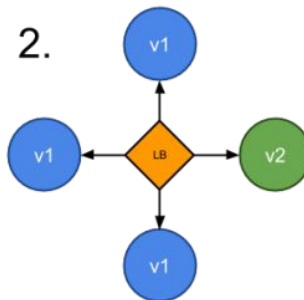
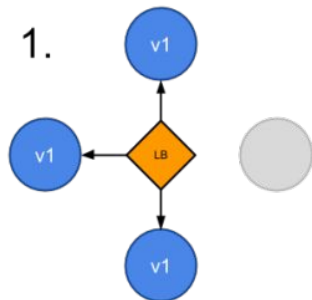
- ▶ не должны откатываться инфраструктурные компоненты, которые катятся вместе с сервисом
- ▶ на всякий случай есть возможность откатиться без регенерации

**Теперь деплоимся
аккуратнее –
канарейкой**

Теперь деплоимся аккуратнее – канарейкой



Теперь деплоимся аккуратнее – канарейкой



Теперь деплоимся аккуратнее – канарейкой

```
avito service canary -p 5
```

ИЛИ

Подтвердите действие

Деплой ветки [master](#) стратегией [canary](#) сервиса **service-wordcounter** в окружении **prod**!

10%



включить скейлинг

[Проверьте изменения](#) перед выкаткой!

Отмена

[Раскатить](#)

Теперь деплоимся аккуратнее – канарейкой

В начале обычный деплой

- ▶ повторяется точь в точь процесс *deploy*, но вместо обычного релиза будет релиз со стратегией *canary* с дополнительными манифестами для канареечной балансировки

Теперь деплоимся аккуратнее – канарейкой

Обновление канарейки



?



выкатить на полную

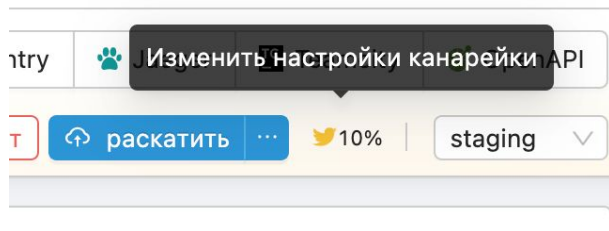


убрать канарейку



увеличить
канарейку

Теперь деплоимся аккуратнее – канарейкой



Настройки канарейки ✕

☒ включить скейлинг

Удалить

Обновить

↻ На весь трафик

Теперь деплоимся аккуратнее – канарейкой

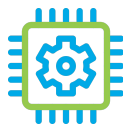
Обновление канарейки

- ▶ отдельный билд в CI ***canary-update*** с приоритетной очередью
- ▶ запускается **Jibe**

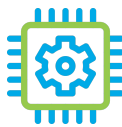
Теперь деплоимся аккуратнее – канарейкой



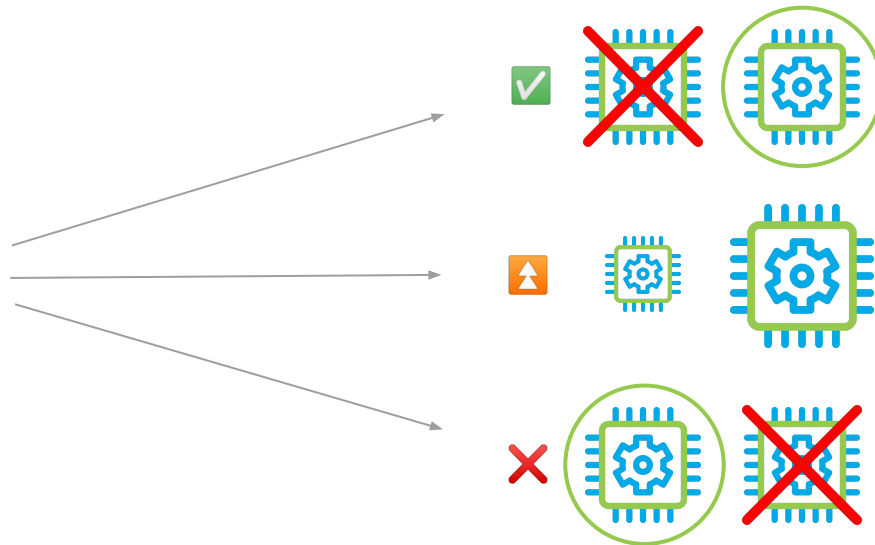
Jibe



v1



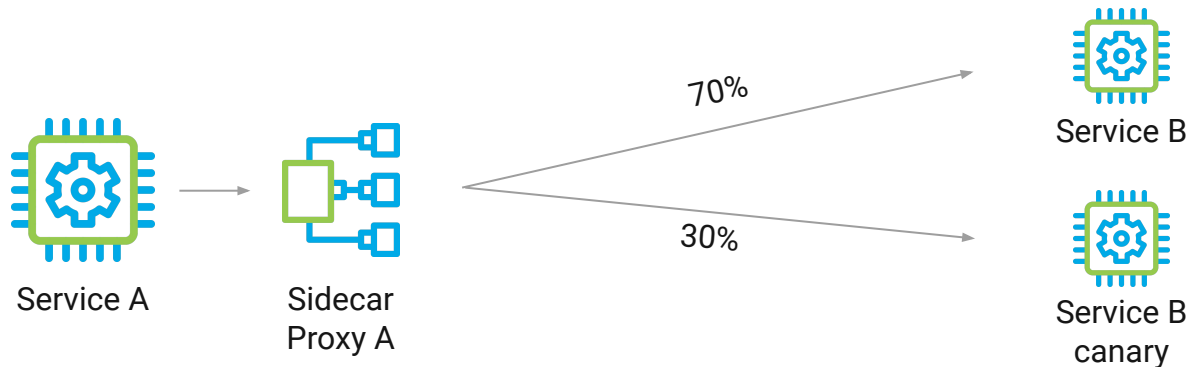
v2
canary



Теперь деплоимся аккуратнее – канарейкой

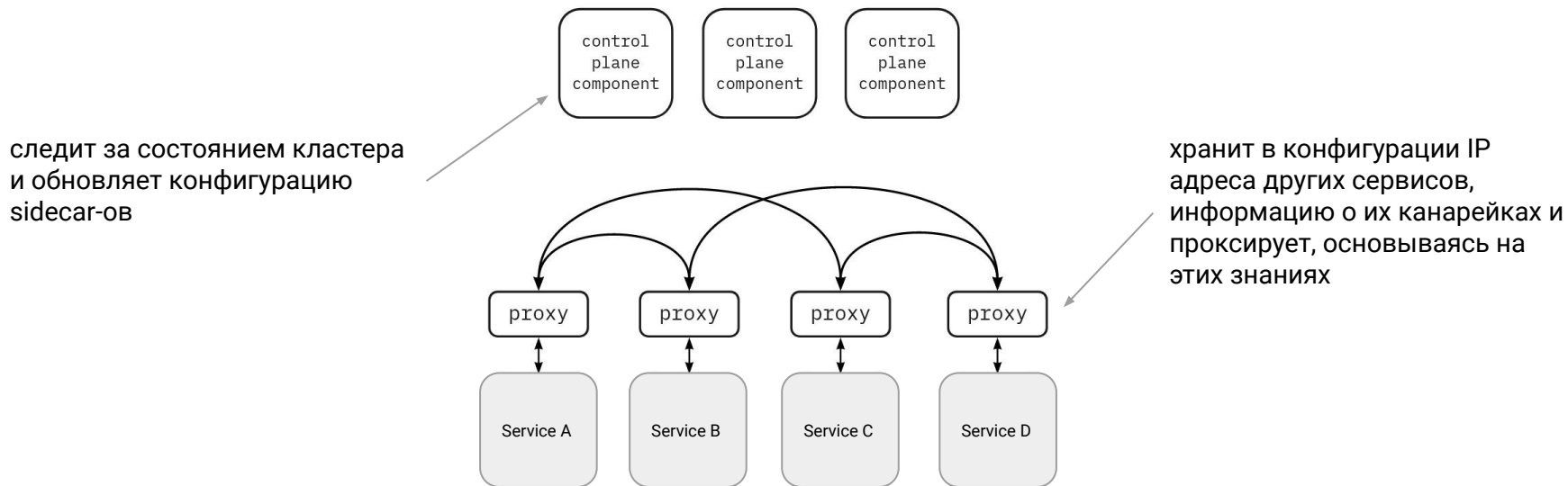
Как происходит балансировка трафика?

- ▶ Рядом с каждой репликой сервиса поднимается sidecar контейнер с умной прокси
- ▶ Все сервисы объединяются в единую сеть – Service Mesh – при этом сервисы ничего не знают о правилах балансировки и о канарейках



Теперь деплоимся аккуратнее – канарейкой

Обновление балансировки





Поподробнее про наше Service Mesh решение
<https://youtu.be/g720zslP3kM>

Доставка в несколько кластеров

Доставка в несколько кластеров

Какие вводные

- ▶ имеется несколько Kubernetes кластеров
- ▶ сервис нужно катить одновременно во все кластеры

Доставка в несколько кластеров

Что еще умеет Jibe


- ▶ выполняет все сценарии в нескольких кластерах
- ▶ каждый этап применяется параллельно в каждом кластере

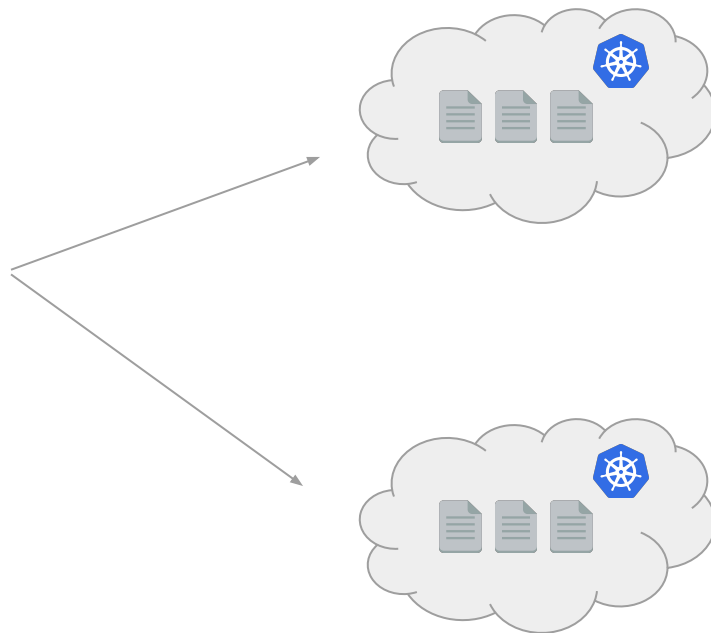
Доставка в несколько кластеров

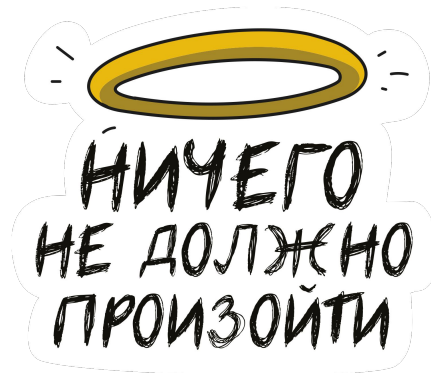
Доставка в Kubernetes-ы



Jibe

<i>init</i>	
<i>application</i>	  
...	 
<i>balancing</i>	

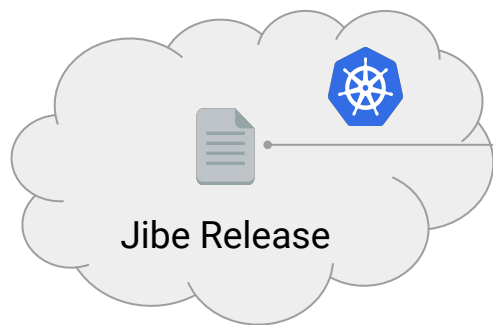




А что если в разных кластерах окажутся разные версии сервиса?

Доставка в несколько кластеров

Кастомный ресурс в Kubernetes



```
1 apiVersion: jibe.avito.ru/v1
2 kind: Release
3 metadata:
4   annotations:
5     jibe.avito.ru/resource-storage-max-size: 0.9MB
6     jibe.avito.ru/resource-storage-type: configmaps
7     releases.jibe.avito.ru/cluster: theta
8     releases.jibe.avito.ru/release-id: 6139d8d85c38ea7739e14ba9
9   creationTimestamp: "2021-09-09T09:50:25Z"
10  labels:
11    app: service
12    cluster: theta
13    environment: prod
14    service: wordcounter
15    version: v243
16  name: wordcounter-v243
17  namespace: wordcounter
18  spec:
19    cluster: theta
20    deployStrategy: simple
21    name: wordcounter
22    resources: '{"resources_meta":[{"Gvk":{"Group": ...}'
23    stage: balancing
24    status: deployed
25    version: v243
```

Доставка в несколько кластеров

Избыточность манифестов

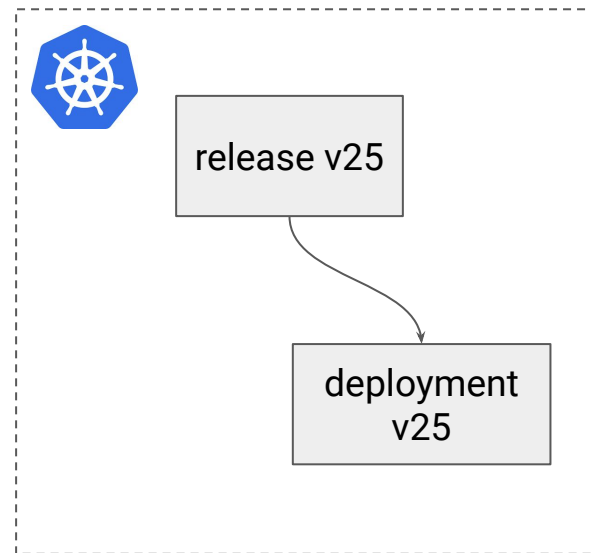
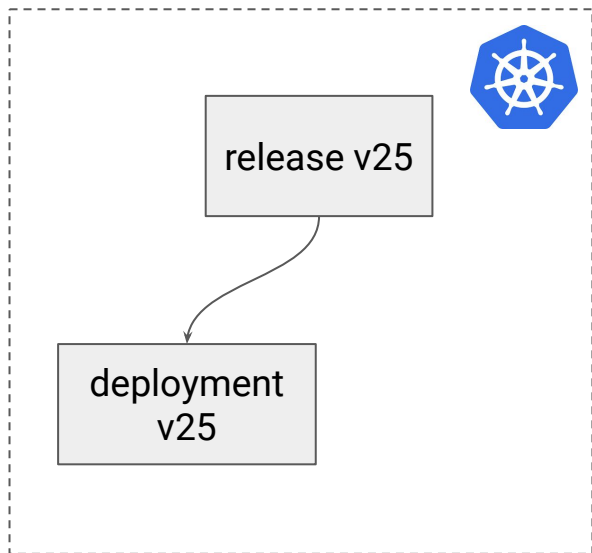
- ▶ рядом с основными релизом в кластере в текстовом виде сохраняются копии манифестов сервиса из других кластеров
- ▶ такая избыточность позволяет синхронизировать релизы сервиса в разных кластерах

Доставка в несколько кластеров

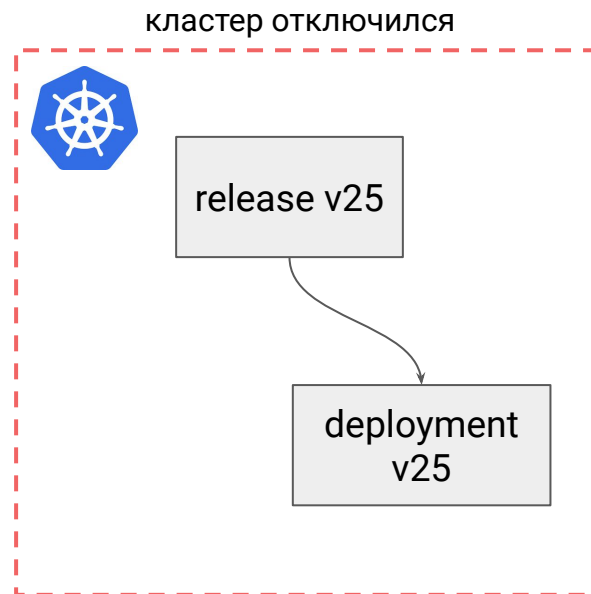
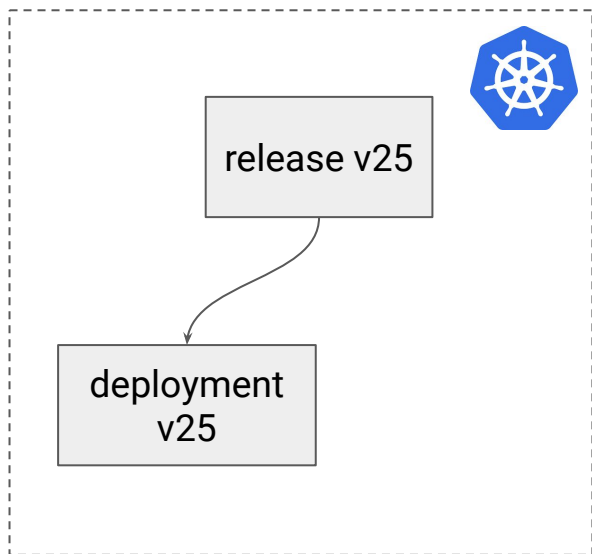
Рассмотрим типовые **use-cases** синхронизации

- ▶ кластер отключили и его возвращают спустя какое-то время с устаревшим состоянием (старые версии сервисов)

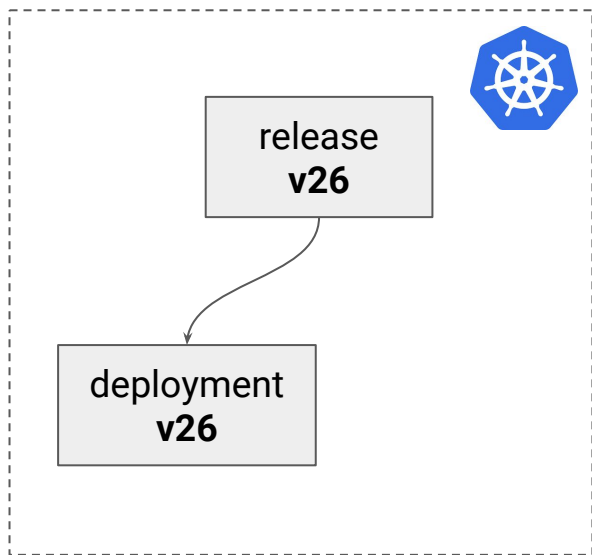
Доставка в несколько кластеров



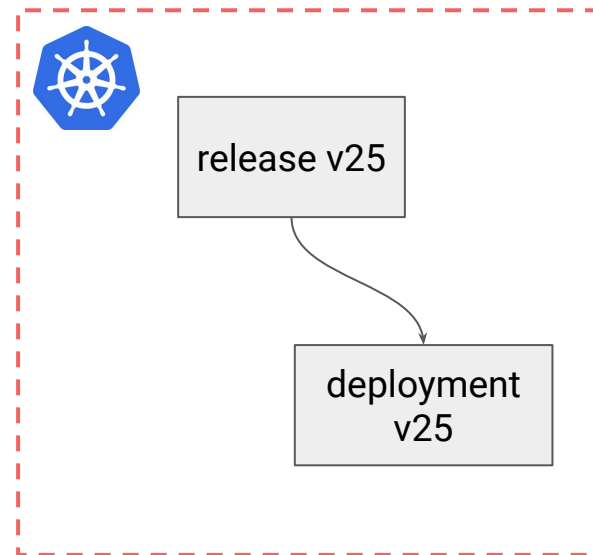
Доставка в несколько кластеров



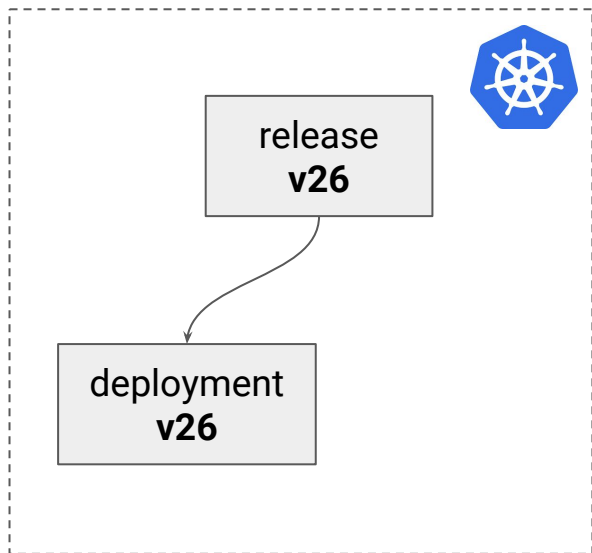
Доставка в несколько кластеров



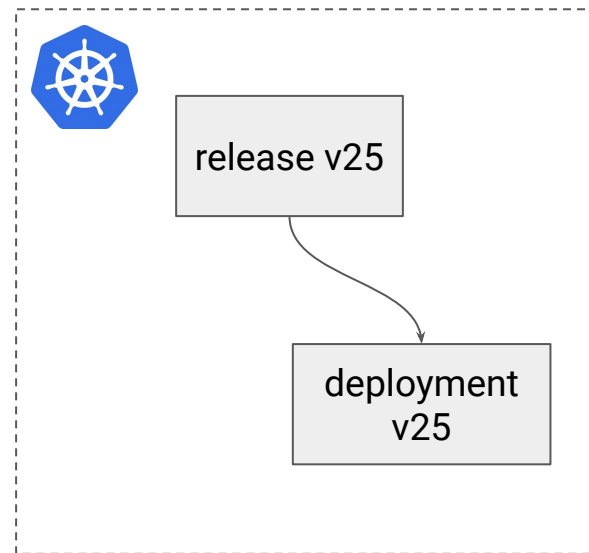
новая версия сервиса
выкатилась в
оставшийся кластер



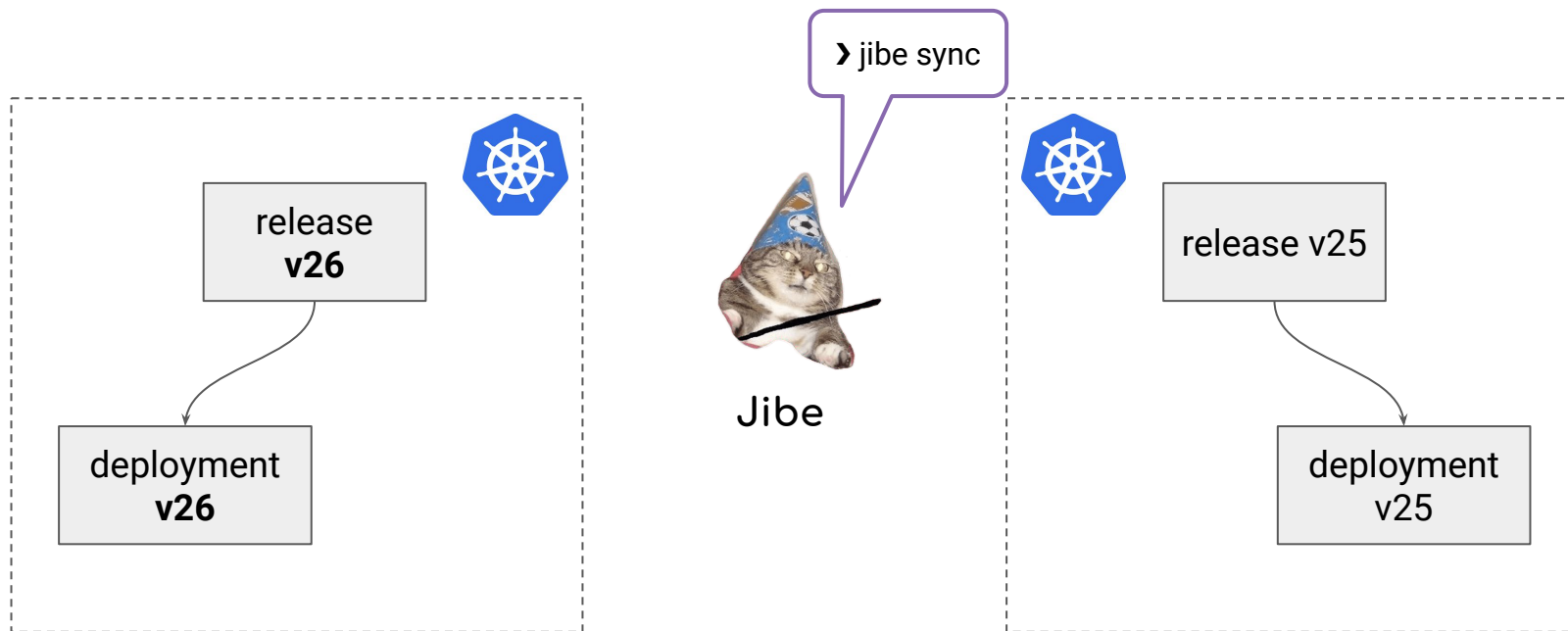
Доставка в несколько кластеров



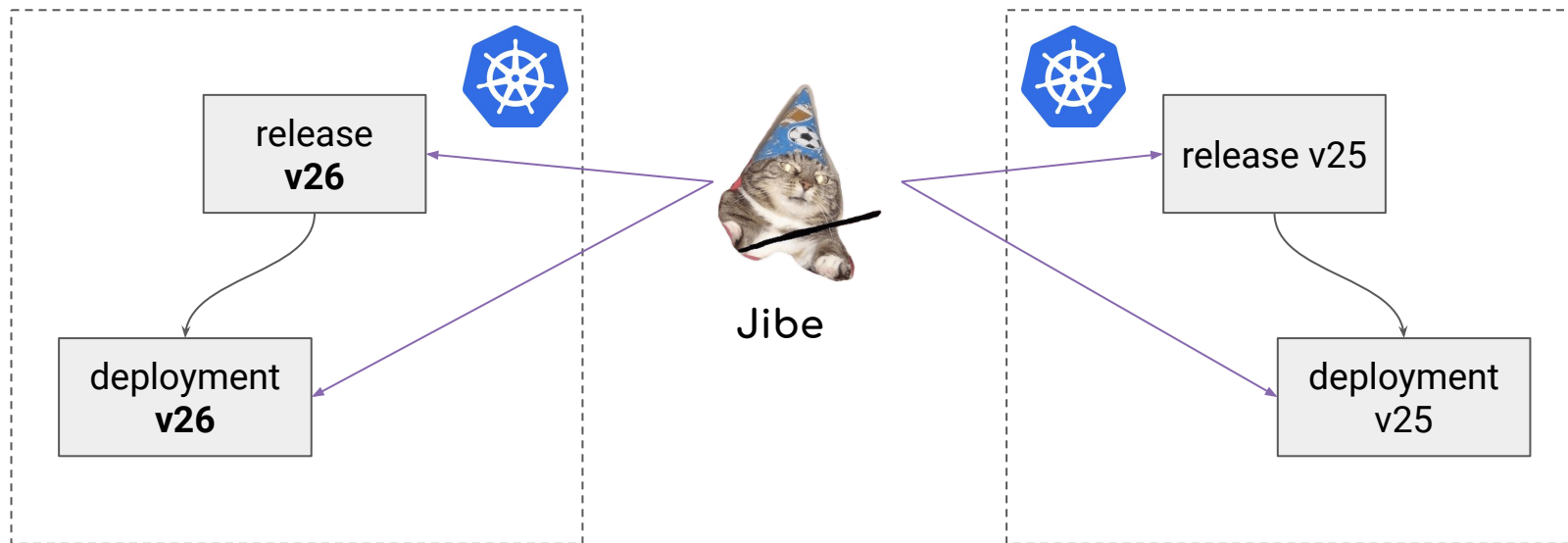
кластер включился обратно



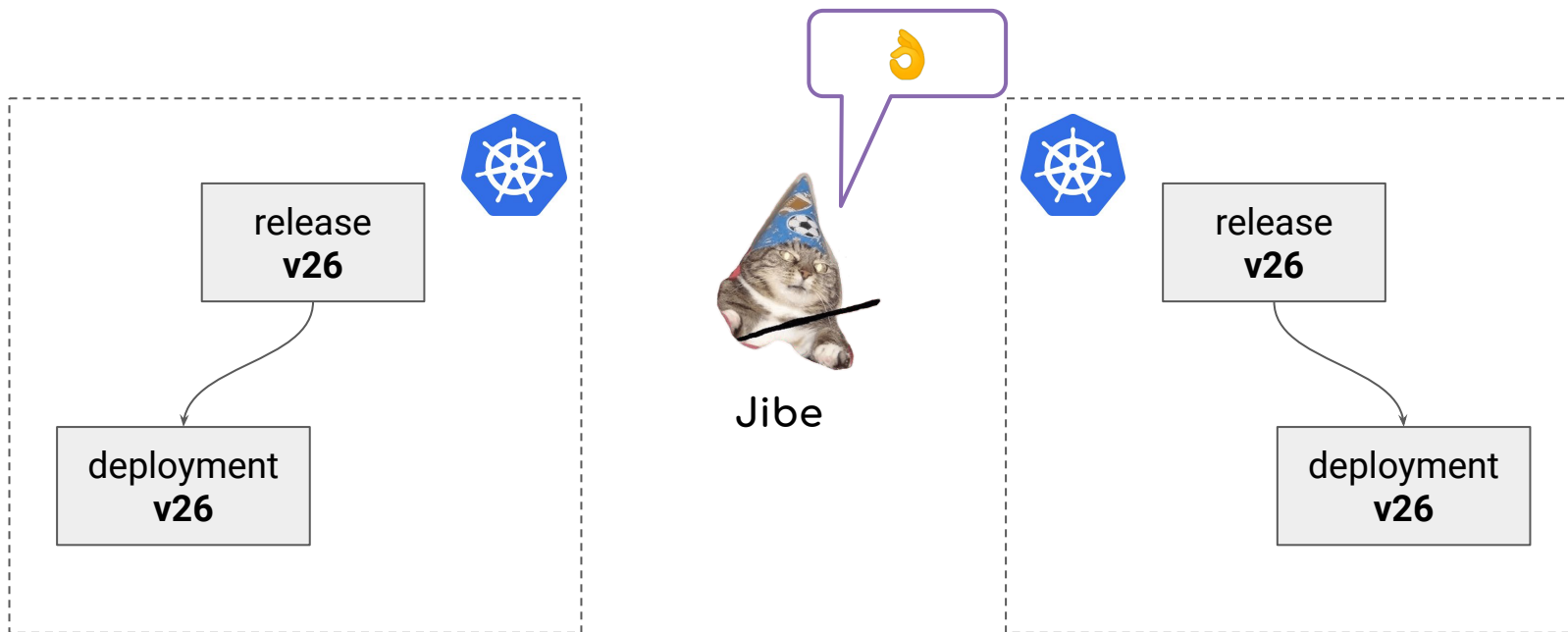
Доставка в несколько кластеров



Доставка в несколько кластеров



Доставка в несколько кластеров

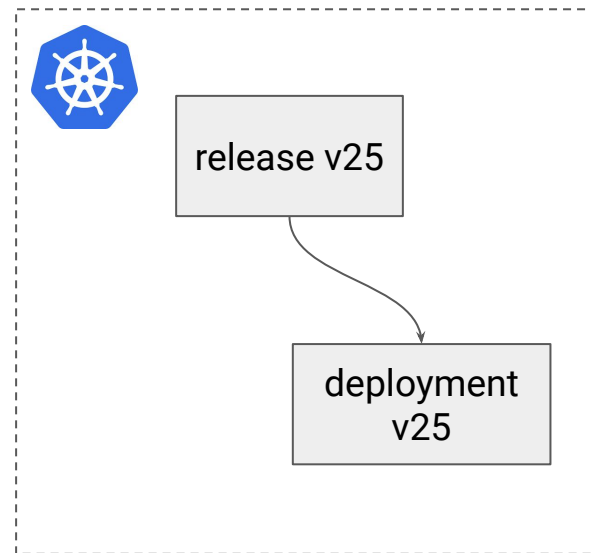
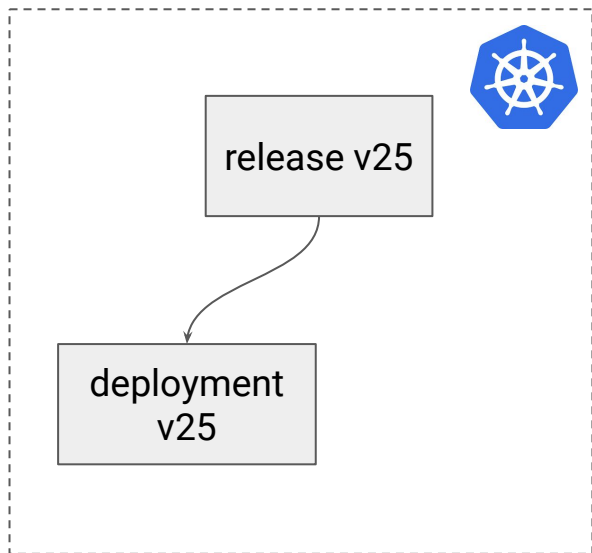


Доставка в несколько кластеров

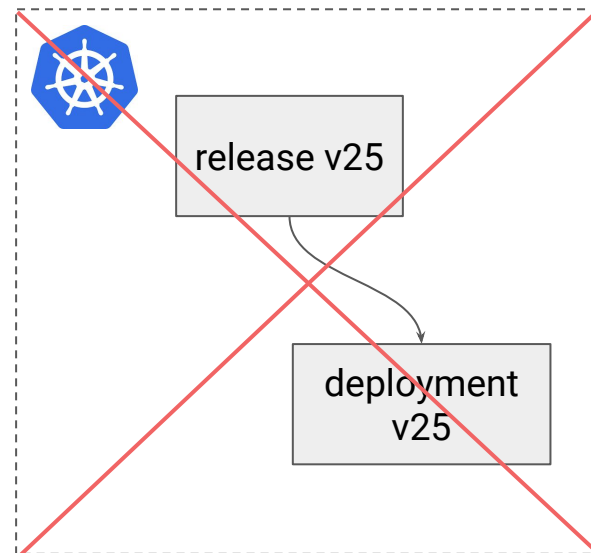
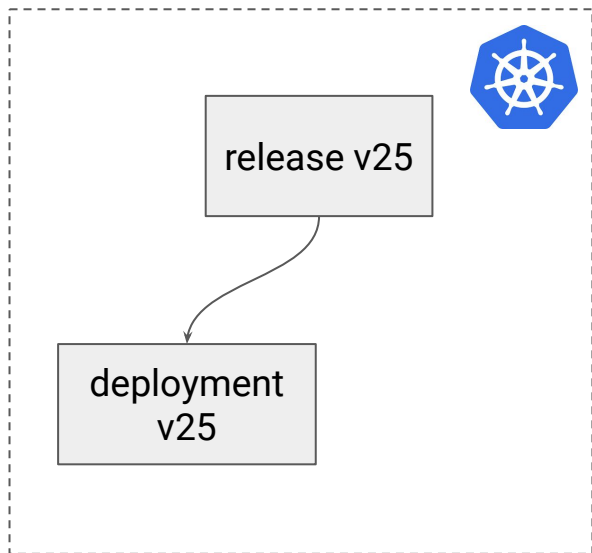
Рассмотрим типовые сценарии

- ▶ кластер отключили и его возвращают спустя какое-то время с устаревшим состоянием (старые версии сервисов)
- ▶ кластер упал и потерялся весь стейт, хранившийся в **etcd**

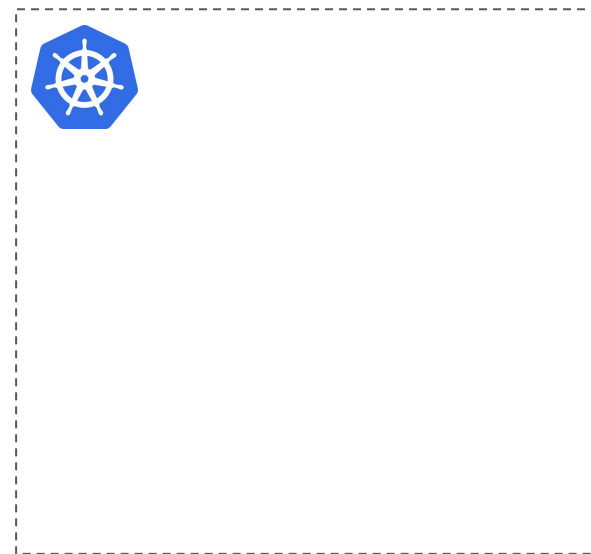
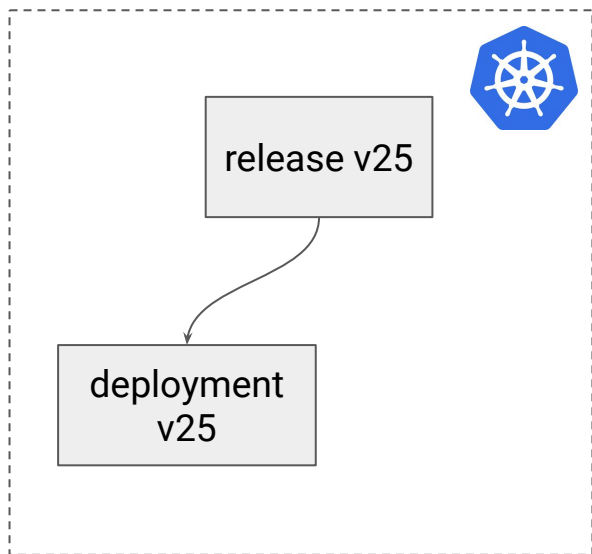
Доставка в несколько кластеров



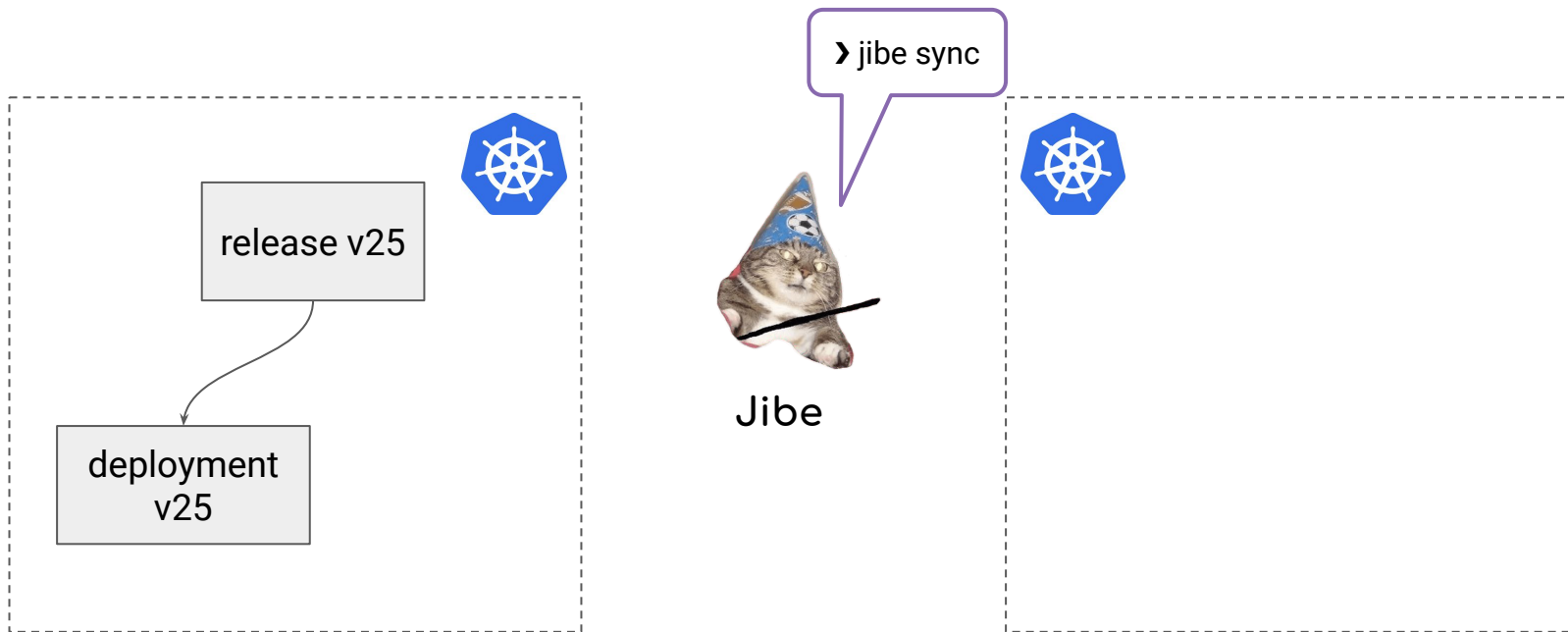
Доставка в несколько кластеров



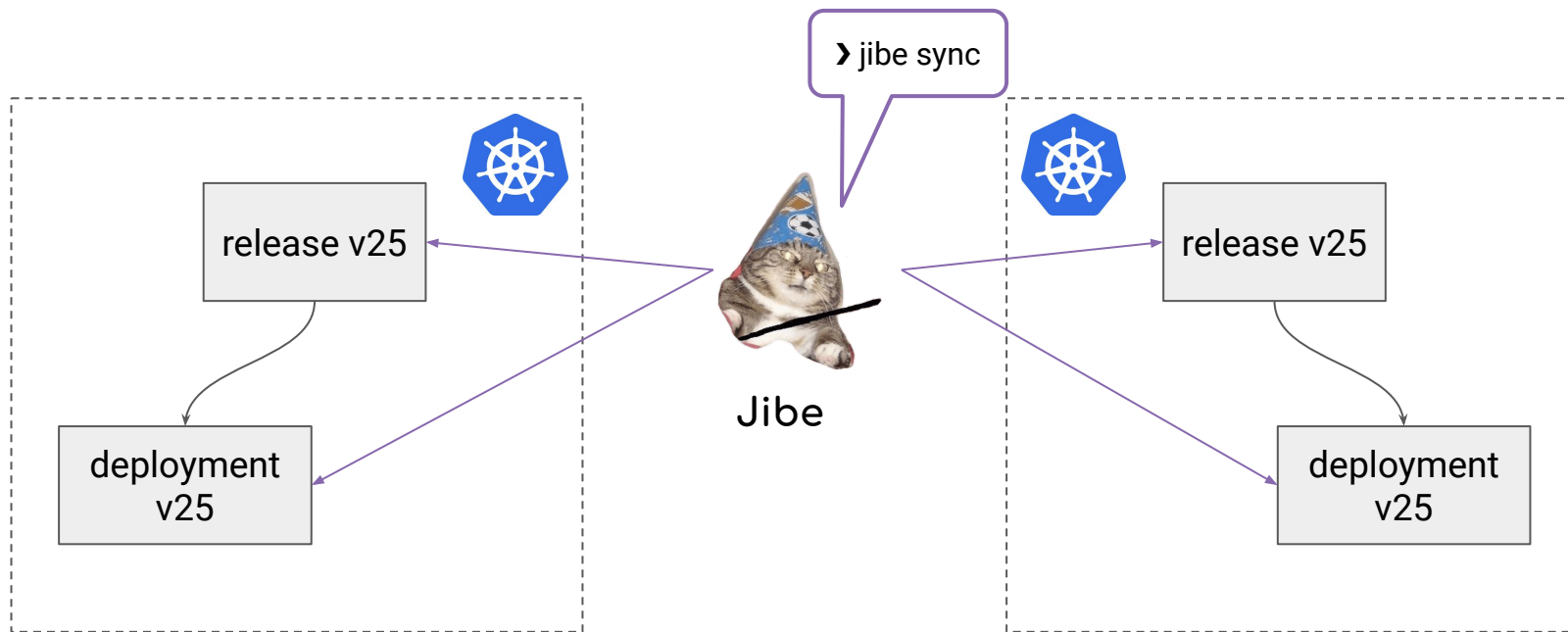
Доставка в несколько кластеров



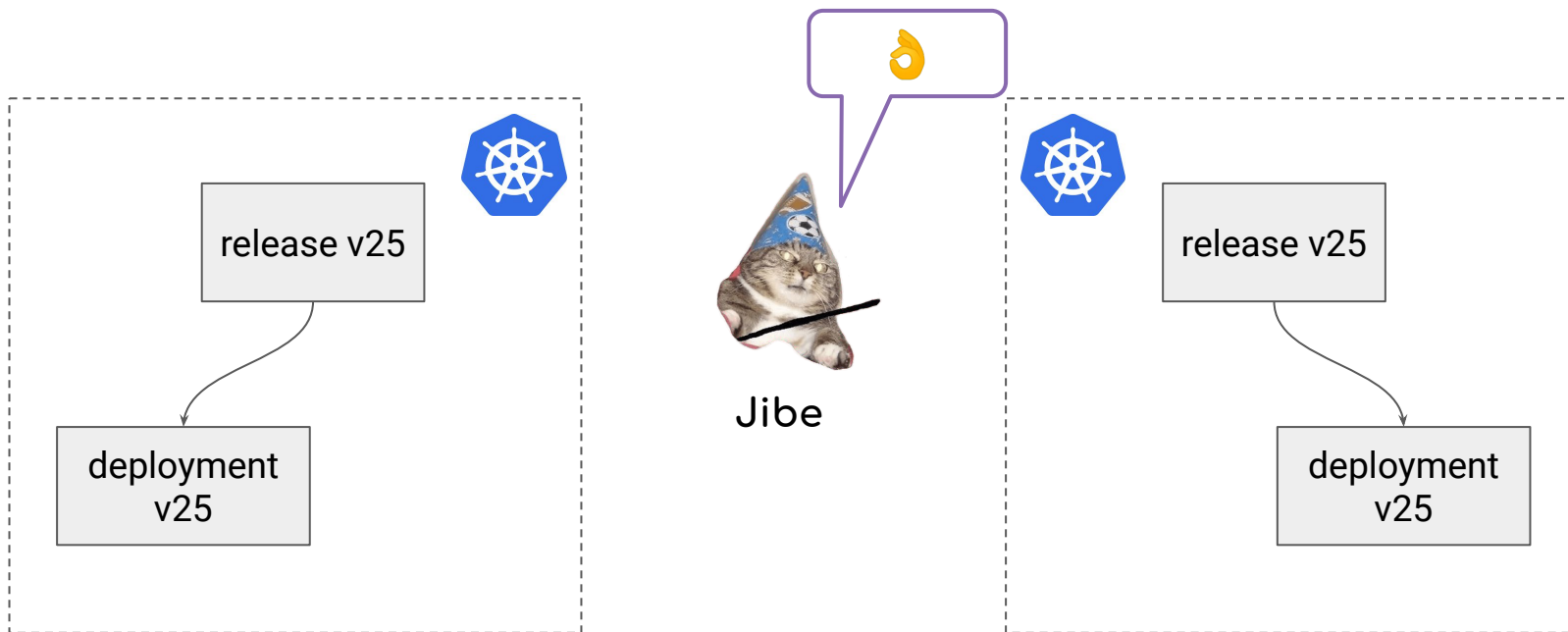
Доставка в несколько кластеров



Доставка в несколько кластеров



Доставка в несколько кластеров



Доставка в несколько кластеров

Jibe sync

- ▶ синхронизирует релизы сервиса в разных кластерах
- ▶ покрывает все сценарии: канарейки, удаление сервиса из кластера и пр.

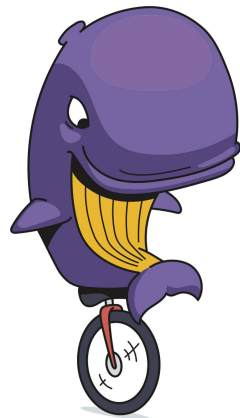
Пути развития



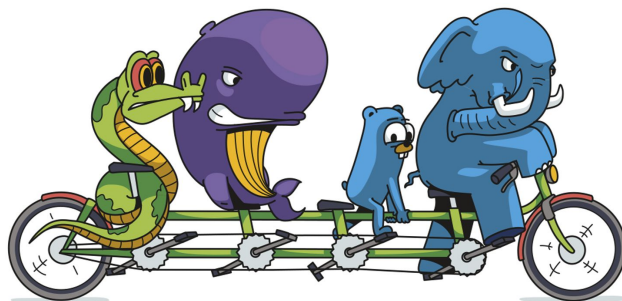
Как сделать CI/CD PaaS-а в Авито еще лучше?



Внедрение автоматических канареек для полноценного Continuous Deployment



**Ускорение сборки докер образа путем
переиспользования докер кэша на CI-агентах**



**Уметь работать с другими СI-системами для большей
гибкости и масштабируемости**

avito.tech

Москва — 2021

Андреев Николай

Engineer @ PaaS



naandreev@avito.ru

Спасибо!