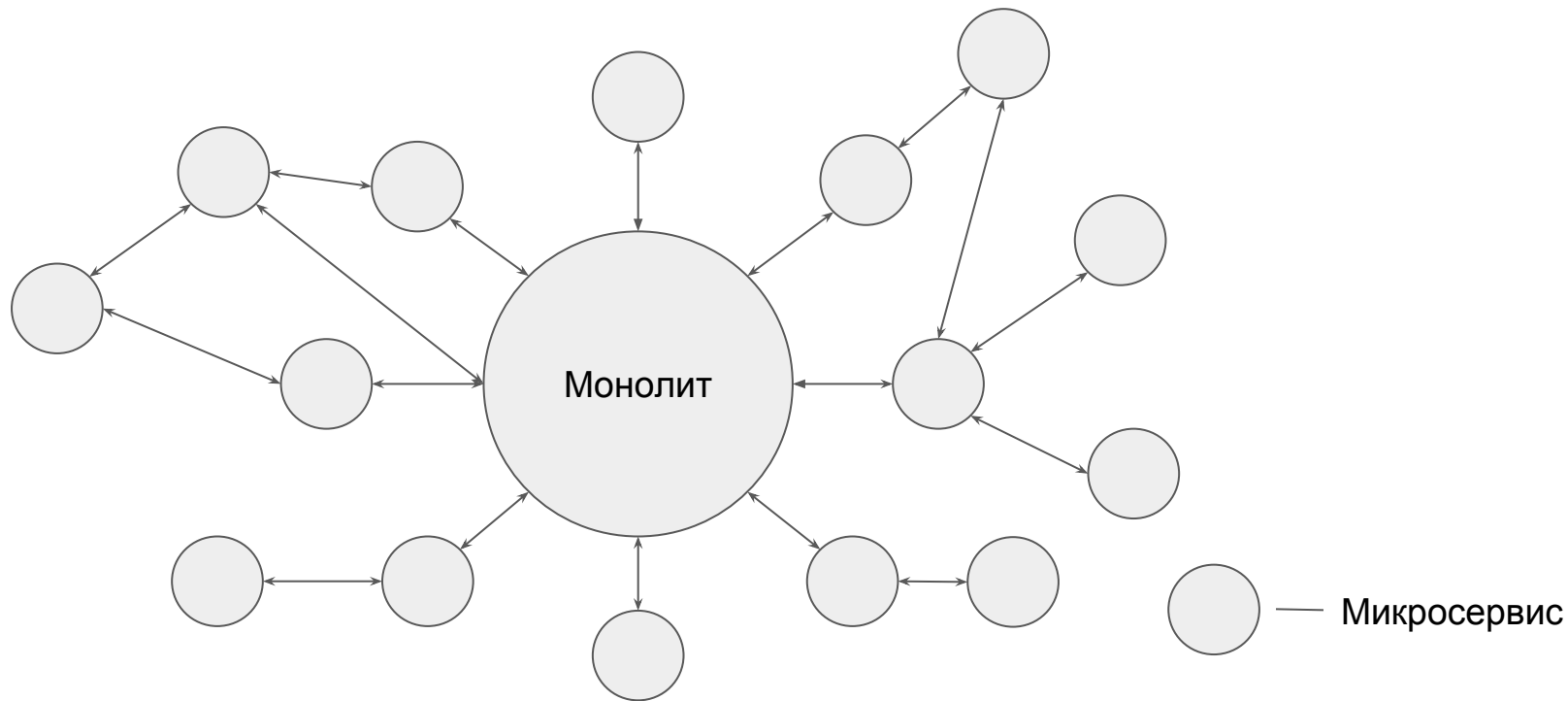


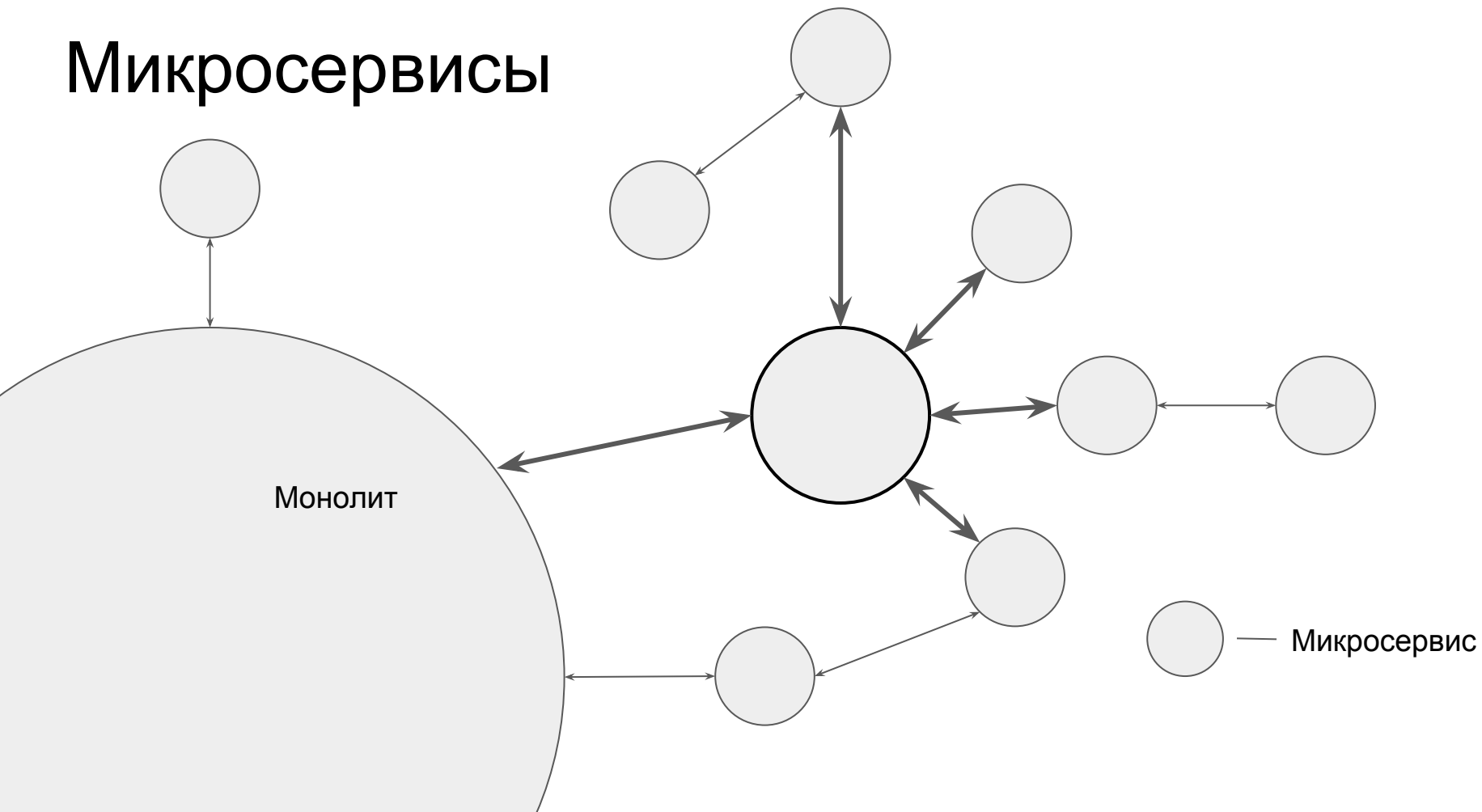
Реализация Consumer-Driven Contracts подхода для тестирования микросервисов в Авито

Фрол Крючков

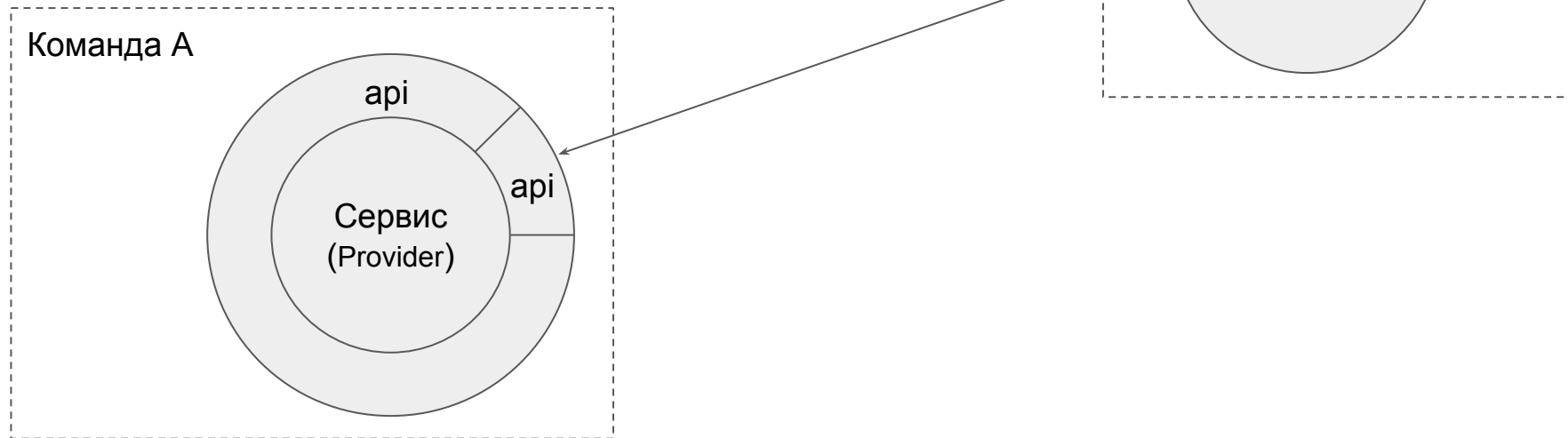
Микросервисы



Микросервисы

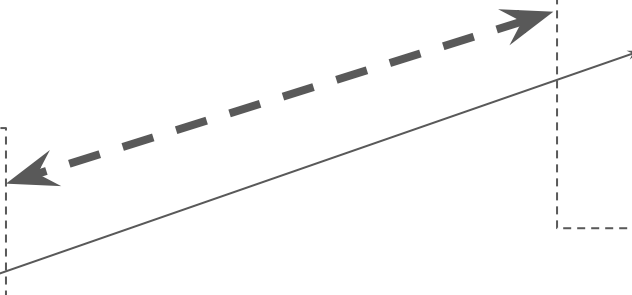
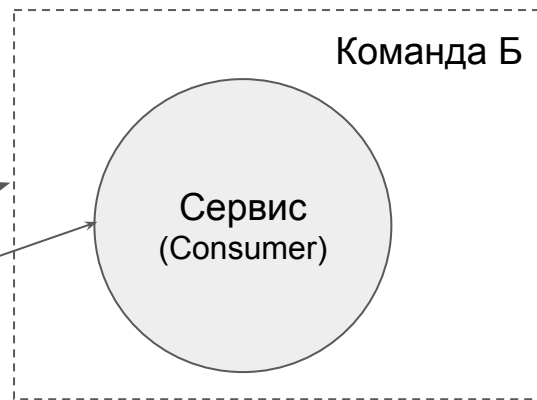
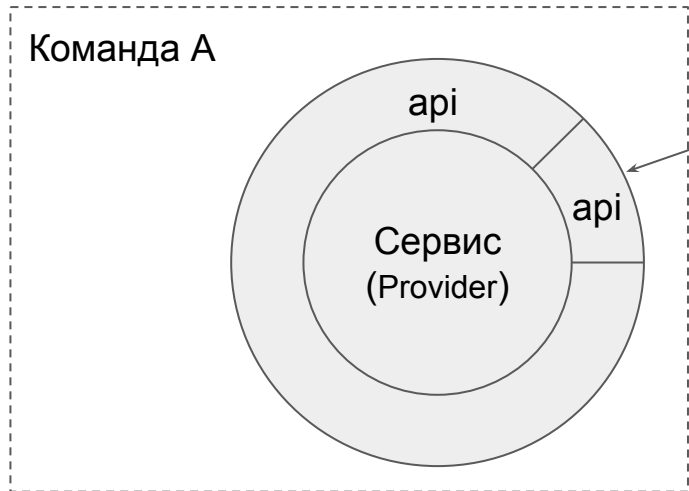


Микросервисы



Микросервисы

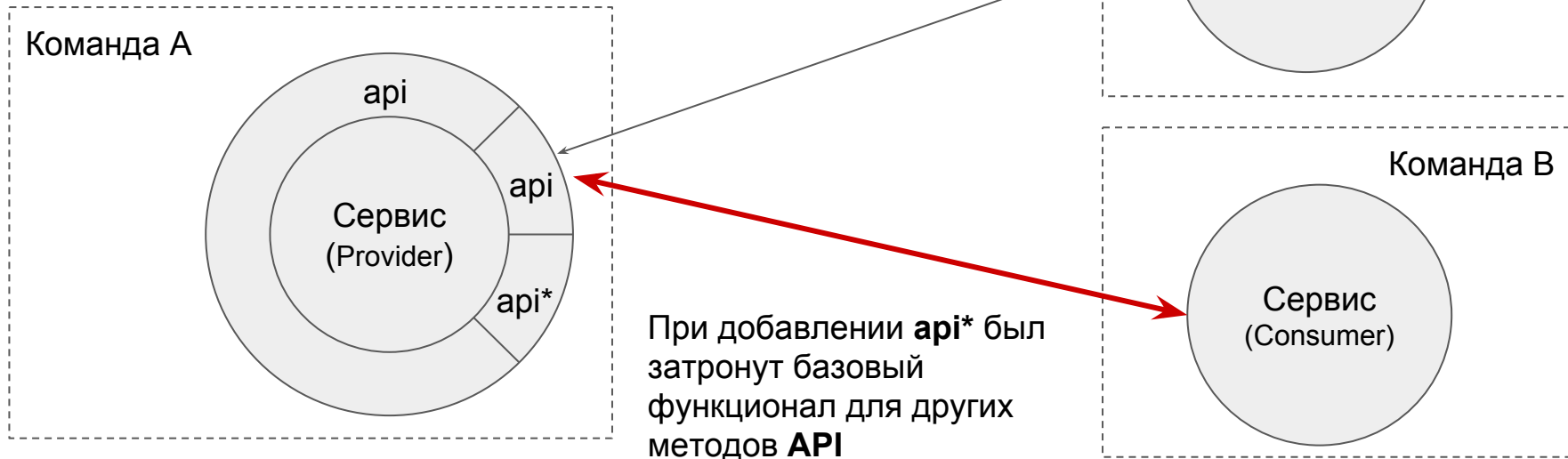
Запрос **команды Б** на
расширение/модификацию
API сервиса **команды А**



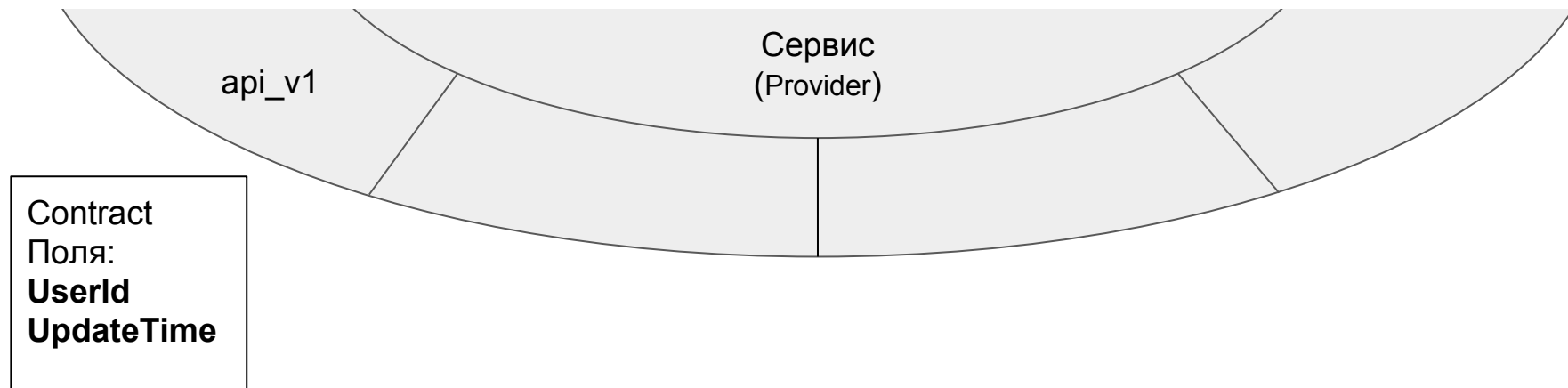
Микросервисы



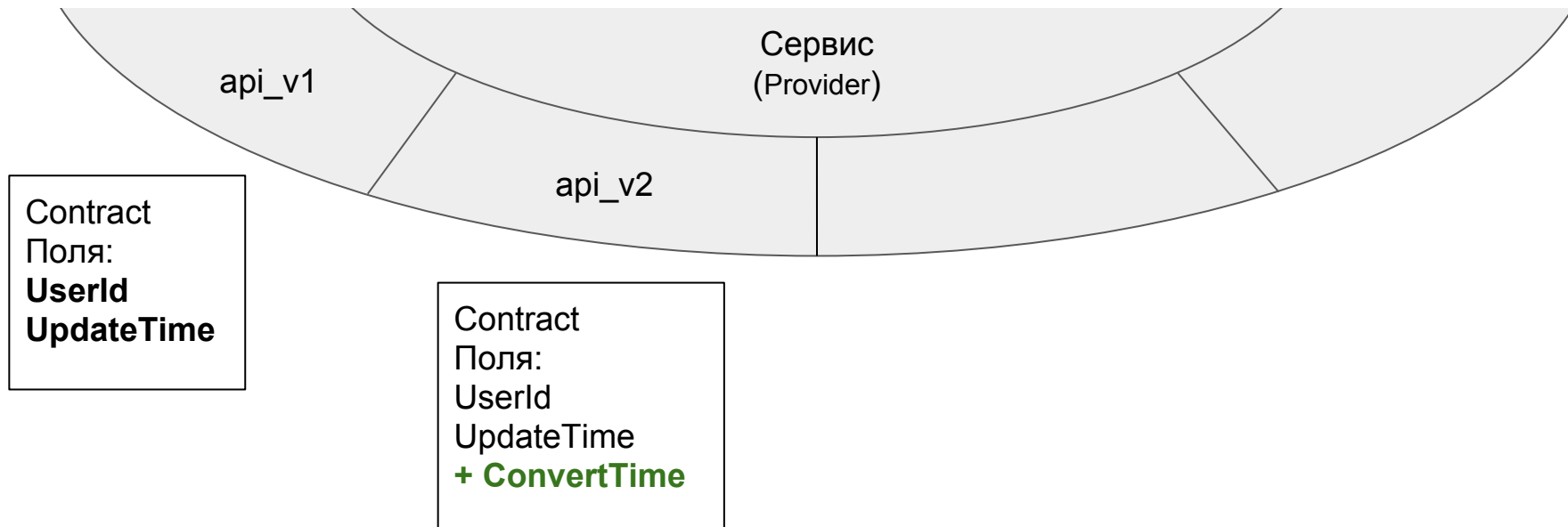
Микросервисы



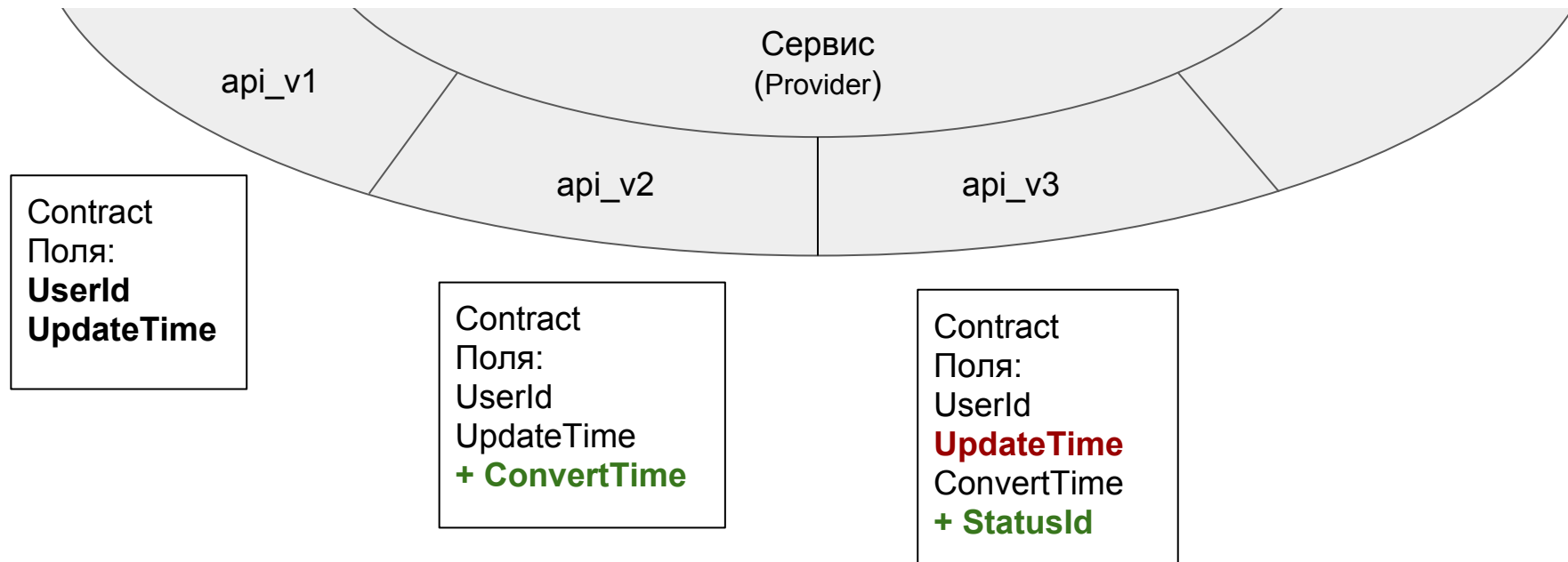
Микросервисы



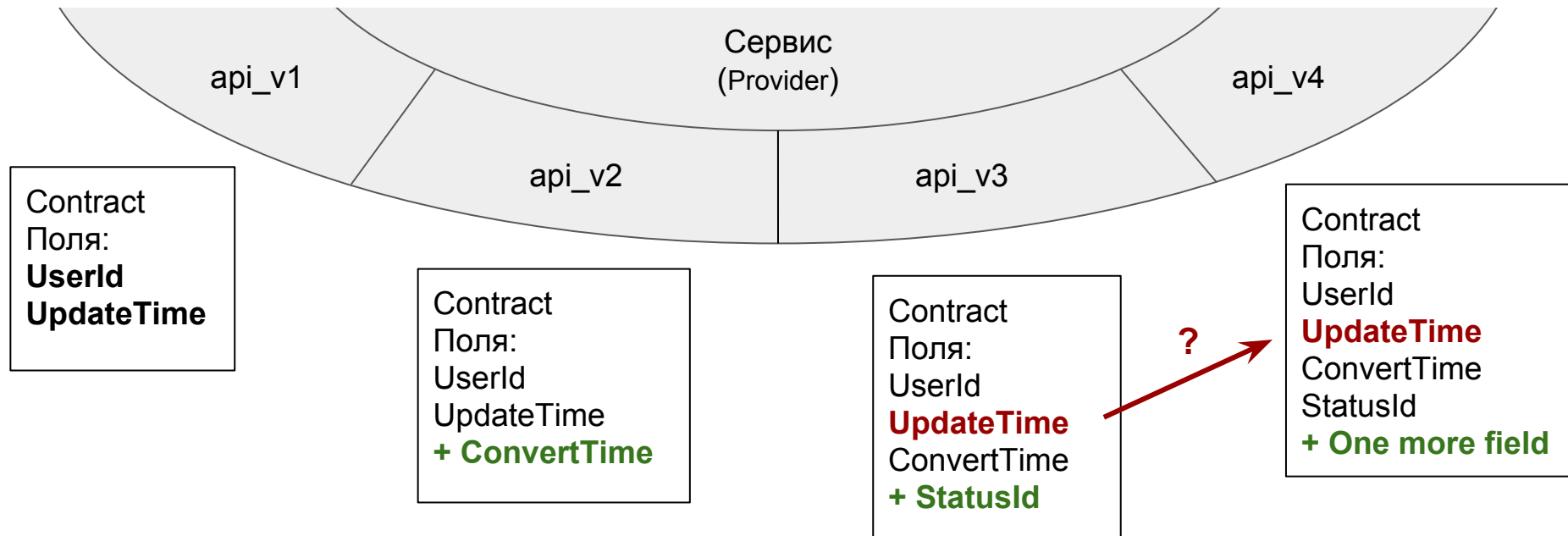
Микросервисы



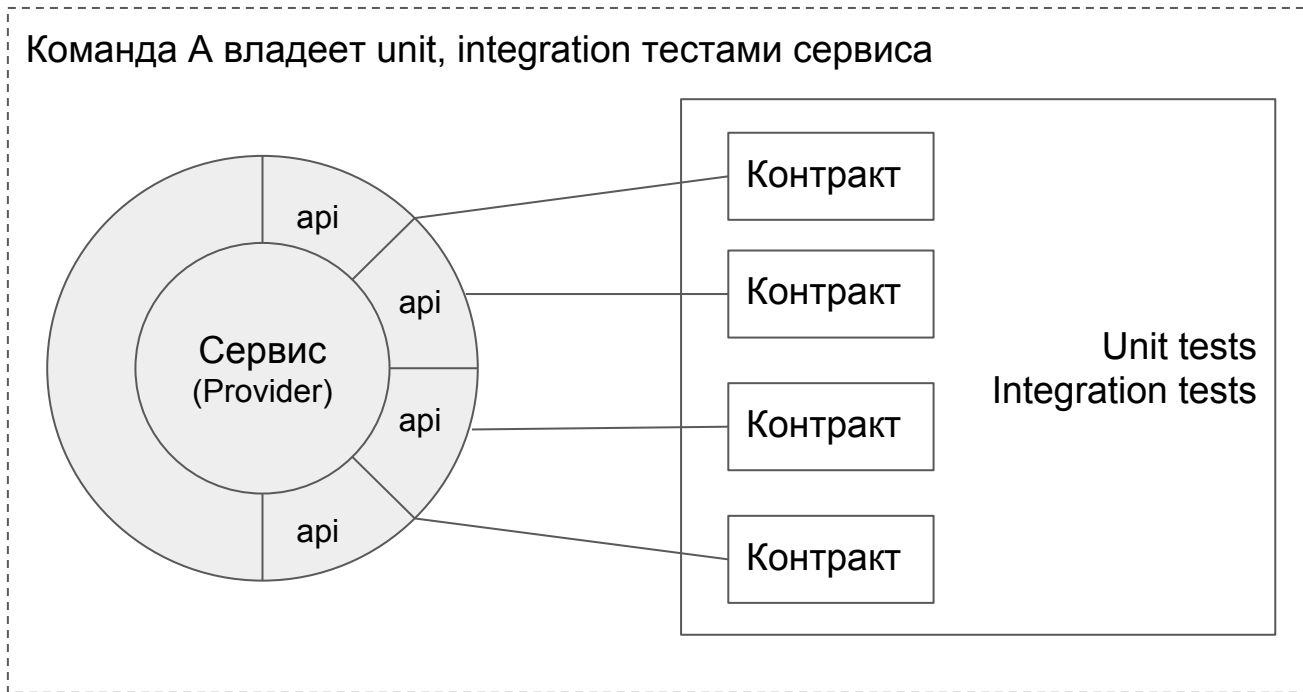
Микросервисы



Микросервисы



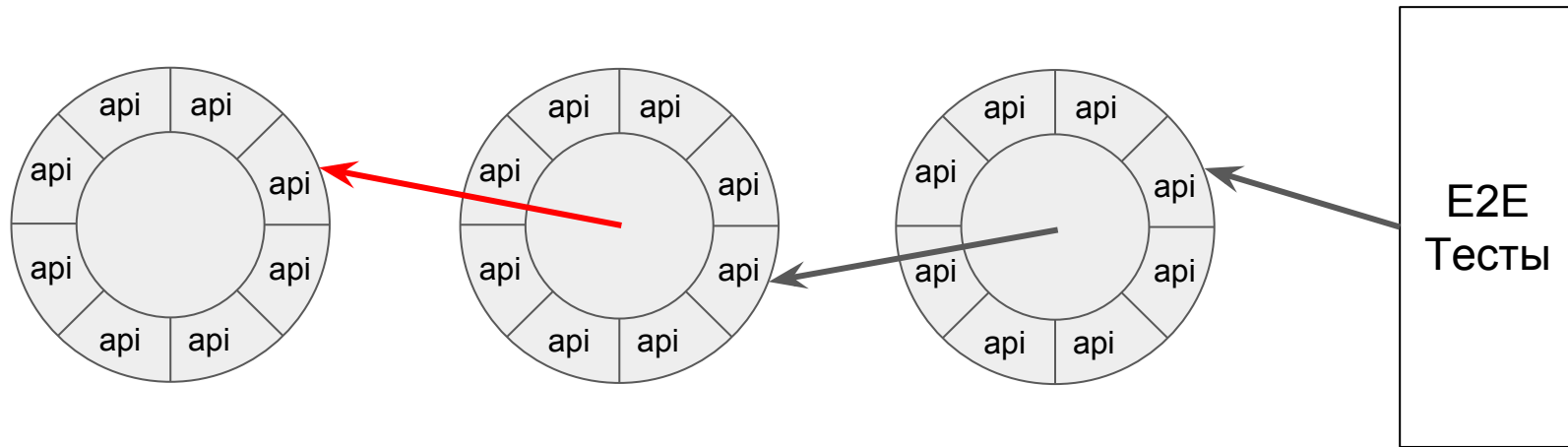
Unit и Integration тесты не работают



Unit и Integration тесты не работают

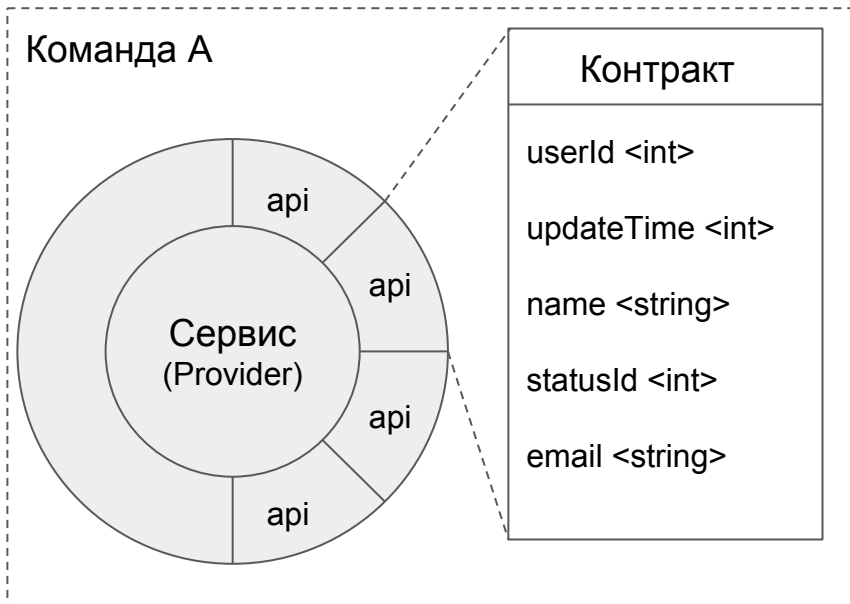
- Grafana
- Swagger
- Distributed tracing

Е2Е тесты выявят дефект, но это поздно и дорого



Consumer Driven Contracts (CDC) вид тестирования, при котором тесты на ожидания от API провайдера пишет потребитель.

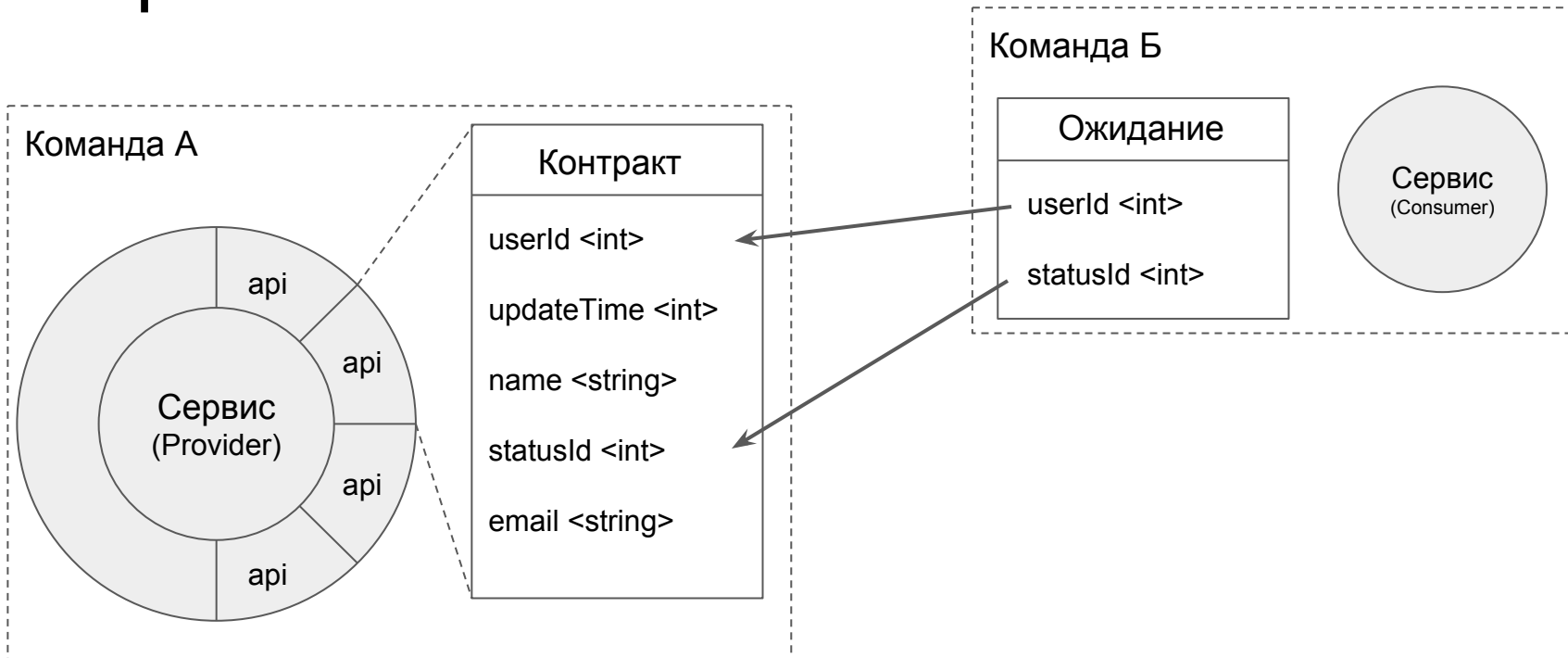
Что такое ожидания от контрактов сервисов



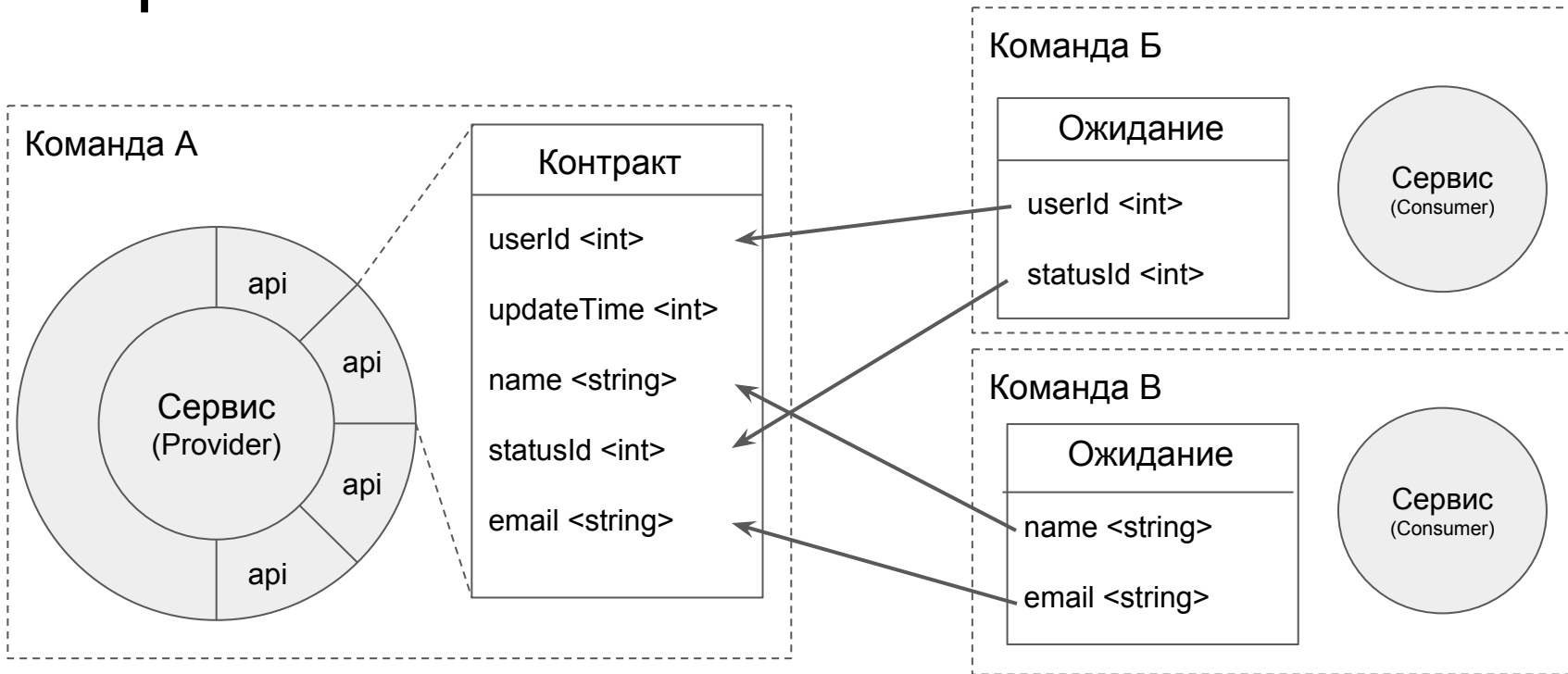
Контракт может быть задекларирован как:

- Json schema
- Xml schema
- ...

Что такое ожидания от контрактов сервисов



Что такое ожидания от контрактов сервисов



Наши требования к CDC framework

- Простота внедрения
- Простота поддержки ожиданий (в идеале, самогенерируемые ожидания)

Какие есть решения?

- Pact framework
- ...

Сервис Consumer

```
// Код микросервиса
function init() {
    ...
    client(host)
}

function action() {
    // Обращение в сервис
    entity = client->getById(id)
    ...

    // Обращение в сервис
    client->update(id, entity)
    ...
}
```

Сервис Consumer

```
// Код микросервиса
function init() {
    ...
    client(host)
}

function action() {
    // Обращение в сервис
    entity = client->getById(id)
    ...

    // Обращение в сервис
    client->update(id, entity)
    ...
}
```

Сервис Consumer. Pact contract

```
// Код микросервиса
function init() {
  ...
  client(host)
}

function action() {
  // Обращение в сервис
  entity = client->getById(id)
  ...

  // Обращение в сервис
  client->update(id, entity)
  ...
}
```

```
// Используя DSL описываем ожидание
// от сервиса provider

Test getById {

  expectedResponse
    ->setStatus(200)
    ->setBody([
      'id' => matcher->like(12)
      'name' => matcher->like('.')
      ...
    ])
}
```

Сервис Consumer. Контракт на основе unit tests

```
// Код микросервиса
function init() {
    ...
    client(host)
}

function action() {
    // Обращение в сервис
    entity = client->getById(id)
    ...

    // Обращение в сервис
    client->update(id, entity)
    ...
}
```

```
// Код unit тестов
function init() {
    ...
    client(env(HOST))
}
// annotation service-provider
TestGetById {
    // Обращение в сервис
    assert(client->getById(id))
}
```


Сервис Consumer. Контракт на основе unit tests

```
// Код микросервиса
function init() {
    ...
    client(host)
}

function action() {
    // Обращение в сервис
    entity = client->getById(id)
    ...

    // Обращение в сервис
    client->update(id, entity)
    ...
}
```

```
// Код unit тестов
function init() {
    ...
    client(env(HOST))
}

// annotation service-provider
TestGetById {
    // Обращение в сервис
    assert(client->getById(id))
}

// annotation service-provider
TestUpdate {
    // Обращение в сервис
    client->update(new entity)
}
```

Сервис Consumer. Контракт на основе unit tests

```
// Код микросервиса
function init() {
    ...
    client(host)
}

function action() {
    // Обращение в сервис
    entity = client->getById(id)
    ...

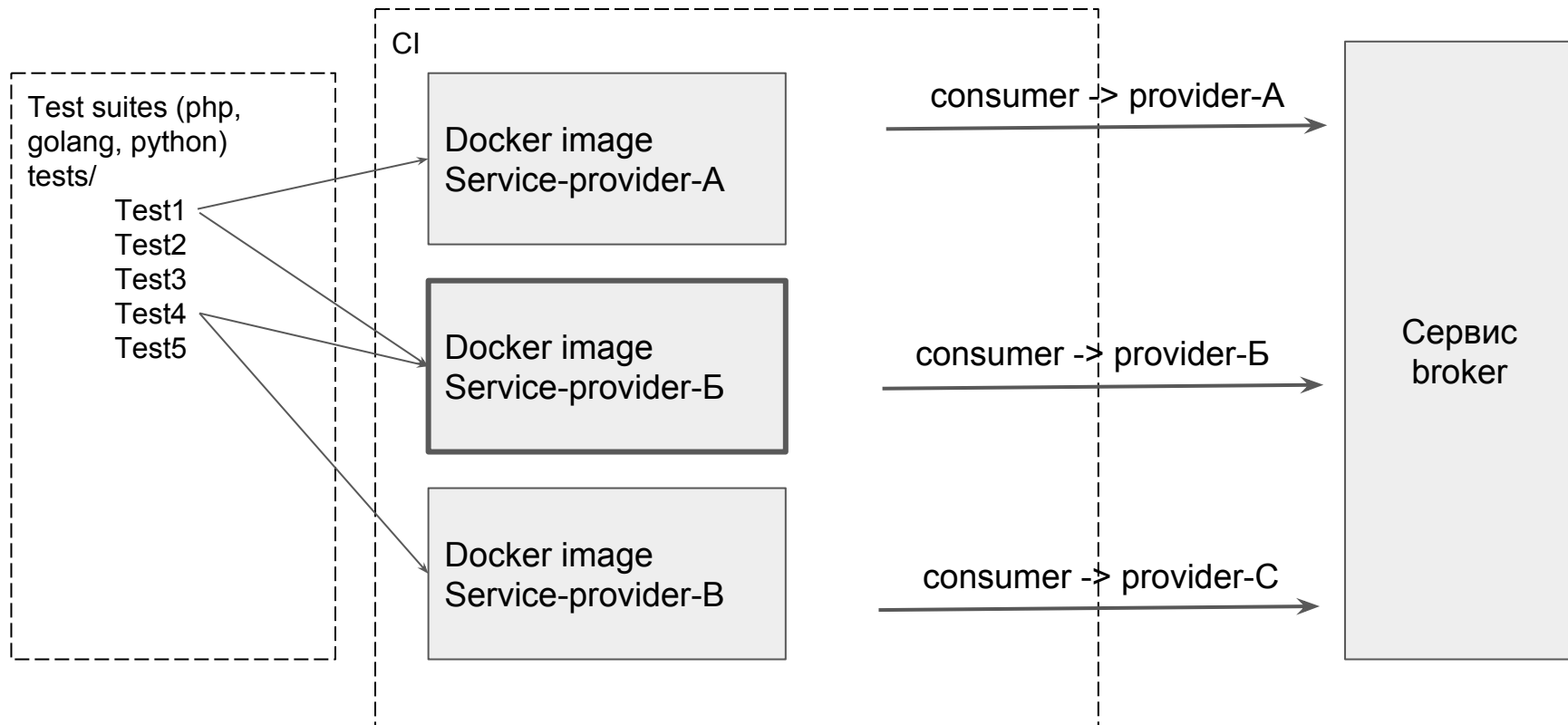
    // Обращение в сервис
    client->update(id, entity)
    ...
}
```

```
// Код unit тестов
function init() {
    ...
    client(env(HOST))
}

// annotation service-provider
TestGetById {
    // Обращение в сервис
    assert(client->getById(id))
}

// annotation service-provider
TestUpdate {
    // Обращение в сервис
    client->update(new entity)
}
```

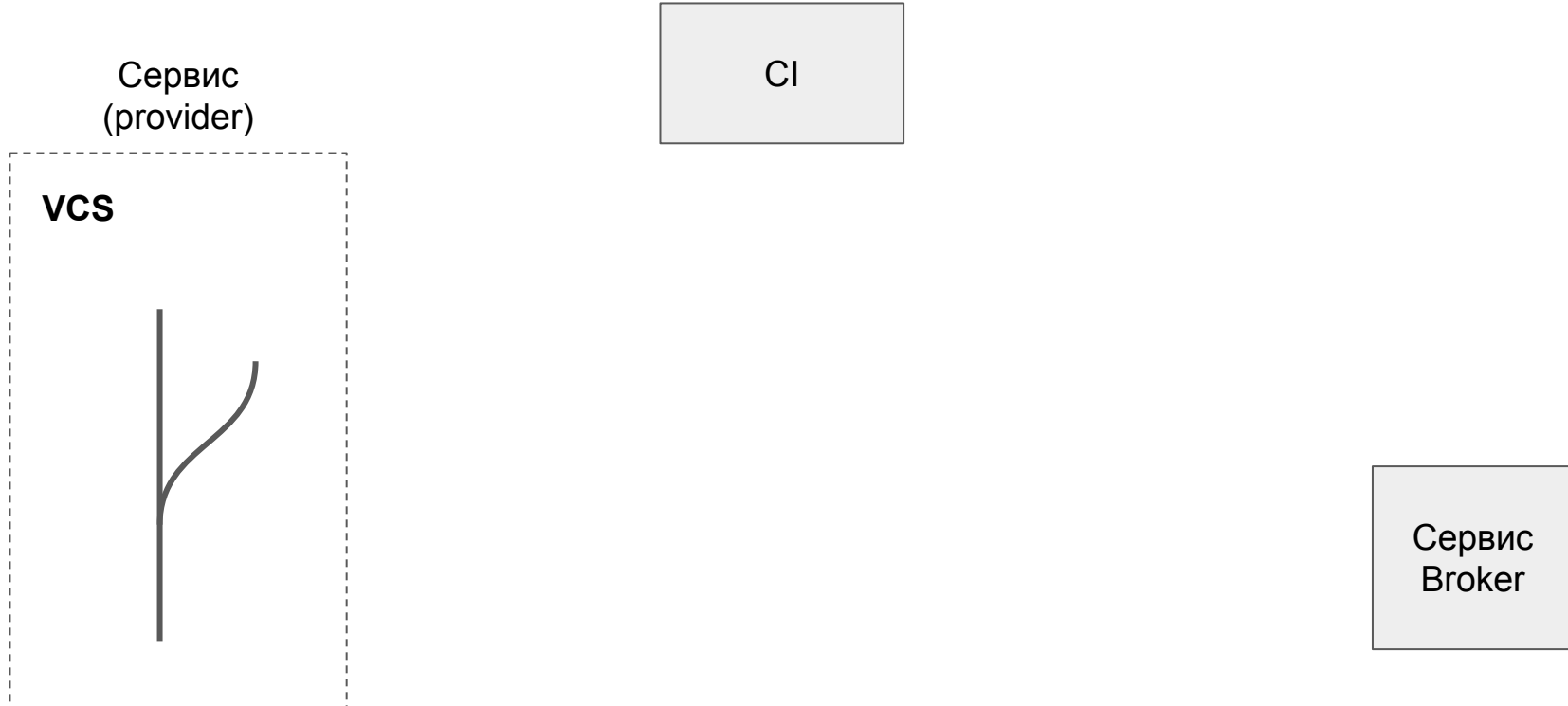
Сервис Consumer. Сервис broker



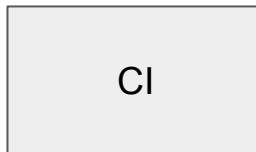
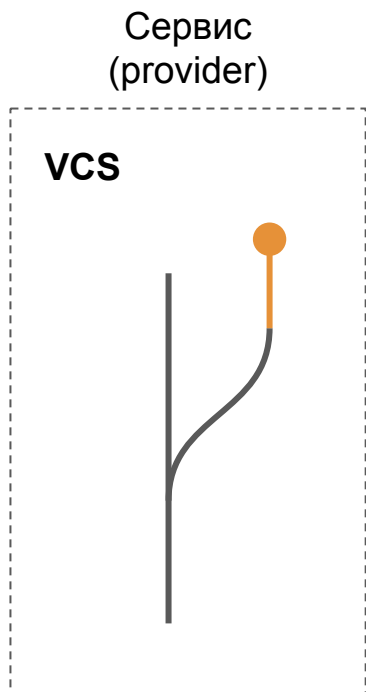
CI для CDC тестирования имеет 2 процесса

- Запуск CDC тестов
- Актуализация CDC тестов

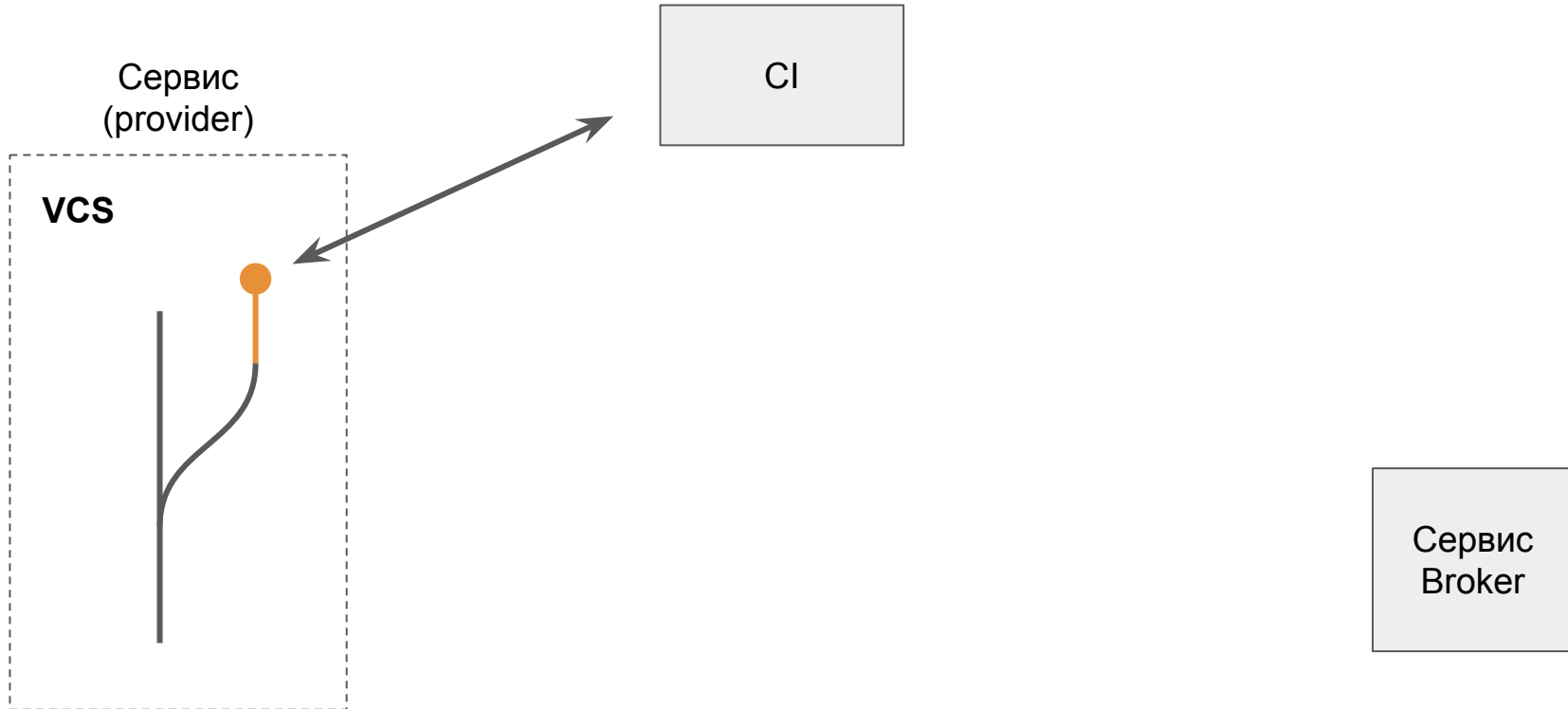
Запуск CDC тестов



Запуск CDC тестов



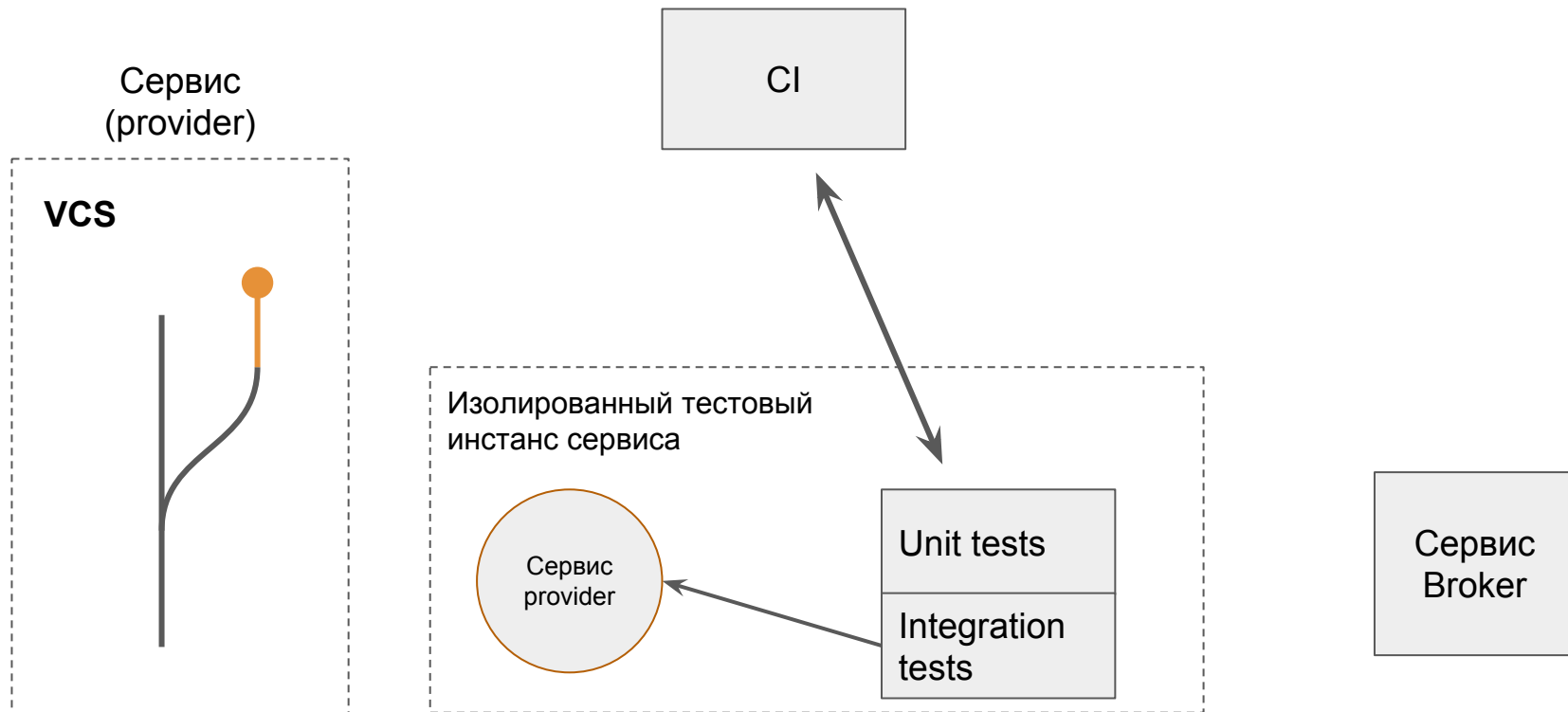
Запуск CDC тестов



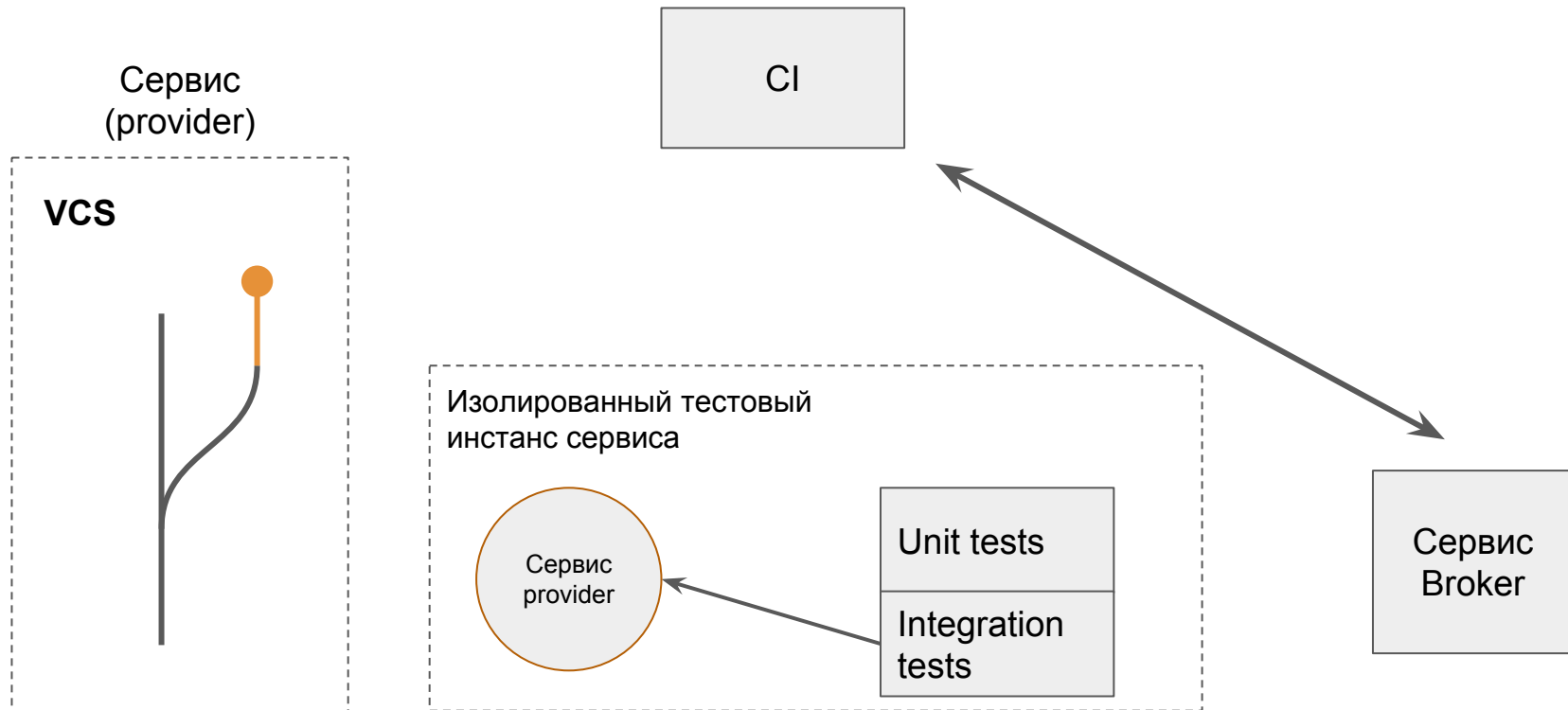
Запуск CDC тестов



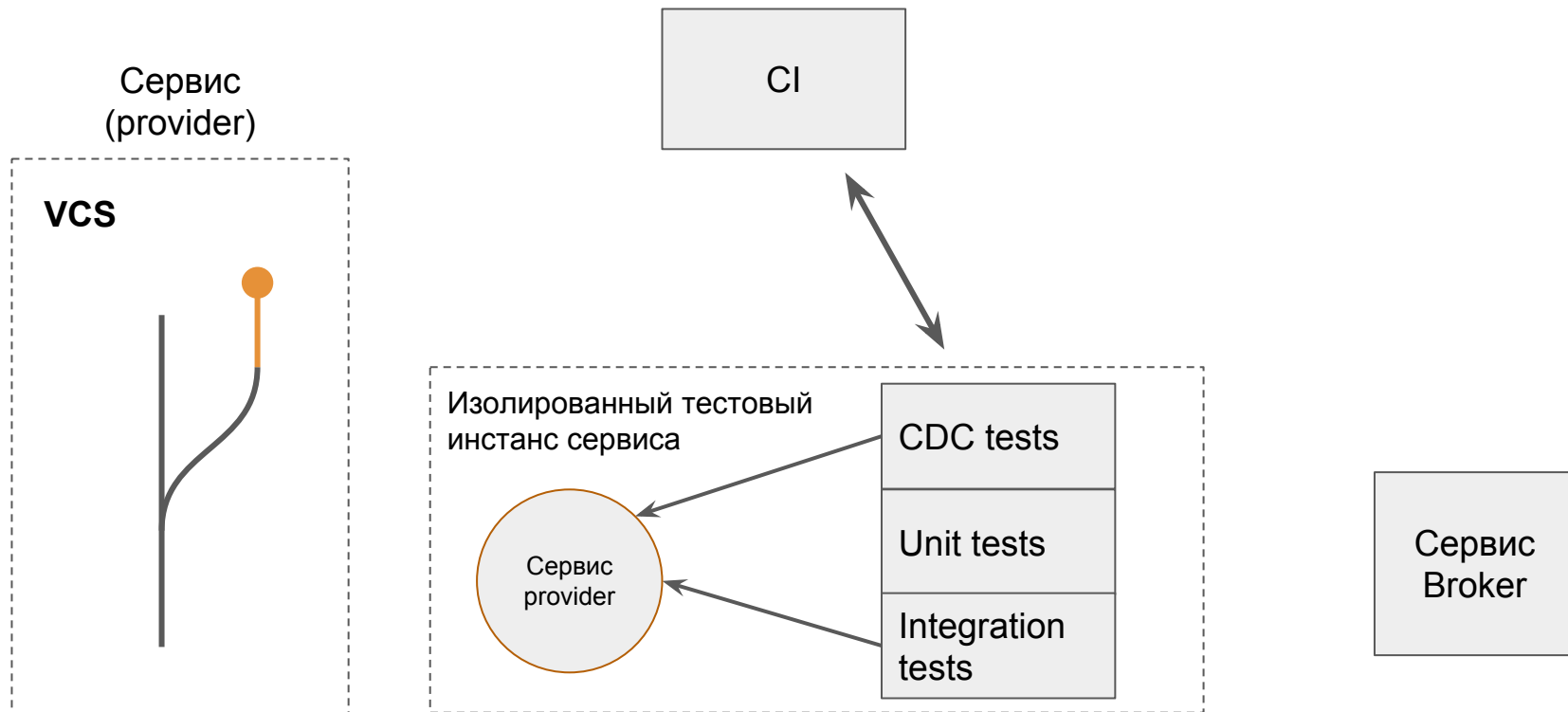
Запуск CDC тестов



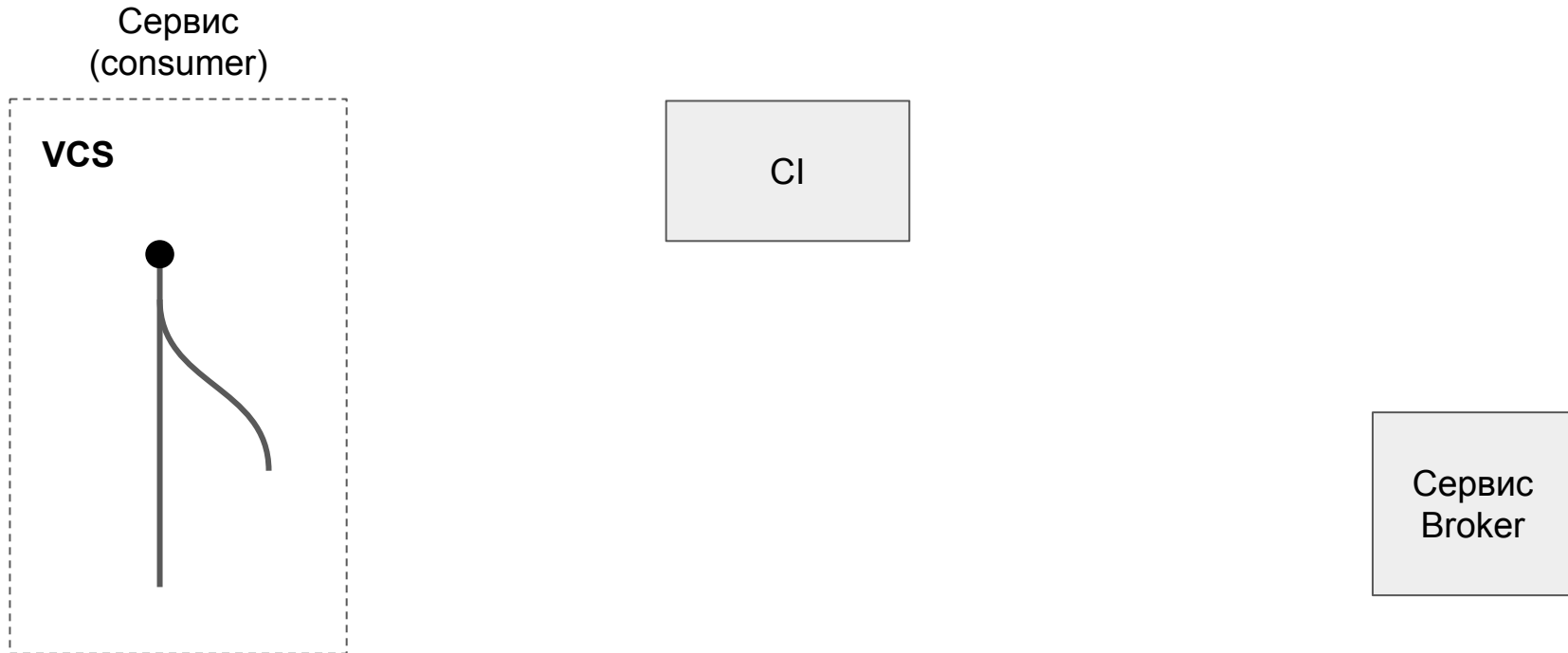
Запуск CDC тестов



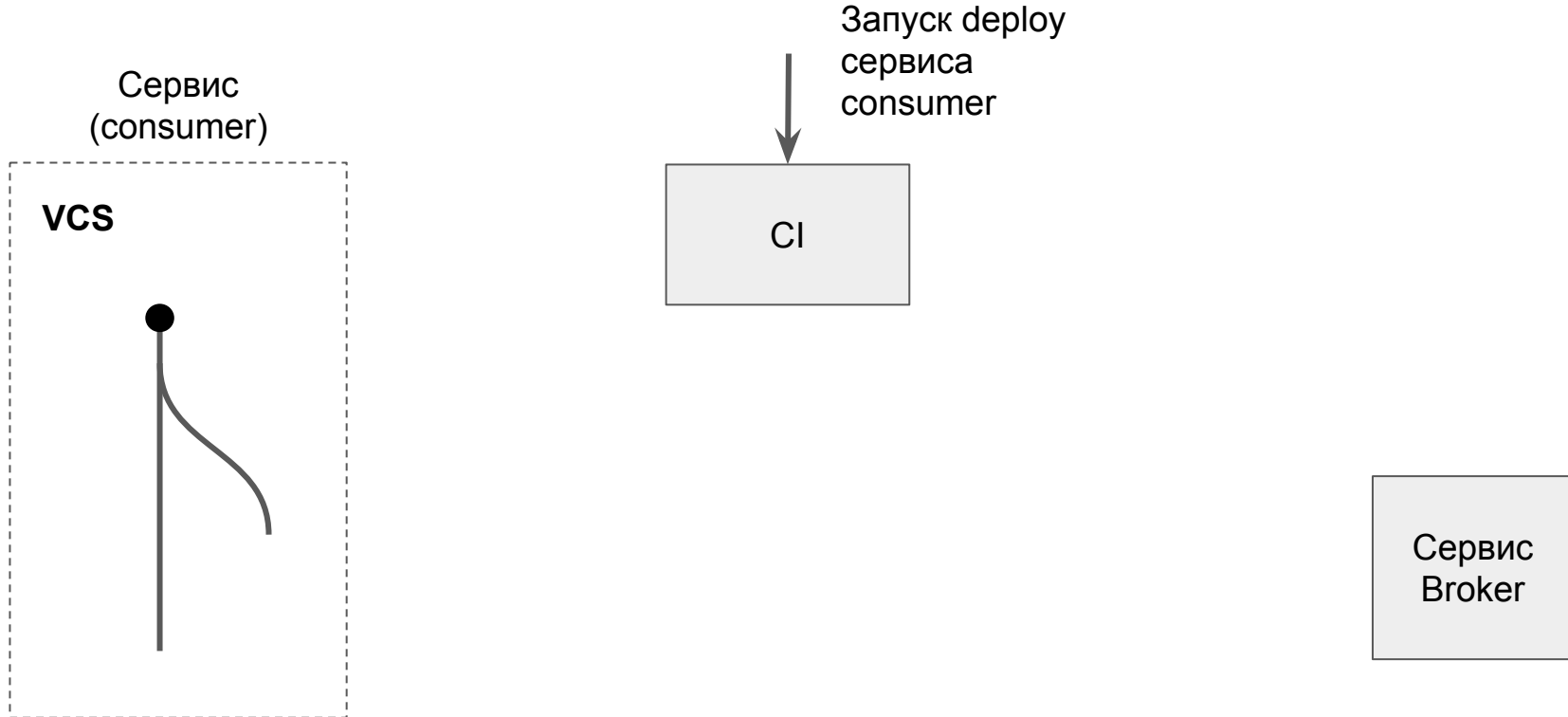
Запуск CDC тестов



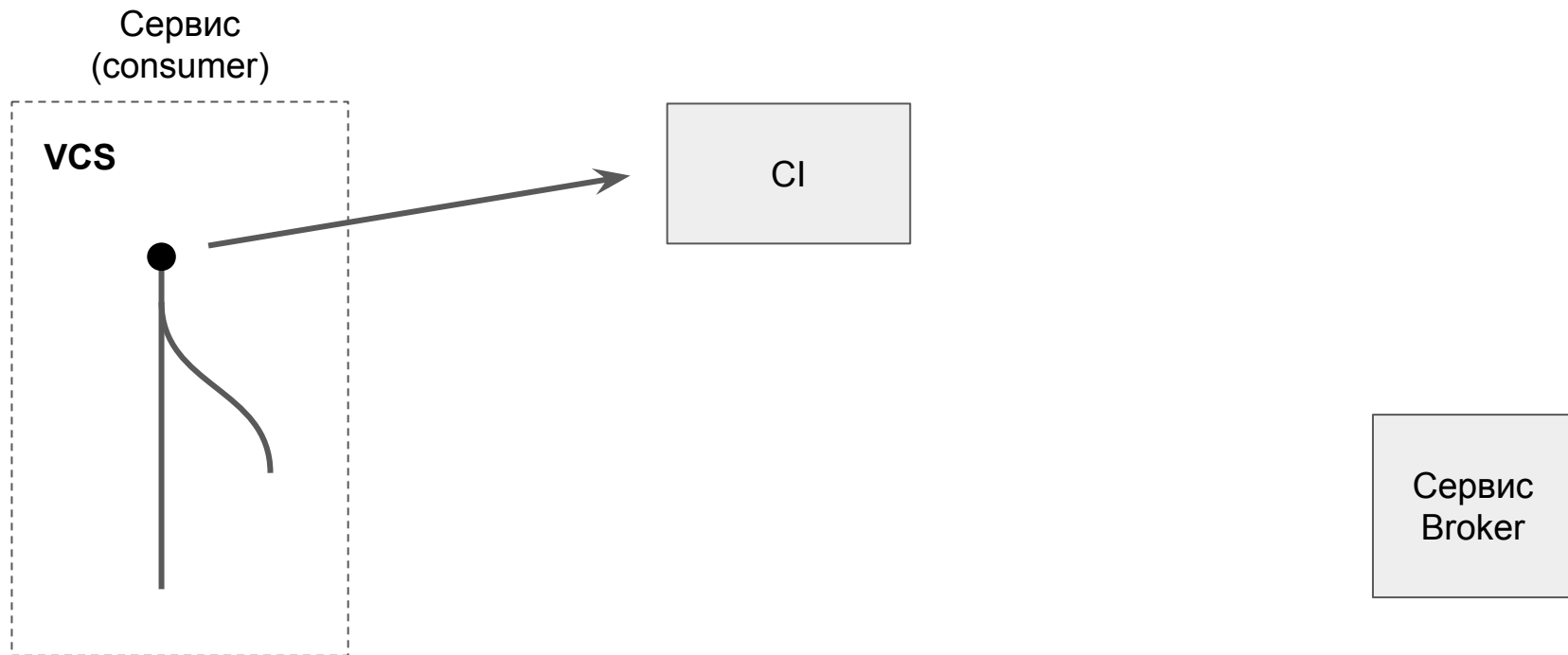
Актуализация CDC тестов



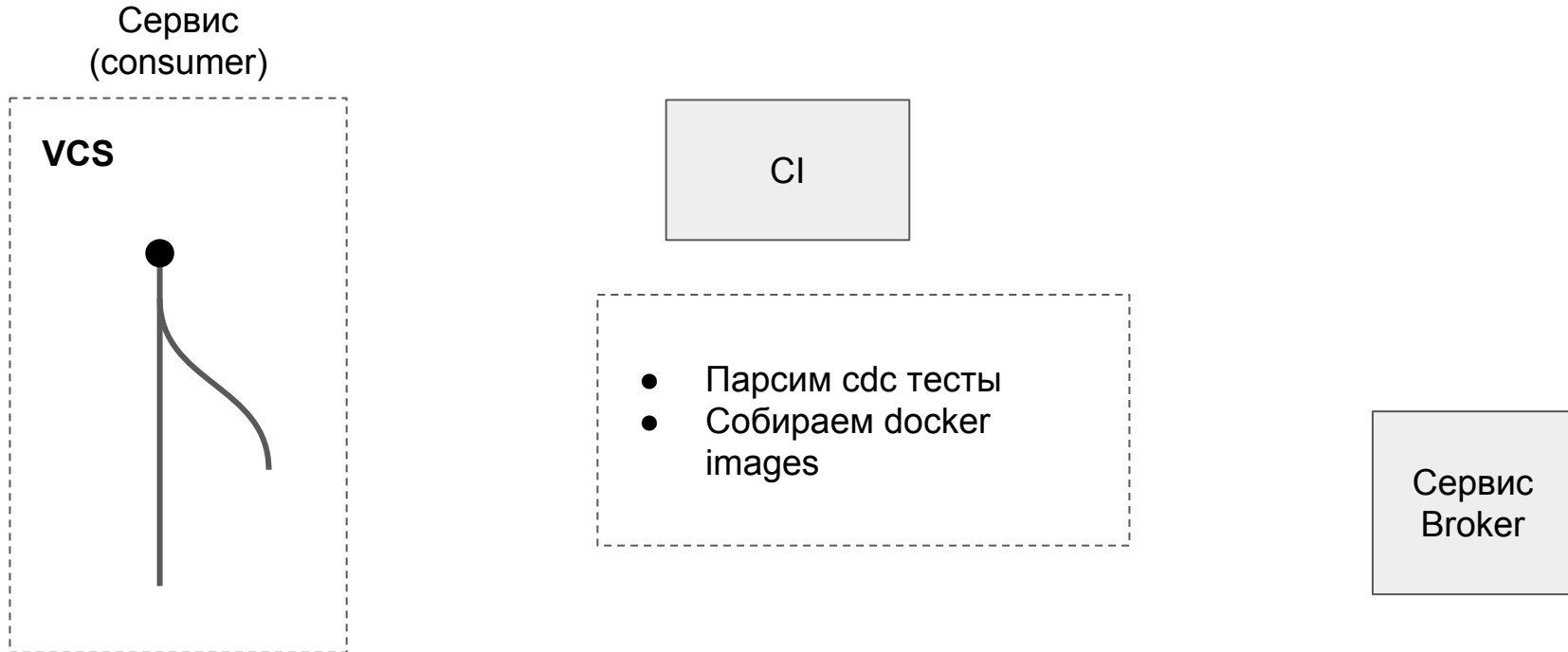
Актуализация CDC тестов



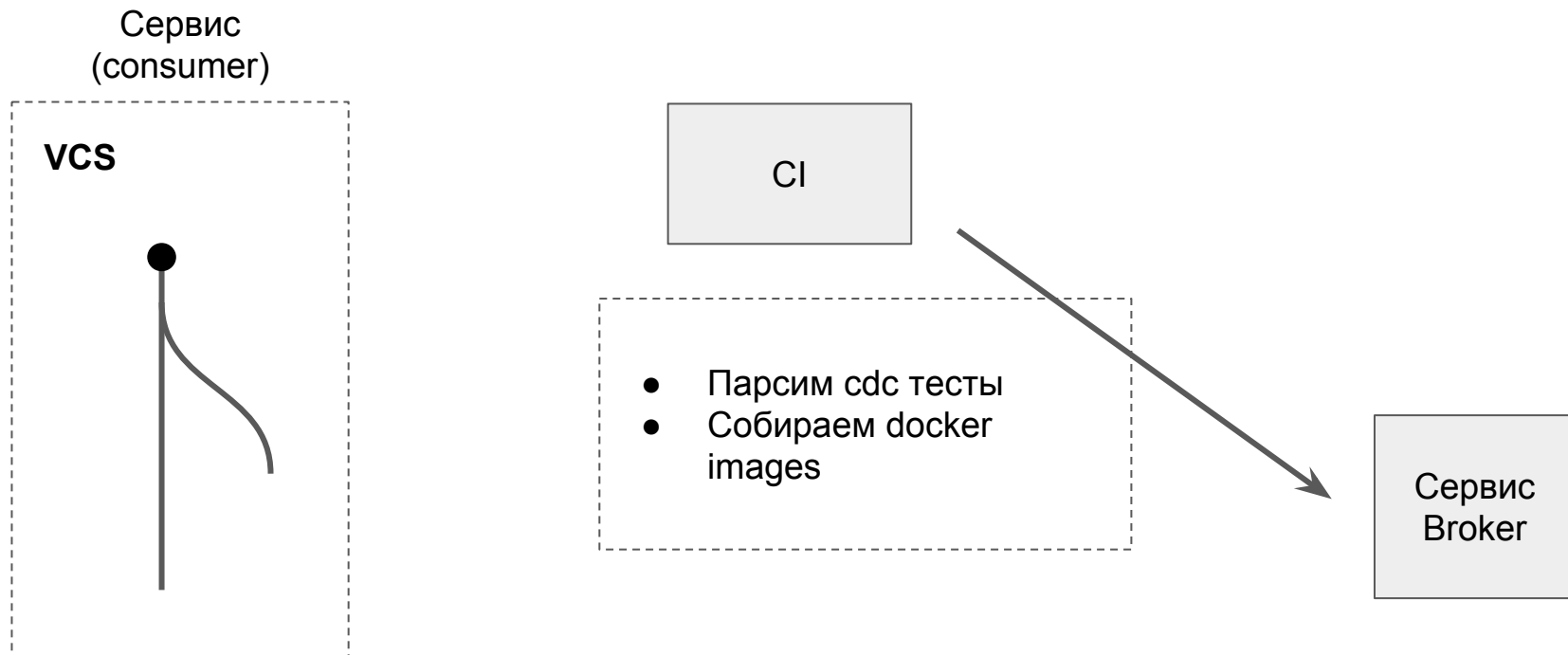
Актуализация CDC тестов



Актуализация CDC тестов



Актуализация CDC тестов



Best practices

- Тесты должны быть изолированными
- Не использовать как функциональные тесты
- Проверять **только** контракты (типы)

Недостатки

- Не реализован mock сервер [WIP]
- Нет вариативности для request/response matching

Спасибо за внимание!