

Skillbox

Service Discovery & Configuration

Управление конфигурацией

Андрей Гордиенков

Solution Architect

ABAX

На прошлом уроке

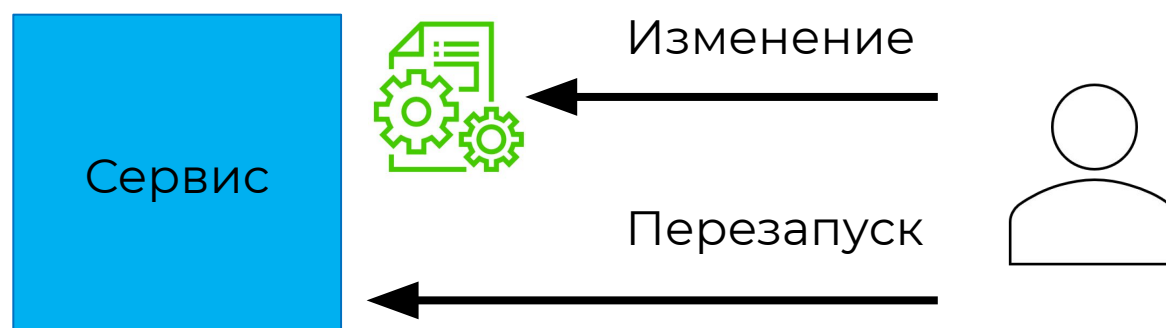
- Управление подпиской зависимых сервисов
- Рост и сокращение экземпляров сервисов

На этом уроке

- Управление конфигурацией
- Разбор примеров применения

Проблема

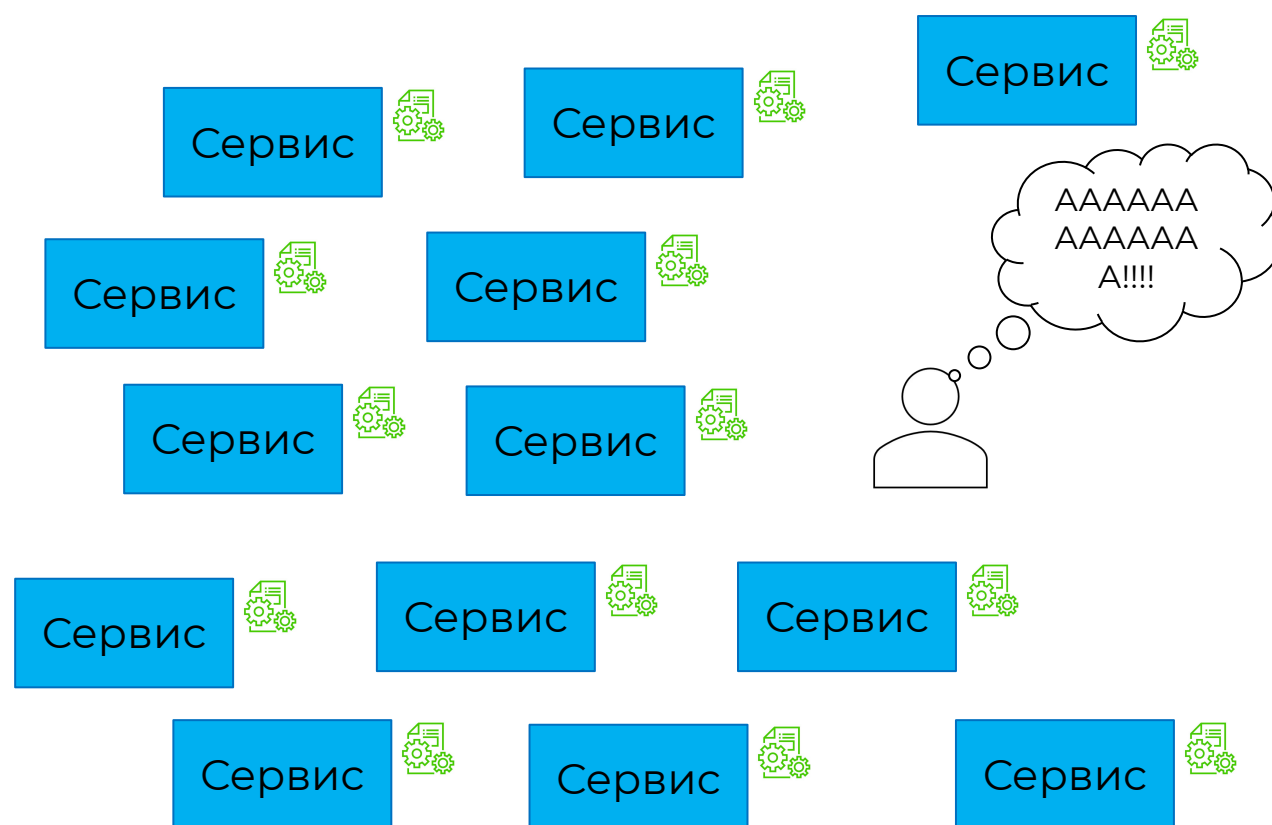
- Изменение файла конфигурации для сервиса
- Перезапуск, чтобы изменения вступили в силу



Можно сделать так, чтобы сервис следил за изменением конфигурации, тогда перезапуск не потребуется.

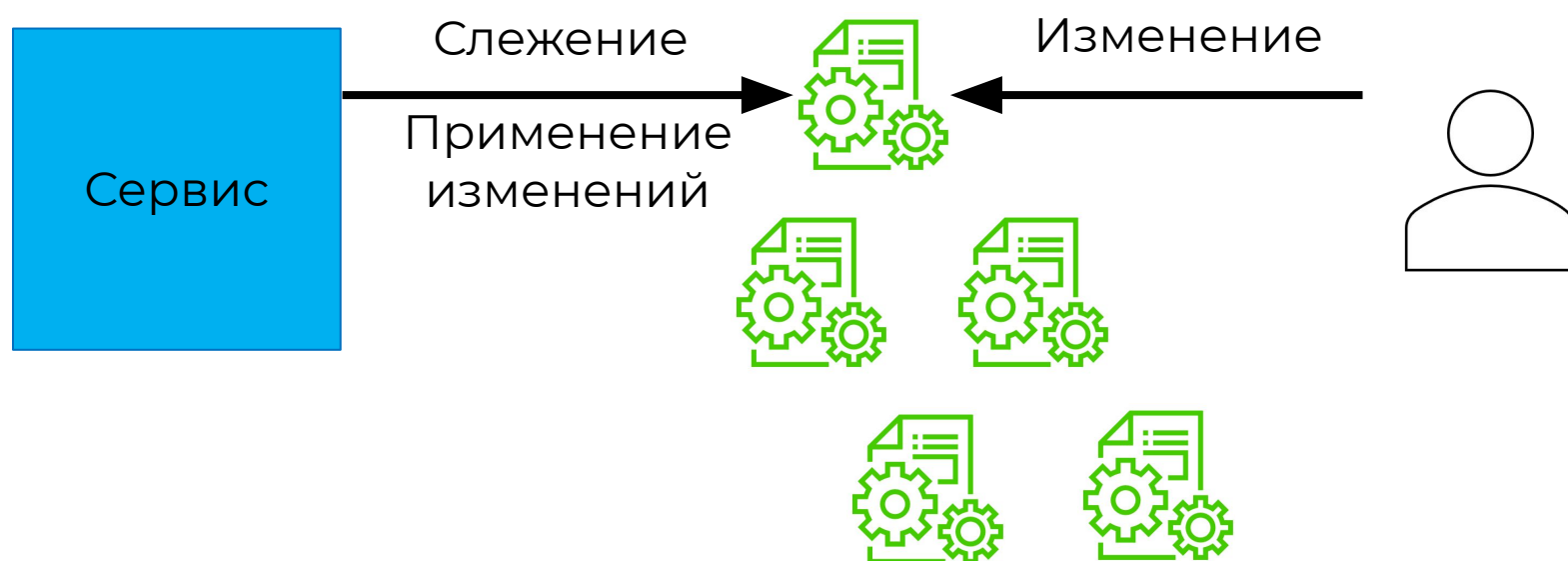
Проблема

- Ручное обновление
- Распределённость конфигов
- Обновление = перезапуск = downtime



Обновление конфигурации

Обновление конфигурации без перезапуска сервиса:



Новые проблемы:

- согласованность конфигураций
- актуальность
- сложные иерархические конфигурации
- конфигурации в зависимости от окружения

Управление конфигурацией

Service Discovery может хранить конфигурации сервисов и высылать их, как только они обновятся (Long polling).



Анатомия Service Discovery

Современные реализации — это целый комплекс продуктов:

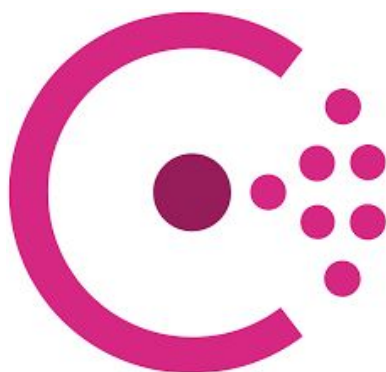
- Service Discovery & Registry
- проверка работоспособности
- безопасность
- управление токенами доступа
- динамическая балансировка
- динамическая конфигурация



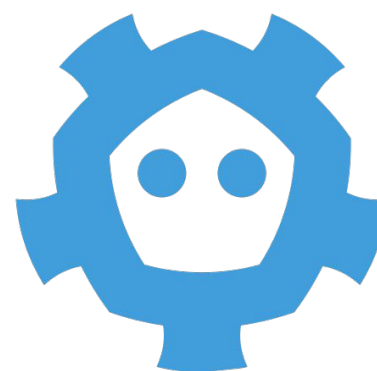
Представители



Zookeeper



Consul



etcd



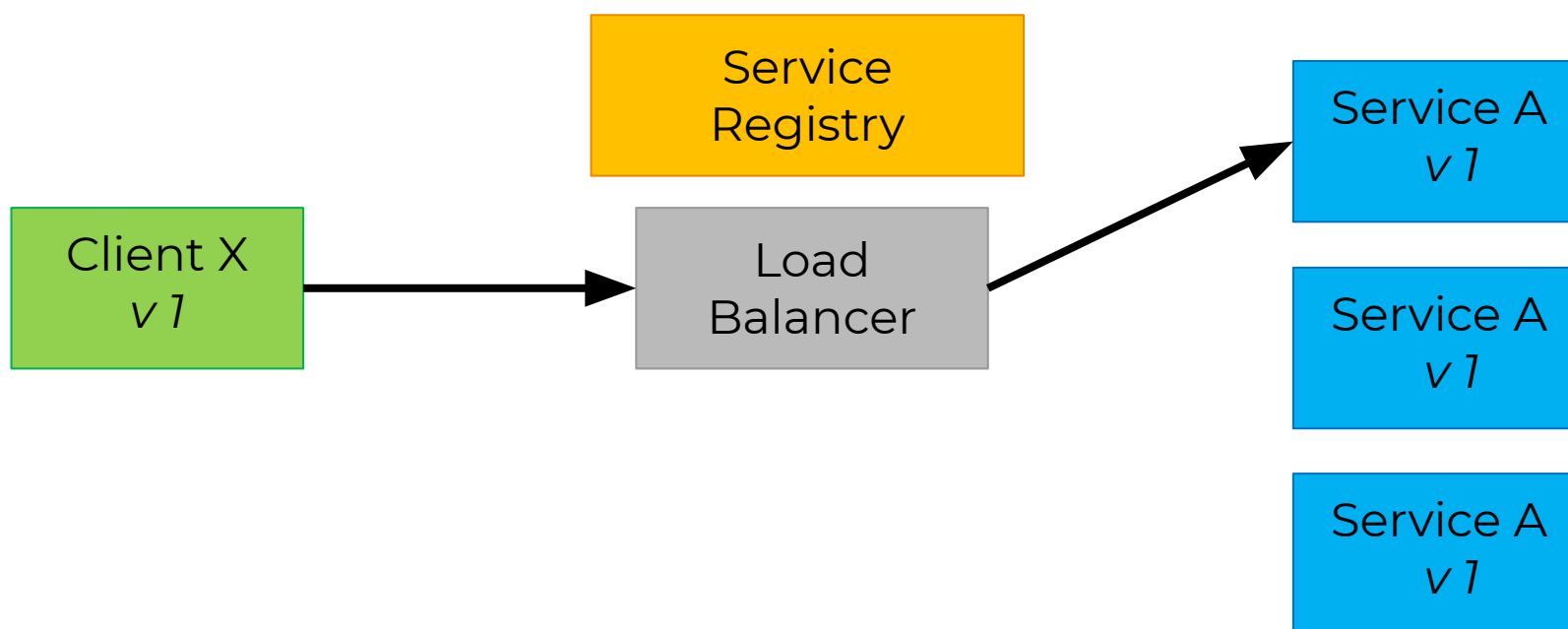
Kubernetes



Istio

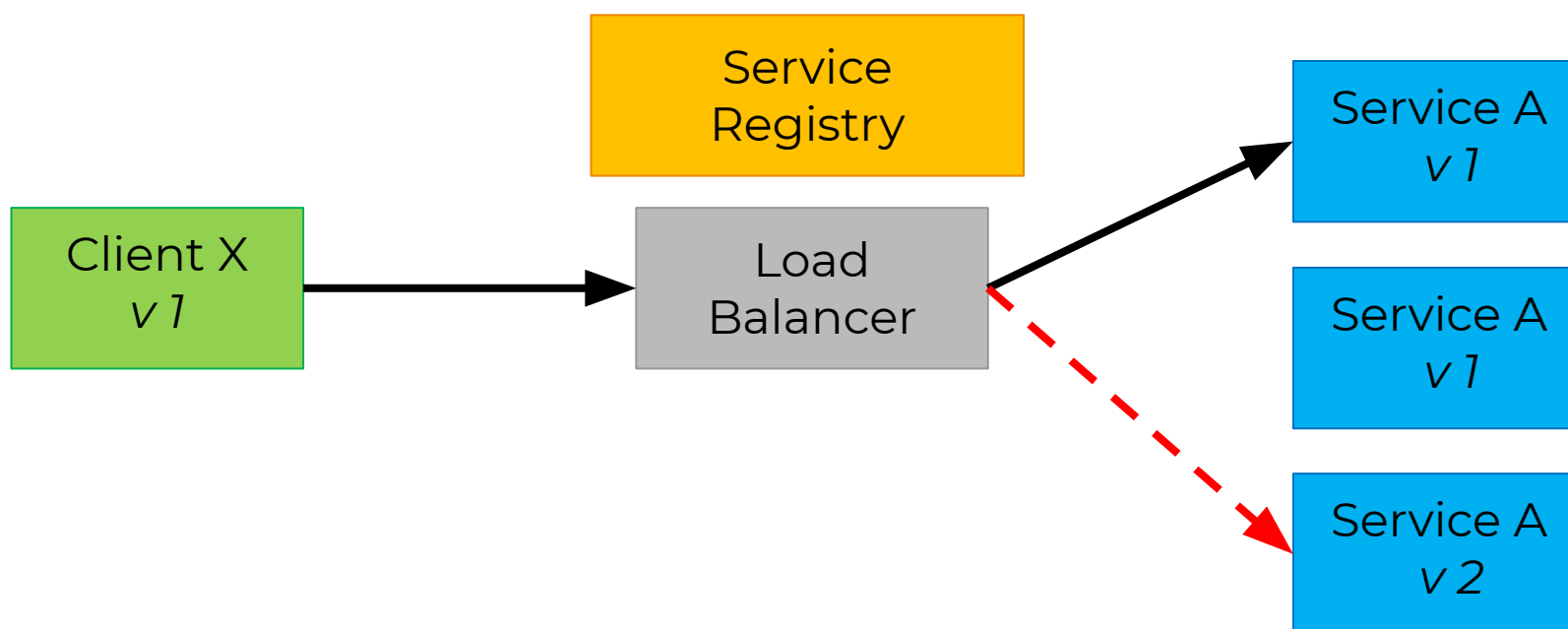
Версии и роутинг

- Управление версиями сервисов
- Правильный роутинг



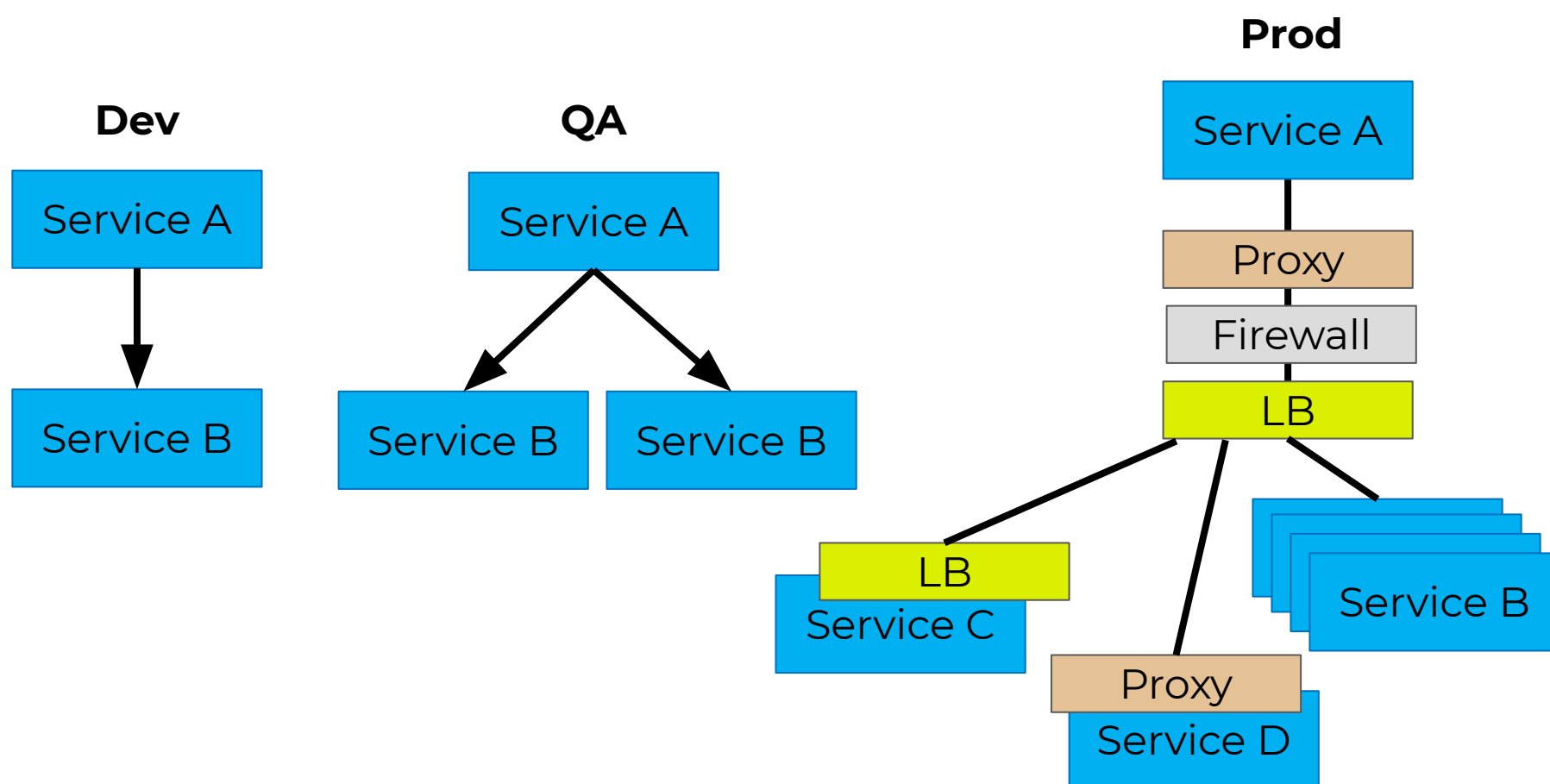
Версии и роутинг

- Управление версиями сервисов
- Правильный роутинг



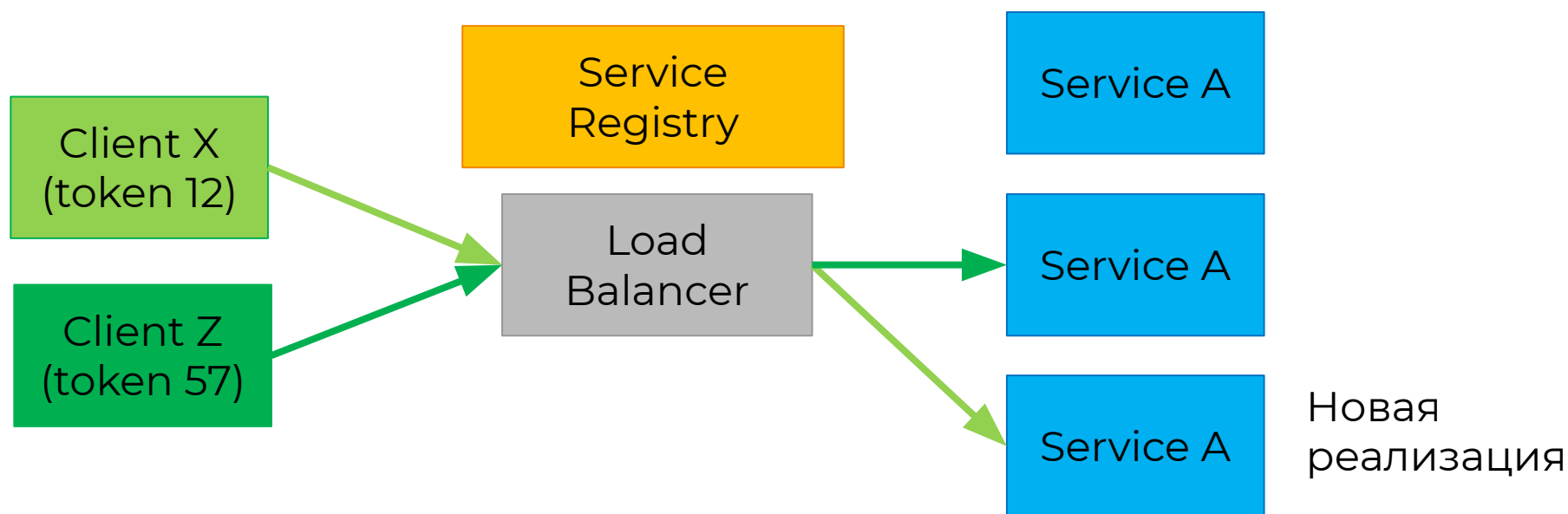
A/B-тестирование

A/B-тестирование позволяет оценить гипотезу, разделяя пользователей по двум версиям, и сравнить результаты на реальном окружении:



A/B-тестирование

A/B-тестирование позволяет оценить гипотезу, разделяя пользователей по двум версиям, и сравнить результаты на реальном окружении:

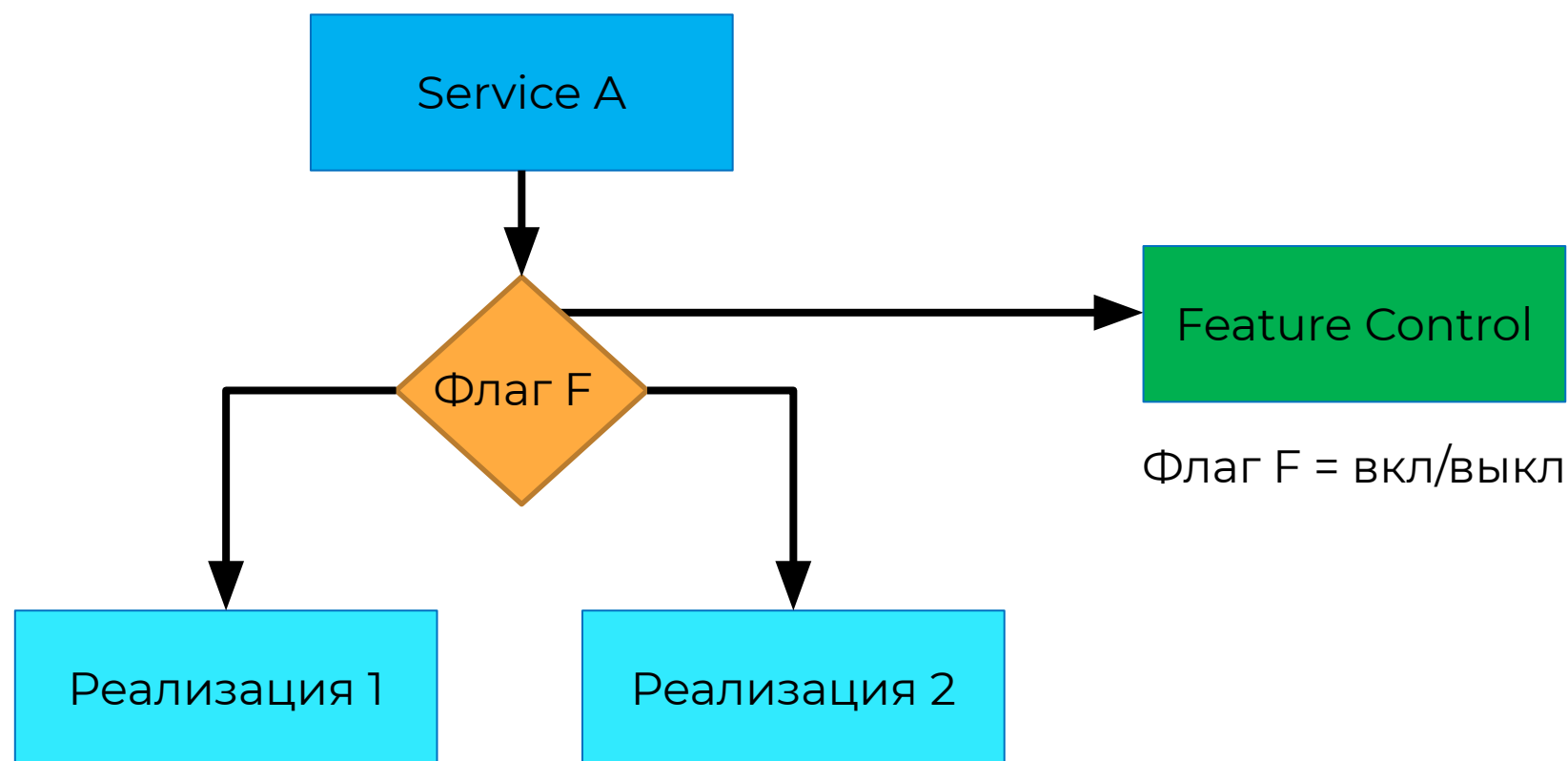


- Используется для маркетинговых целей
- Является практическим использованием Feature Toggling

Feature Toggling

Feature Flags

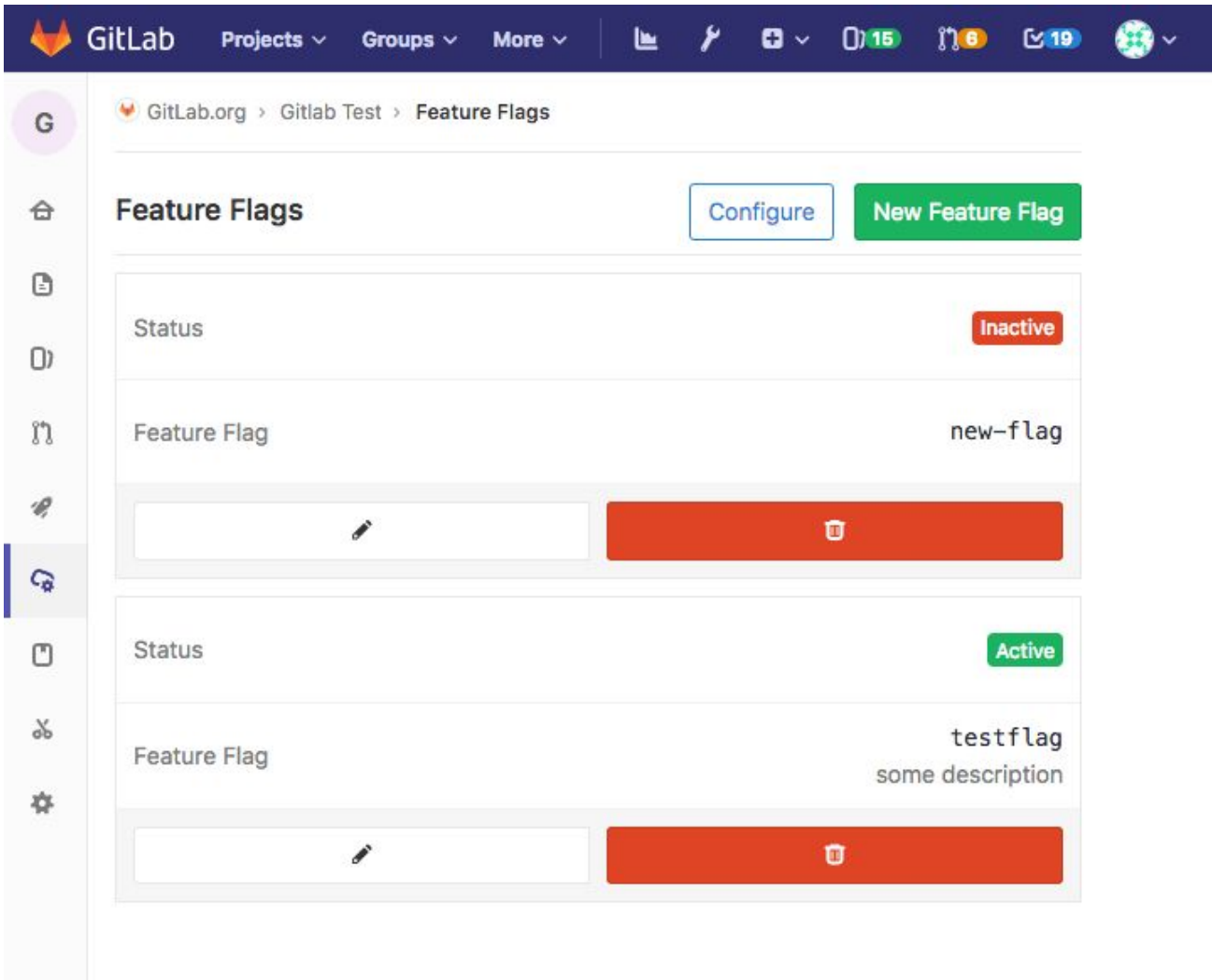
Кондиционная логика, управляемая извне приложения:



Используется для A/B-тестирования, «канареечных» релизов.

Feature Toggling

Управление функционалом в зависимости от окружения:




Feature Toggling

Гибкий выбор пользователей:

- процент от общего количества пользователей
- прямое указание пользователей
- СПИСКИ ИСКЛЮЧЕНИЙ

Target environments

Feature Flag behavior is built up by creating a set of rules to define the status of target environments. A default wildcard rule  for **All Environments** is set, and you are able to add as many rules as you need by choosing environment specs below. You can toggle the behavior for each of your rules to set them **Active** or **Inactive**.

Environment Spec	Status	Rollout Strategy
* (All Environments)	<input checked="" type="checkbox"/>	All users
<input type="text" value="production"/>	<input type="checkbox"/>	All users
<input type="text" value="staging"/>	<input checked="" type="checkbox"/>	Percent rollout (logged in users) 15 %
<input type="text" value="Search an environment spec"/>	<input type="checkbox"/>	All users

Save changes

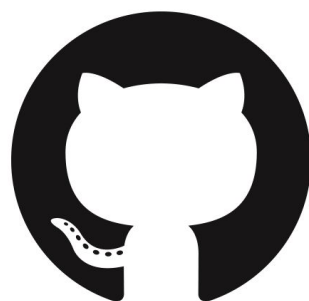
Cancel

Представители

- Почти все CI-/CD-инструменты
- Специализированные продукты



GitLab



GitHub



Togglz



Optimizely

Выводы

- Ручное управление и контроль конфигураций чреваты ошибками и требуют много времени
- Проектируйте сервисы, которые могут читать и применять конфиг без перезагрузки
- Специализированные инструменты заметно сокращают усилия по поддержанию согласованной конфигурации сервисов
- Конфигурация — это не только строки соединений, с их помощью можно управлять поведением приложений

Skillbox

Что дальше?

Шаблон Chassis.

Skillbox

**Спасибо
за внимание!**