

Обновление приложений

Стратегии обновлений

Кирилл Касаткин
DevOps-инженер, Renue



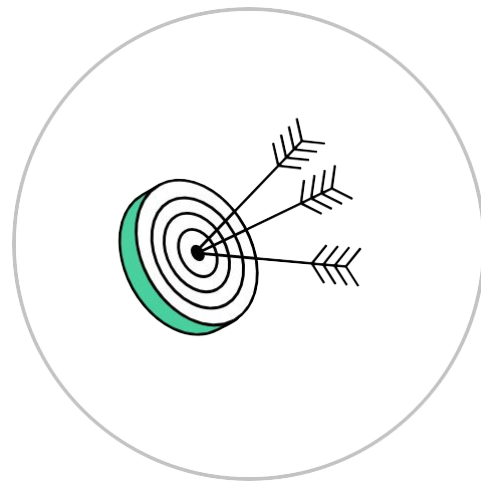
Кирилл Касаткин

DevOps-инженер, Renuе



Цели занятия

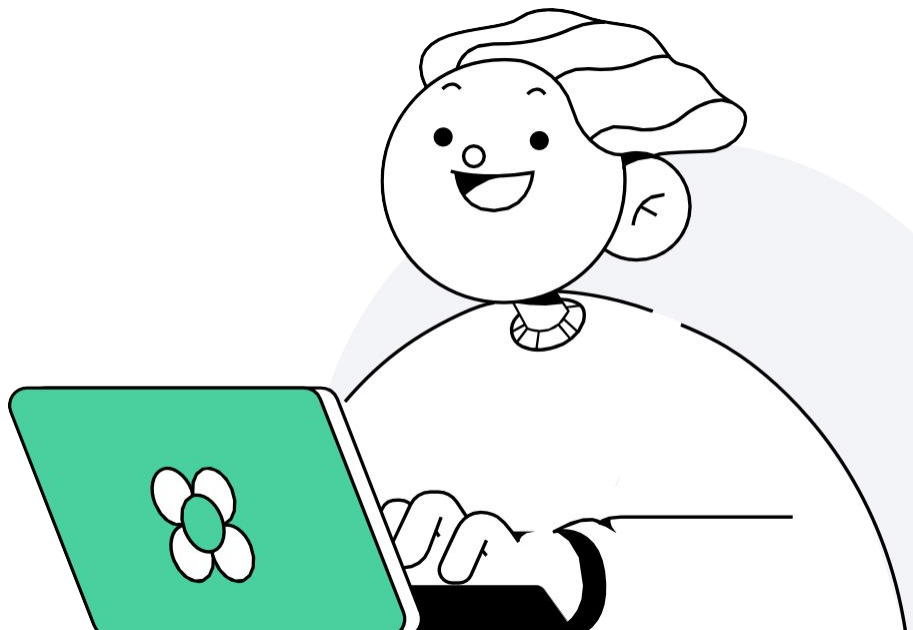
- Изучить стратегии обновления приложений
- Рассмотреть принципы реализации стратегий



План занятия

- 1 Rolling update
- 2 Recreate
- 3 Blue-green
- 4 Canary
- 5 A/B-тест (dark launch)
- 6 Итоги
- 7 Домашнее задание

*Нажми на нужный раздел для перехода



Rolling update

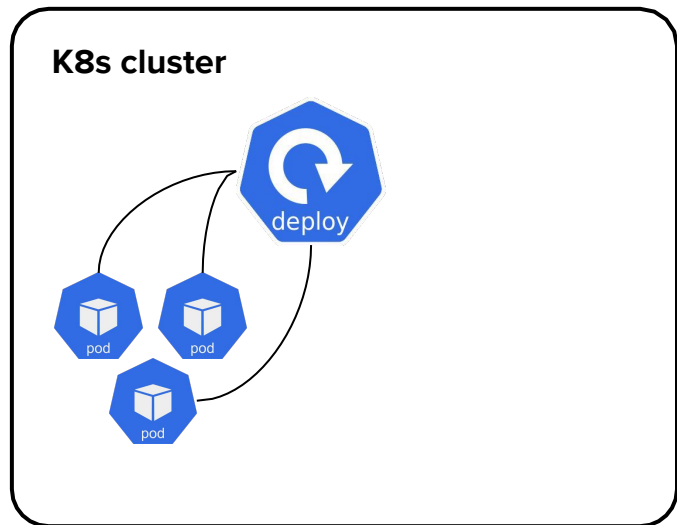


1

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.

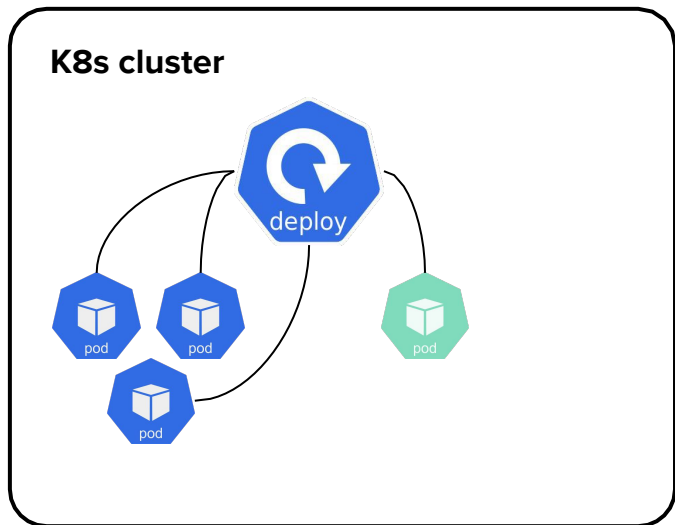


- Вы указываете образ новой версии deployment (развёртывания)

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.

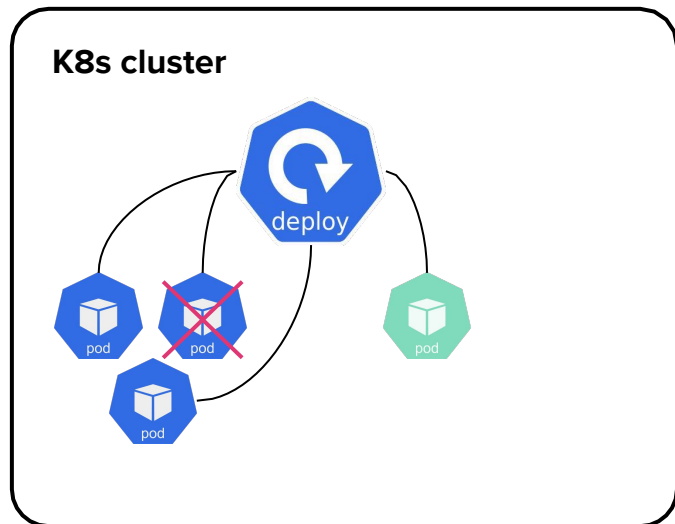


- Вы указываете образ новой версии deployment (развёртывания)
- K8s поднимает несколько pod с новой версией и дожидается их готовности к работе

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.

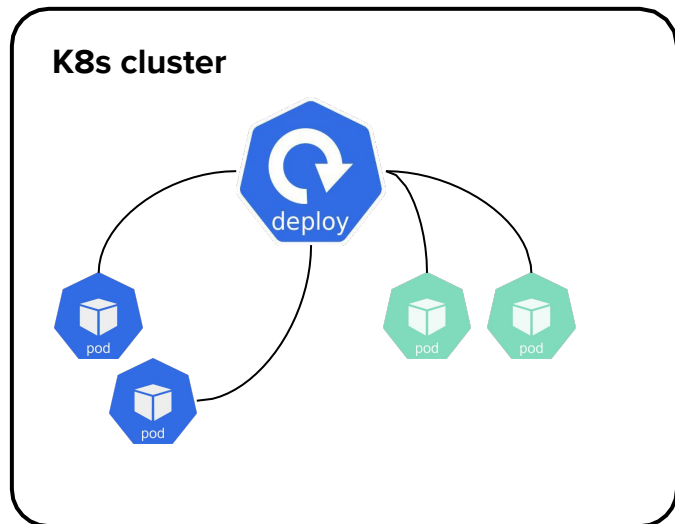


- Вы указываете образ новой версии deployment (развёртывания)
- K8s поднимает несколько pod с новой версией и дожидается их готовности к работе
- Затем несколько старых подов сворачиваются, и процесс повторяется, пока все поды не будут заменены

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.

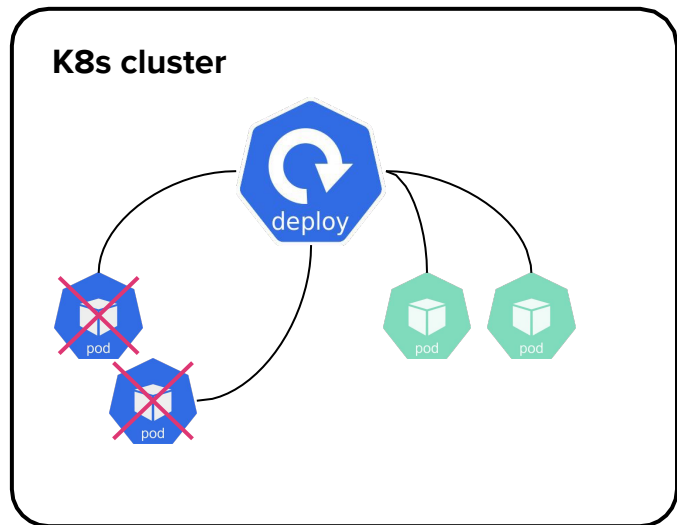


- Вы указываете образ новой версии deployment (развёртывания)
- K8s поднимает несколько pod с новой версией и дожидается их готовности к работе
- Затем несколько старых подов сворачиваются, и процесс повторяется, пока все поды не будут заменены

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.

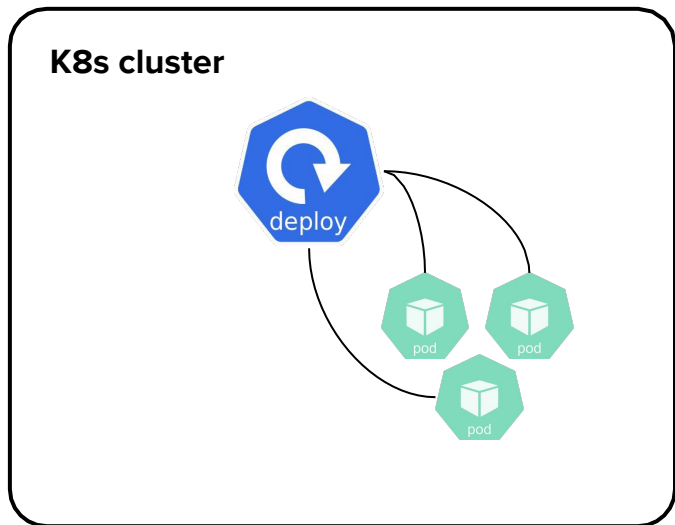


- Вы указываете образ новой версии deployment (развёртывания)
- K8s поднимает несколько pod с новой версией и дожидается их готовности к работе
- Затем несколько старых подов сворачиваются, и процесс повторяется, пока все поды не будут заменены

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.

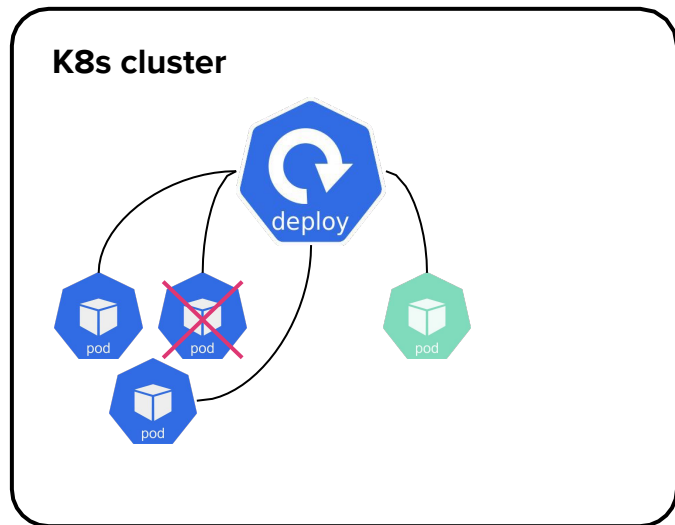


- Вы указываете образ новой версии deployment (развёртывания)
- K8s поднимает несколько pod с новой версией и дожидается их готовности к работе
- Затем несколько старых подов сворачиваются, и процесс повторяется, пока все поды не будут заменены

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.

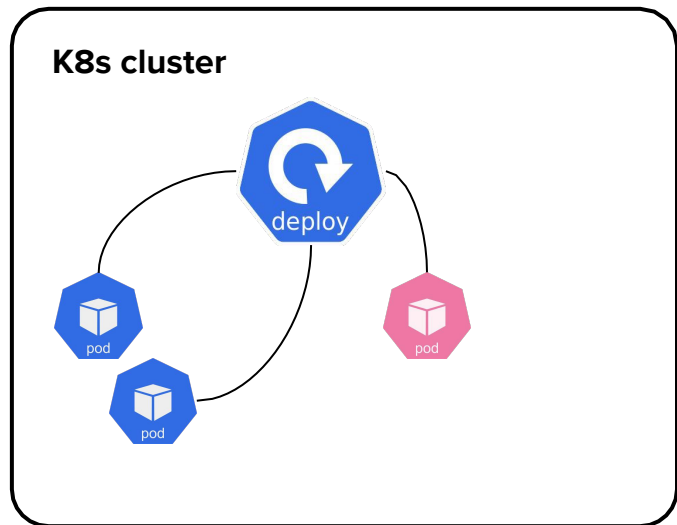


- Вы указываете образ новой версии deployment (развёртывания)
- K8s поднимает несколько pod с новой версией и дожидается их готовности к работе
- Затем несколько старых подов сворачиваются, и процесс повторяется, пока все поды не будут заменены
- Если возникнут проблемы, дальнейшие поды не обновятся и процесс приостановится. Pod со старой версией останутся в работе

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.

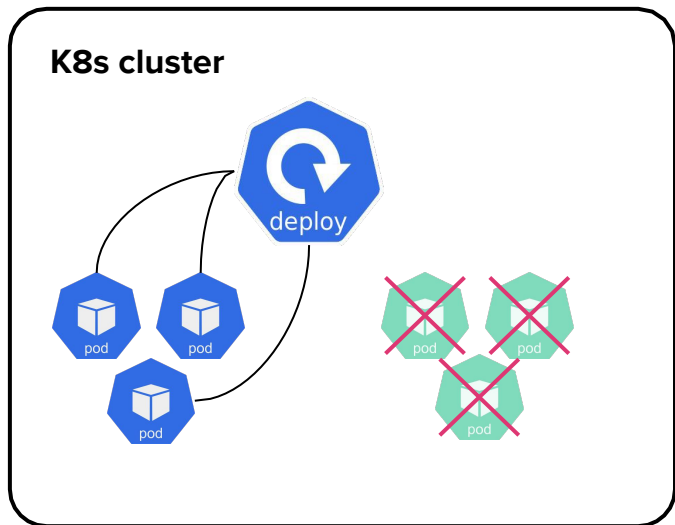


- Вы указываете образ новой версии deployment (развёртывания)
- K8s поднимает несколько pod с новой версией и дожидается их готовности к работе
- Затем несколько старых подов сворачиваются, и процесс повторяется, пока все поды не будут заменены
- Если возникнут проблемы, дальнейшие поды не обновятся и процесс приостановится. Pod со старой версией останутся в работе

Rolling update — постепенное обновление

Это стандартная стратегия развёртывания в Kubernetes.

Она постепенно, один за другим заменяет поды со старой версией приложения на поды с новой версией без простоя кластера.



- Вы указываете образ новой версии deployment (развёртывания)
- K8s поднимает несколько pod с новой версией и дожидается их готовности к работе
- Затем несколько старых подов сворачиваются, и процесс повторяется, пока все поды не будут заменены
- Если возникнут проблемы, дальнейшие поды не обновятся и процесс приостановится. Pod со старой версией останутся в работе
- Есть возможность вернуться к предыдущей версии — так же постепенно

Пример конфигурации

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  strategy:
    type: RollingUpdate
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels: app:
        nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
          ports:
            - containerPort: 80
```

- **maxSurge** — максимальное число подов, которое может быть создано свыше желаемого количества. Может быть выражено в абсолютном значении или в процентах. По умолчанию — 25%
- **maxUnavailable** — максимальное число подов, которые могут быть недоступны в процессе обновления. Может быть выражено в абсолютном значении или в процентах. По умолчанию — 25%

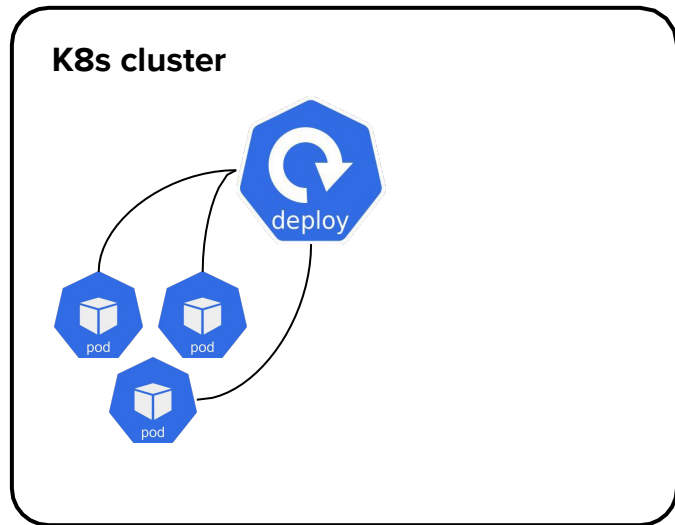
Recreate



2

Recreate — повторное создание

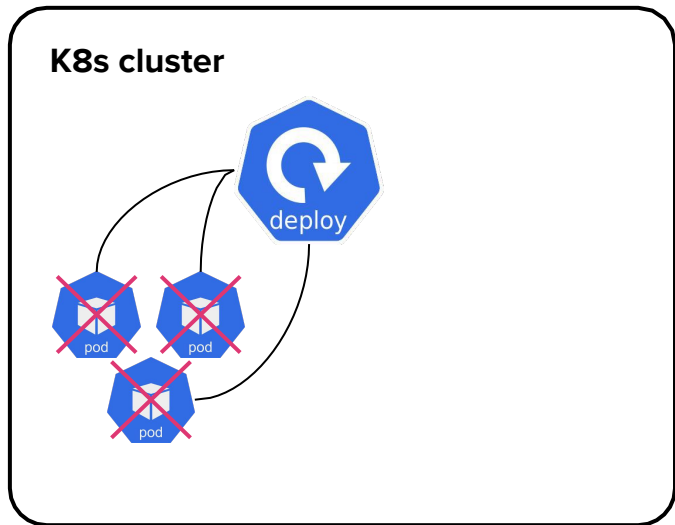
Стратегия, когда все старые поды одновременно удаляются и заменяются новыми.



- Вы указываете образ новой версии deployment

Recreate — повторное создание

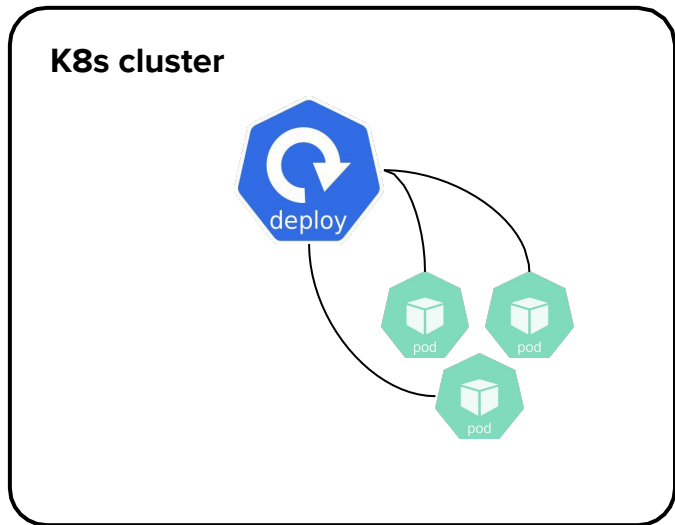
Стратегия, когда все старые поды одновременно удаляются и заменяются новыми.



- Вы указываете образ новой версии deployment
- K8s уничтожает все поды со старой версией

Recreate — повторное создание

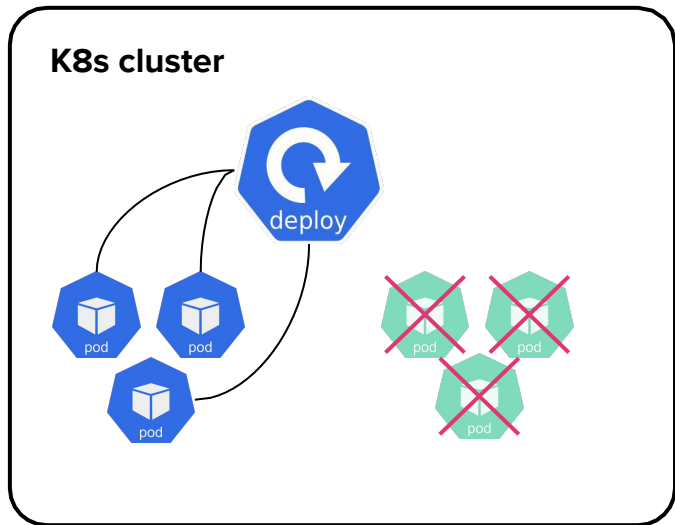
Стратегия, когда все старые поды одновременно удаляются и заменяются новыми.



- Вы указываете образ новой версии deployment
- K8s уничтожает все поды со старой версией
- Затем добавляются поды с новой версией

Recreate — повторное создание

Стратегия, когда все старые поды одновременно удаляются и заменяются новыми.



- Вы указываете образ новой версии deployment
- K8s уничтожает все поды со старой версией
- Затем добавляются поды с новой версией
- Возврат к предыдущей версии происходит мгновенно, разом

Пример конфигурации

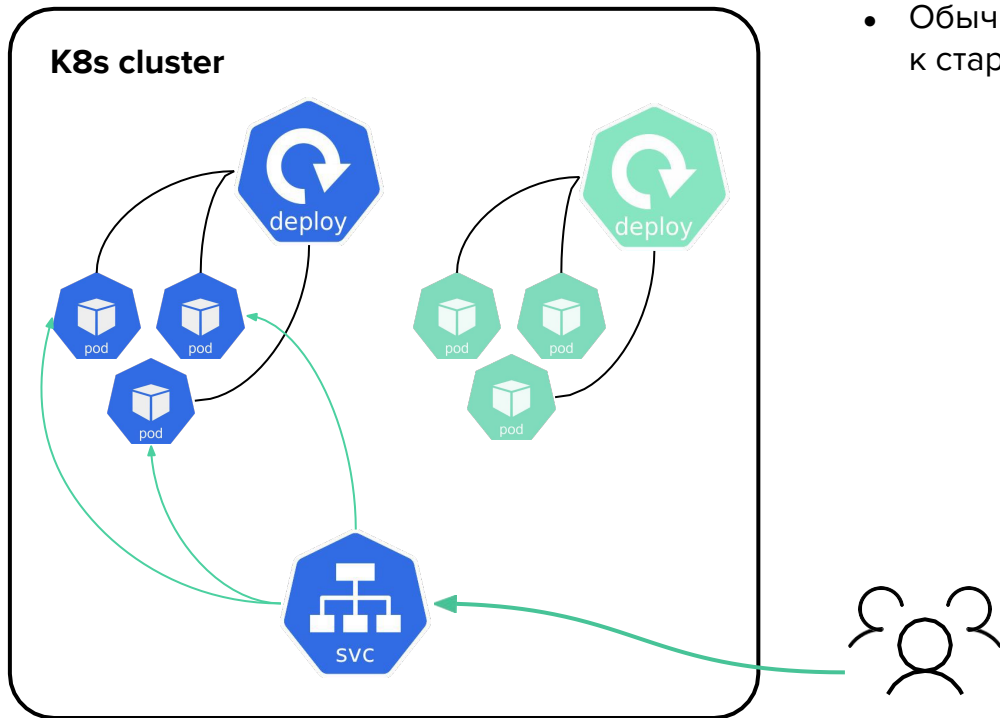
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  strategy:
    type: Recreate
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels: app:
        nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.14.2
        ports:
        - containerPort: 80
```

Blue-green

3

Blue-green — создание копии и переброс трафика

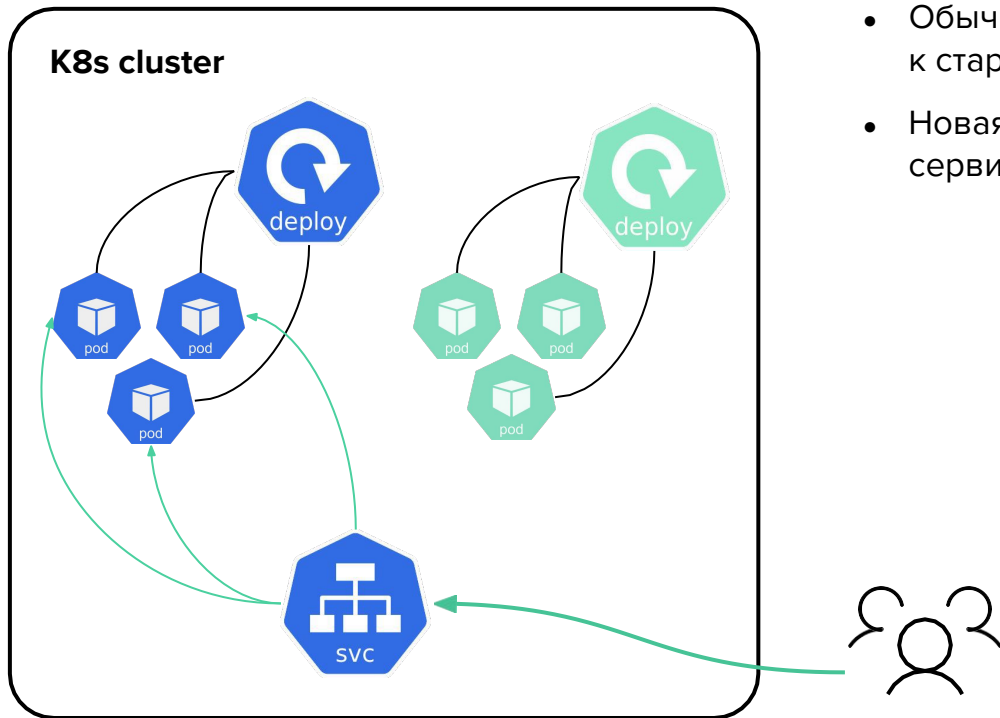
Одновременное развёртывание старой и новой версий приложения.



- Обычные пользователи получают доступ к старой версии

Blue-green — создание копии и переброс трафика

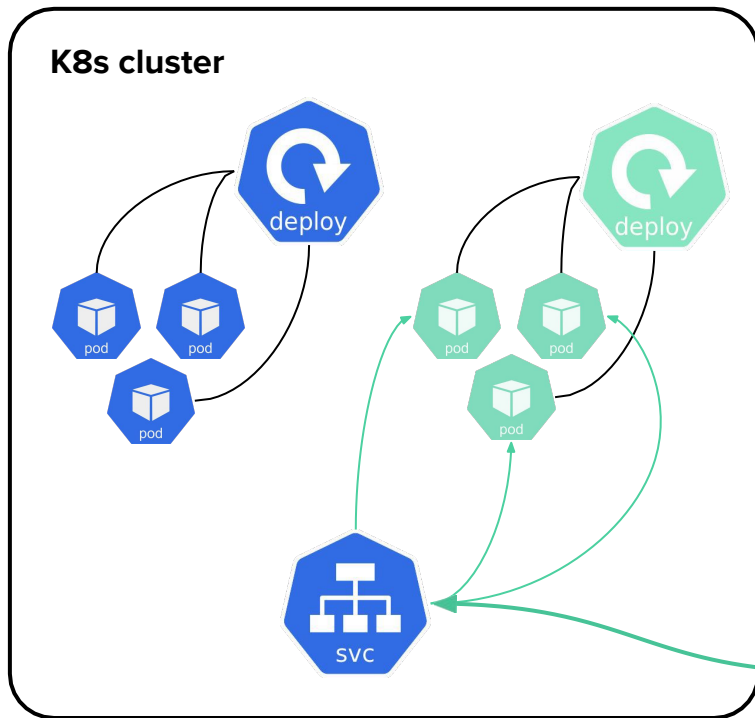
Одновременное развёртывание старой и новой версий приложения.



- Обычные пользователи получают доступ к старой версии
- Новая версия доступна для тестов через отдельный сервис или прямой проброс портов

Blue-green — создание копии и переброс трафика

Одновременное развёртывание старой и новой версий приложения.



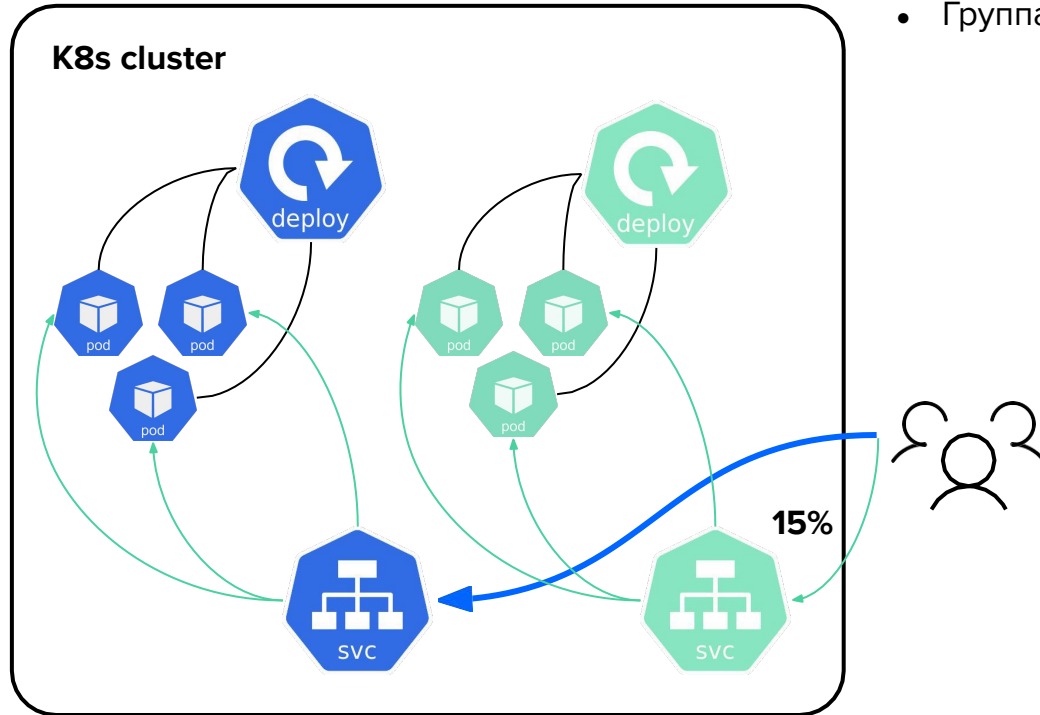
- Обычные пользователи получают доступ к старой версии
- Новая версия доступна для тестов через отдельный сервис или прямой проброс портов
- Когда новая версия протестирована, сервис переключается на неё, а старая версия сворачивается

Canary

4

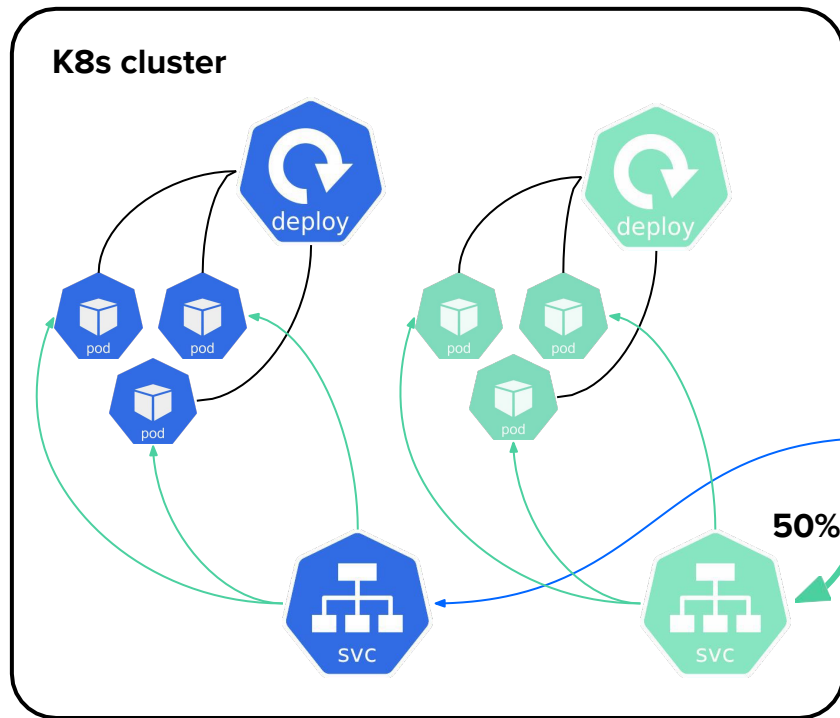
Canary — частичное обновление и проверка

Используется, чтобы протестировать новую функциональность на отдельной группе пользователей.



Canary — частичное обновление и проверка

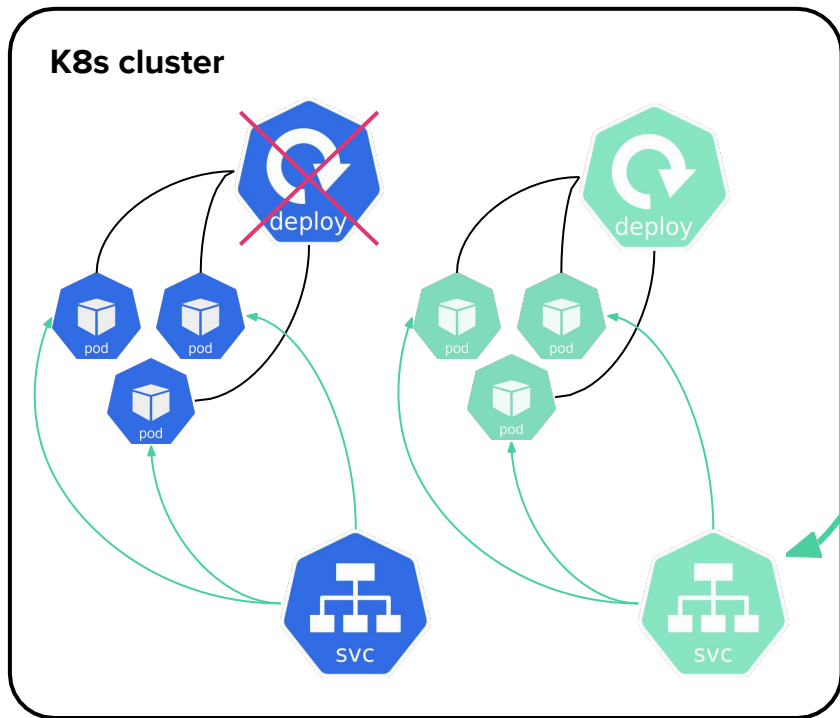
Используется, чтобы протестировать новую функциональность на отдельной группе пользователей.



- Группа выделяется по определённому признаку
- При этом тест не должен затрагивать работу основной версии приложения и его пользователей

Canary — частичное обновление и проверка

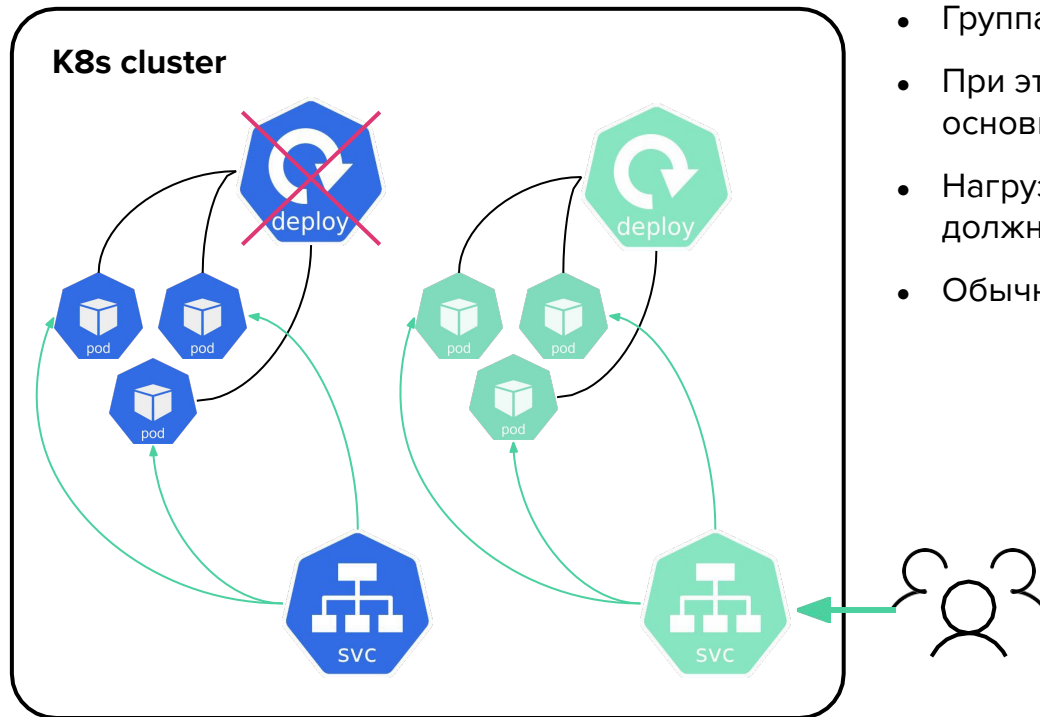
Используется, чтобы протестировать новую функциональность на отдельной группе пользователей.



- Группа выделяется по определённому признаку
- При этом тест не должен затрагивать работу основной версии приложения и его пользователей
- Нагрузка между двумя версиями приложения должна распределяться предсказуемо

Canary — частичное обновление и проверка

Используется, чтобы протестировать новую функциональность на отдельной группе пользователей.



- Группа выделяется по определённому признаку
- При этом тест не должен затрагивать работу основной версии приложения и его пользователей
- Нагрузка между двумя версиями приложения должна распределяться предсказуемо
- Обычно применяется при работе с backend

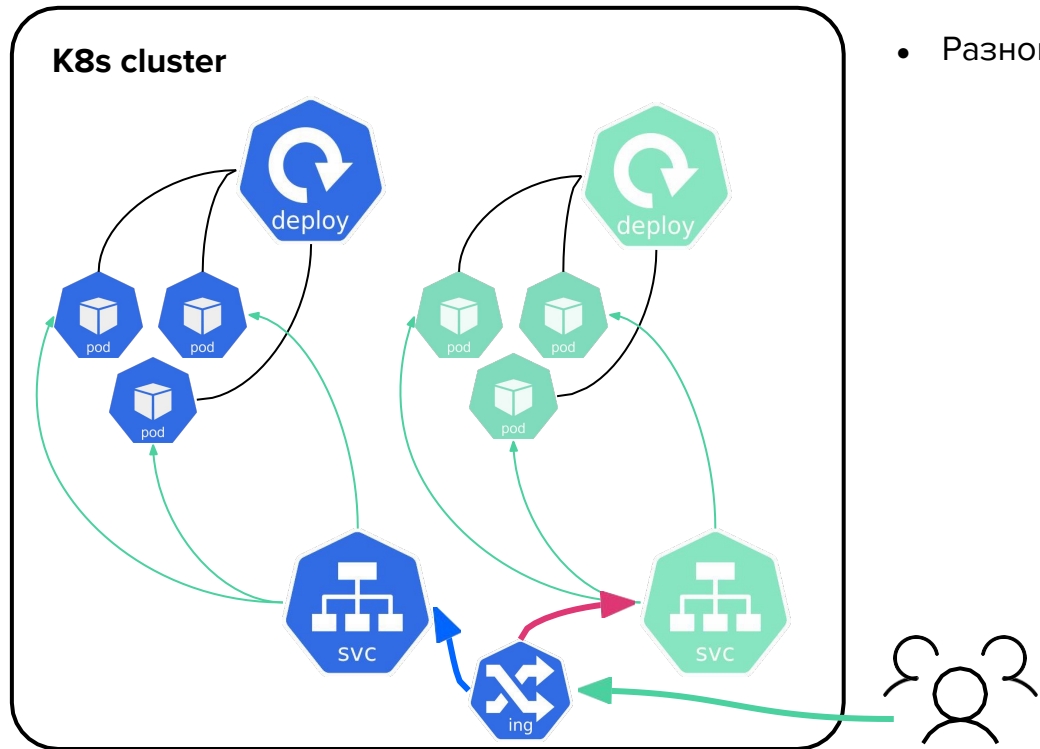
A/B-тест (dark launch)



5

A/B-тестирование

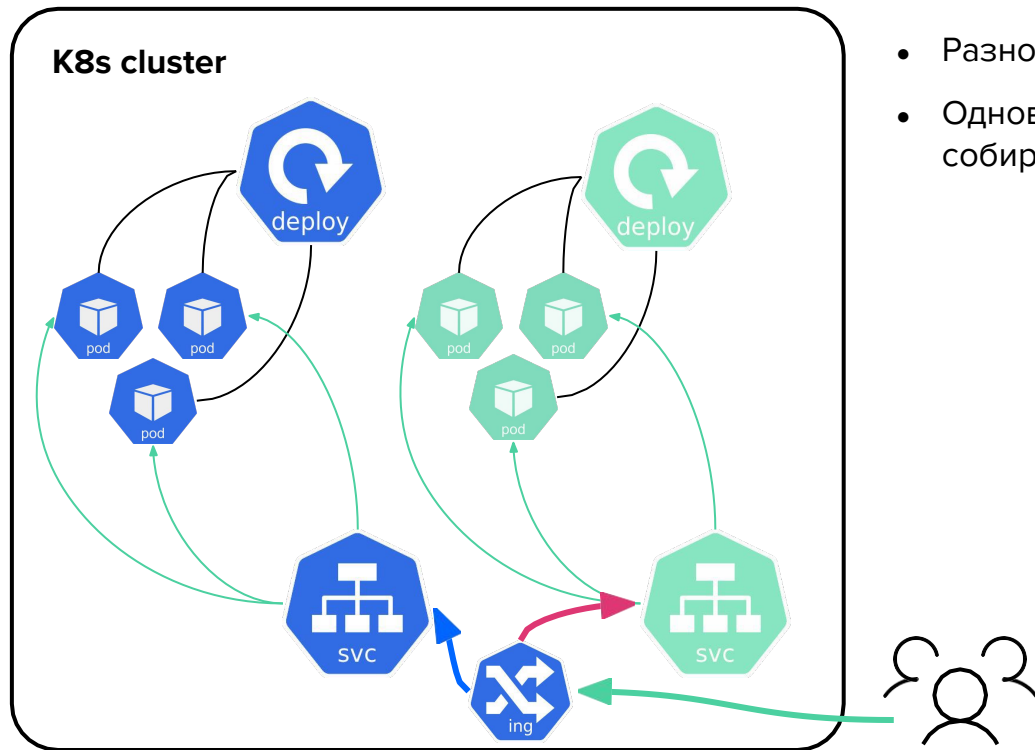
Применяется, когда есть две версии приложения и мы не уверены на 100%, какая из них улучшит взаимодействие с пользователем.



- Разновидность канареечного тестирования

A/B-тестирование

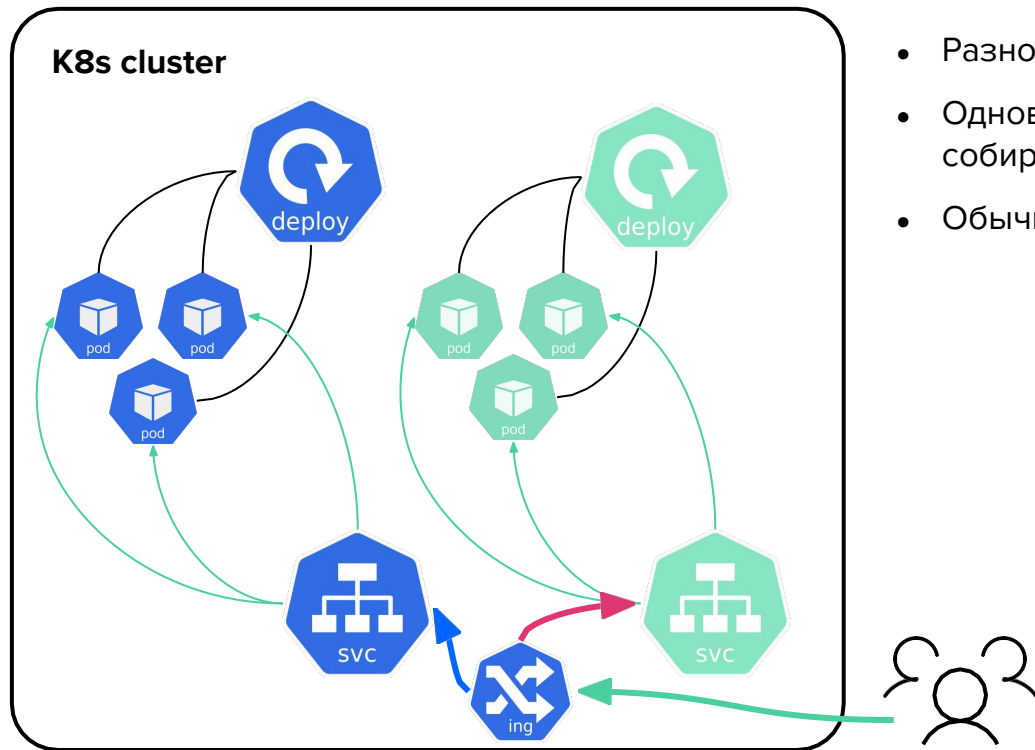
Применяется, когда есть две версии приложения и мы не уверены на 100%, какая из них улучшит взаимодействие с пользователем.



- Разновидность канареечного тестирования
- Одновременно запускаются обе версии и собираются метрики пользователей

A/B-тестирование

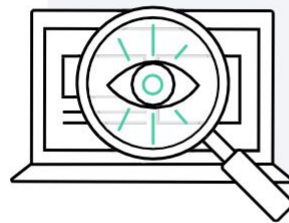
Применяется, когда есть две версии приложения и мы не уверены на 100%, какая из них улучшит взаимодействие с пользователем.



- Разновидность канареечного тестирования
- Одновременно запускаются обе версии и собираются метрики пользователей
- Обычно применяется при работе с frontend

Демонстрация работы

UpdateStrategy



Итоги

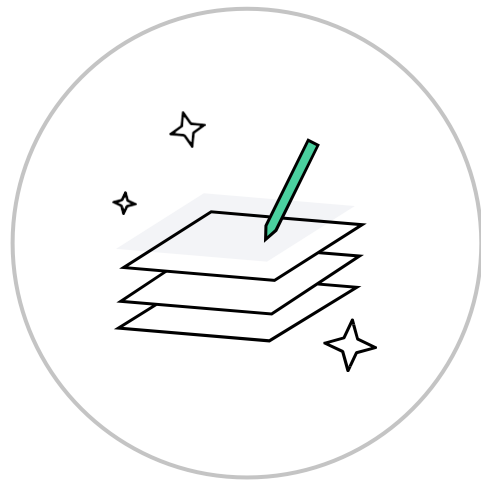
- Узнали, какие есть стратегии обновления приложений
- Выяснили, как работает каждая стратегия
- Рассмотрели примеры манифестов объектов K8s
- Подключились к кластеру и наблюдали за работой объектов, изученных на занятии



Домашнее задание

Давайте посмотрим ваше домашнее задание.

- 1 Вопросы по домашней работе задавайте в чате группы
- 2 Задачи можно сдавать по частям
- 3 Зачёт по домашней работе ставится после того, как приняты все задачи



**Задавайте вопросы
и пишите отзыв о лекции**

