



Мы решили все проблемы

мультирепы

Хотите так же?

Михаил Харитончик
iOS-MobilePlatform

Вводные

5 000 000

Строк кода

5 МИН

Скорость сборки

Мобильная разработка в цифрах:

200

iOS разработов в одном только СБОЛ

200

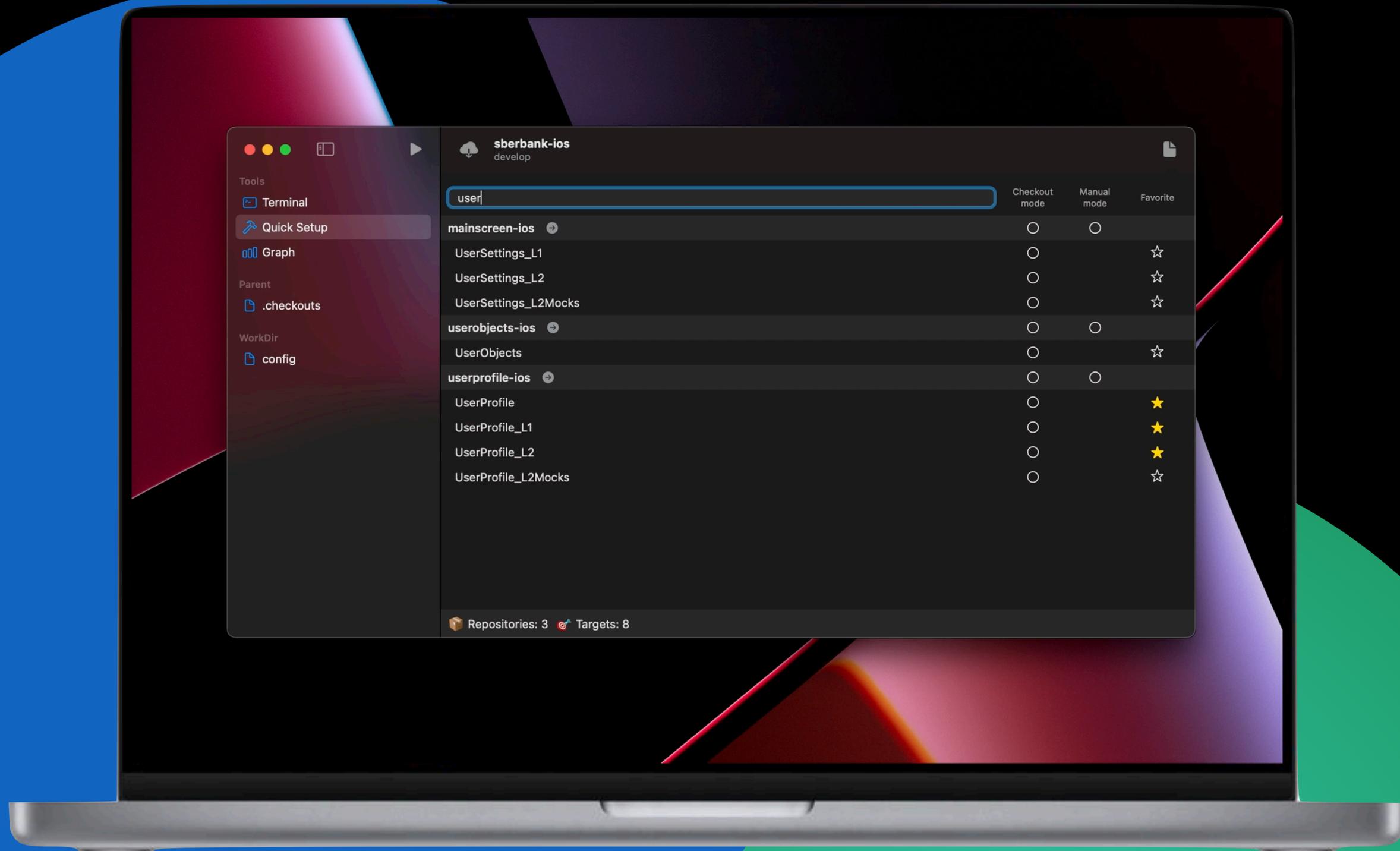
Модулей

50к

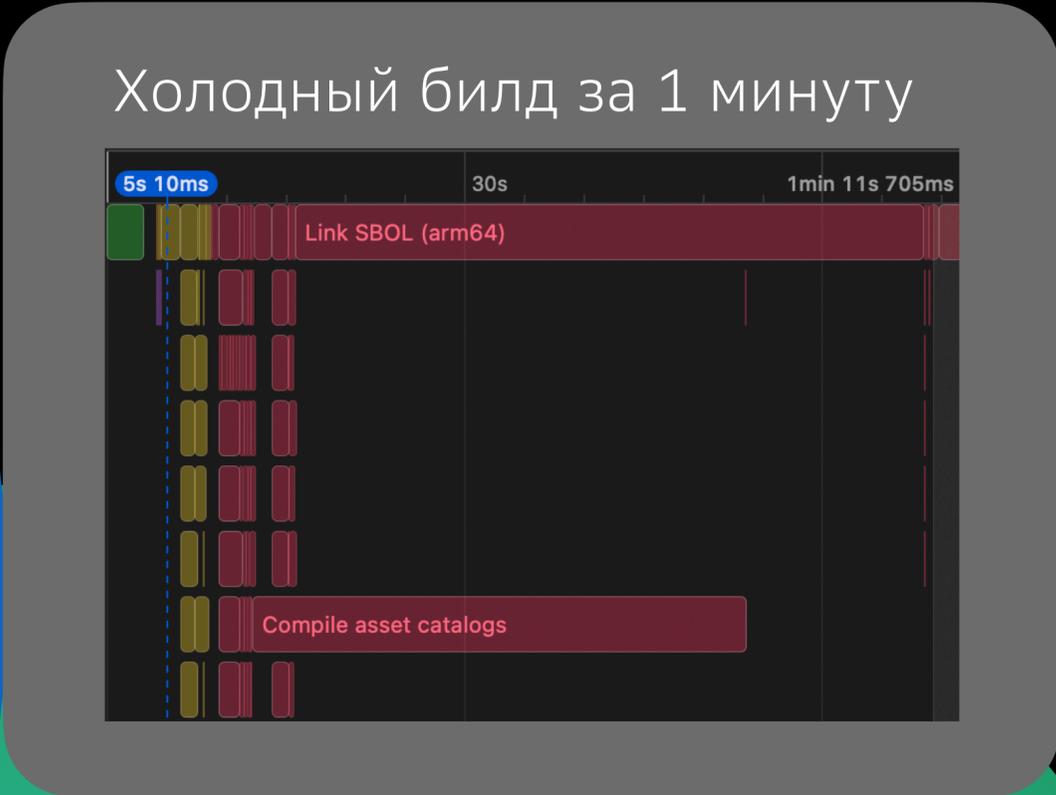
Строк кода в своей системе сборки

12 лет

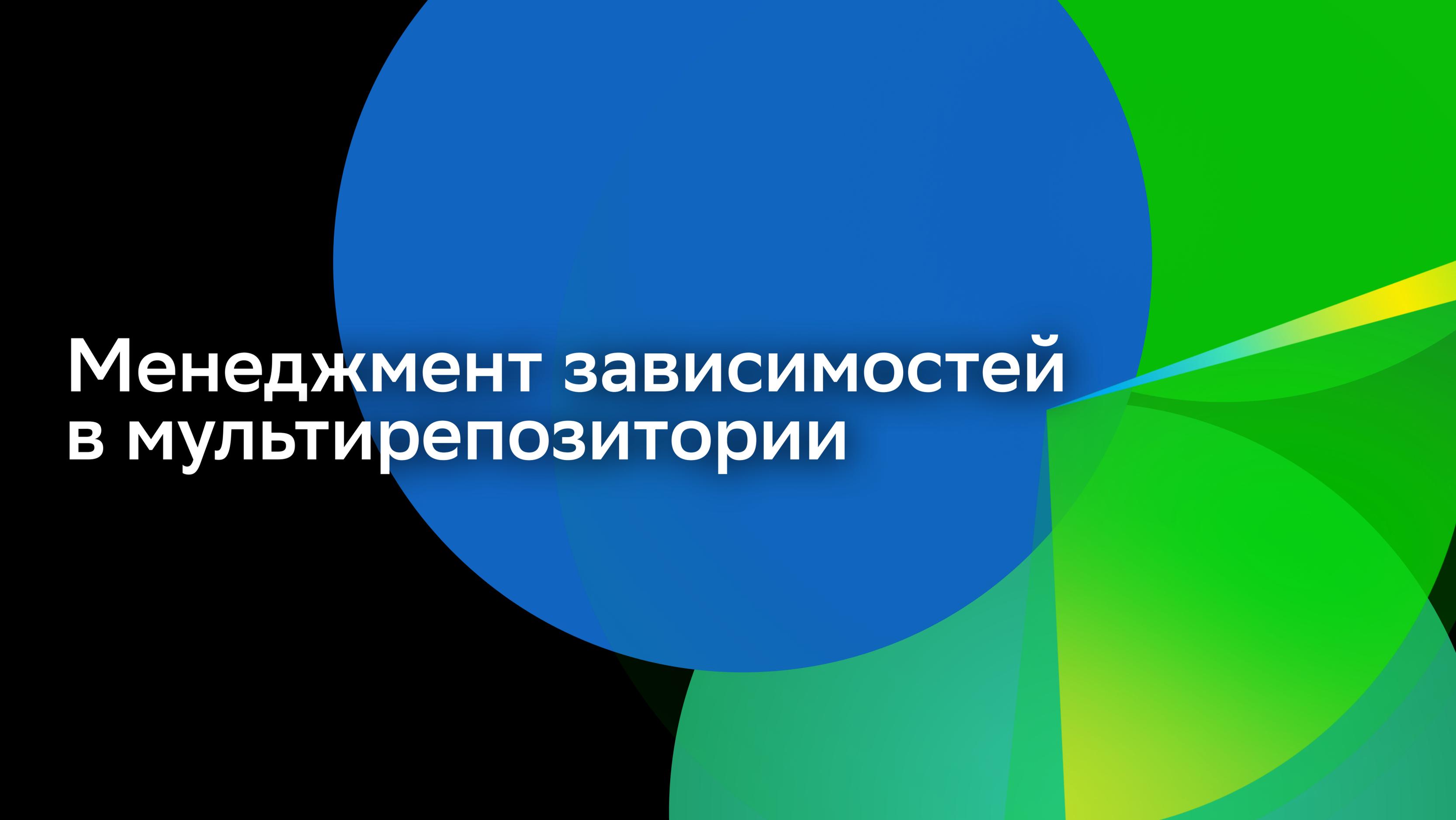
Возраст кодовой базы iOS



Сквозь кротовую нору: что скрывается за быстрой сборкой проекта» / Владимир Озеров



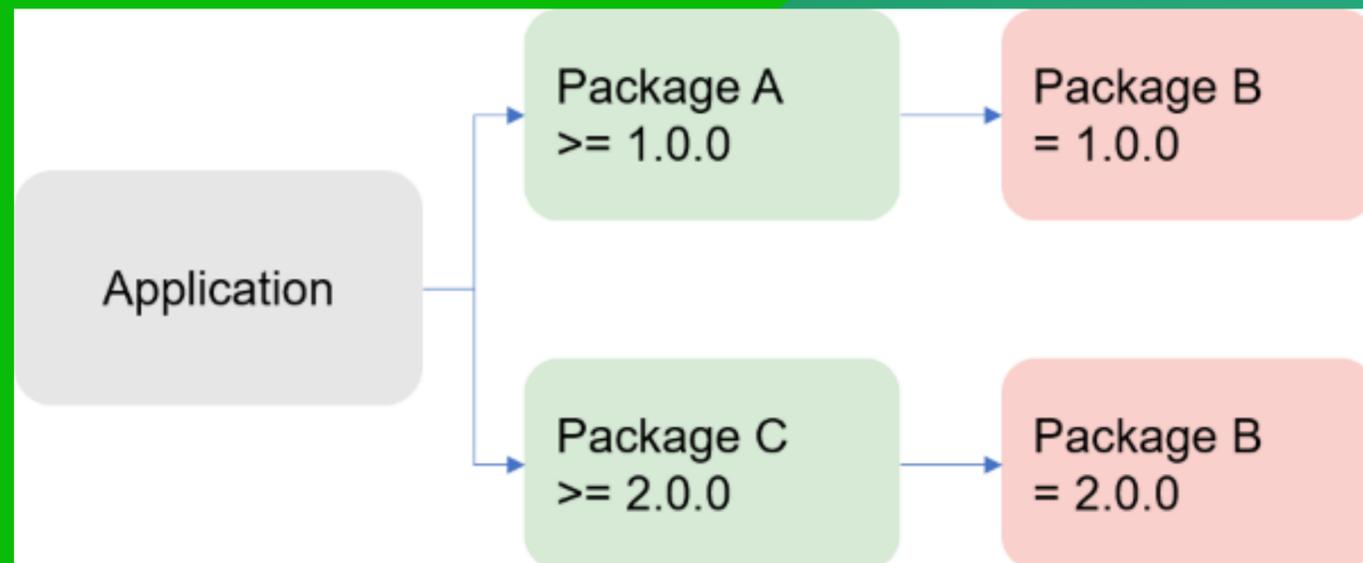
Сквозь кротовую нору: что скрывается за быстрой сборкой проекта» / Владимир Озеров



Менеджмент зависимостей в мультирепозитории

Мультирепозиторий...

Несовместимость транзитивных зависимостей



Мультирепозиторий...

Несовместимость транзитивных зависимостей

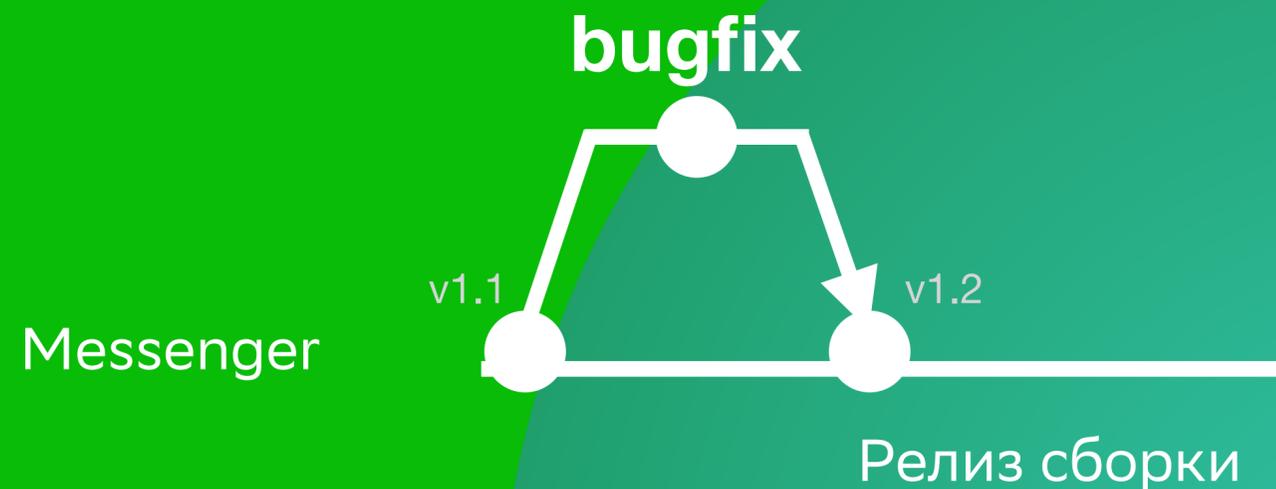
Нужно делать несколько ПРов:
влить фичу, задеплоить, поднять версию



Мультирепозиторий...

Несовместимость транзитивных зависимостей

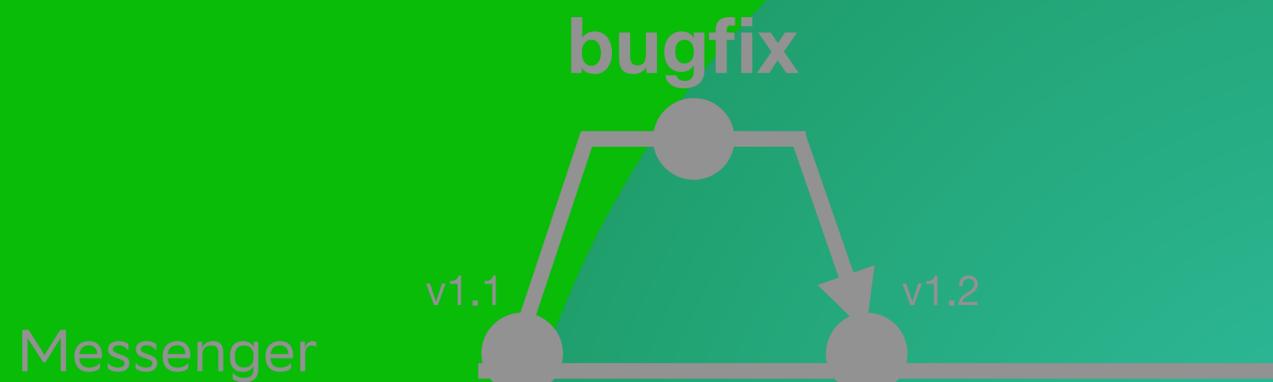
Нужно делать несколько ПРов:
влить фичу, задеплоить, поднять версию



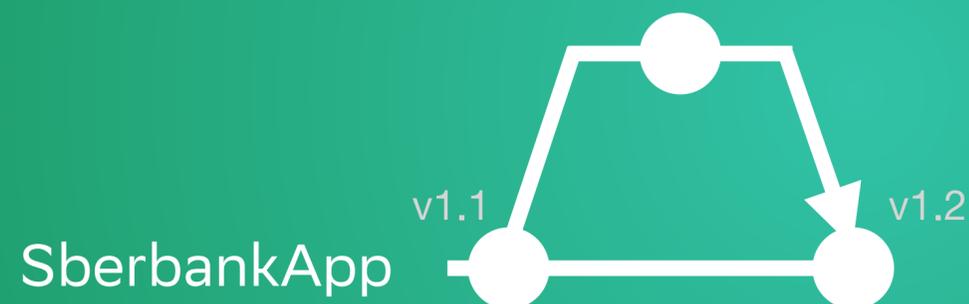
Мультирепозиторий...

Несовместимость транзитивных зависимостей

Нужно делать несколько ПРов:
влить фичу, задеплоить, поднять версию



bump version



Мультирепозиторий...

Несовместимость транзитивных зависимостей

Нужно делать несколько ПРов:
вливать фичу, задеплоить, поднять версию

Затруднен дебаггинг



```
1 UIKitCore`-[UILabel setText:]:
2 -> 0x105f0a028 <+0>: pushq %rbp
3 0x105f0a029 <+1>: movq %rsp, %rbp
4 0x105f0a02c <+4>: movq 0xf904ed(%rip), %rsi ; "_setText:"
5 0x105f0a033 <+11>: popq %rbp
6 0x105f0a034 <+12>: jmpq *0xbdd46e(%rip) ; (void *)0x0000000102aeaa80: objc_msgSend
7
```

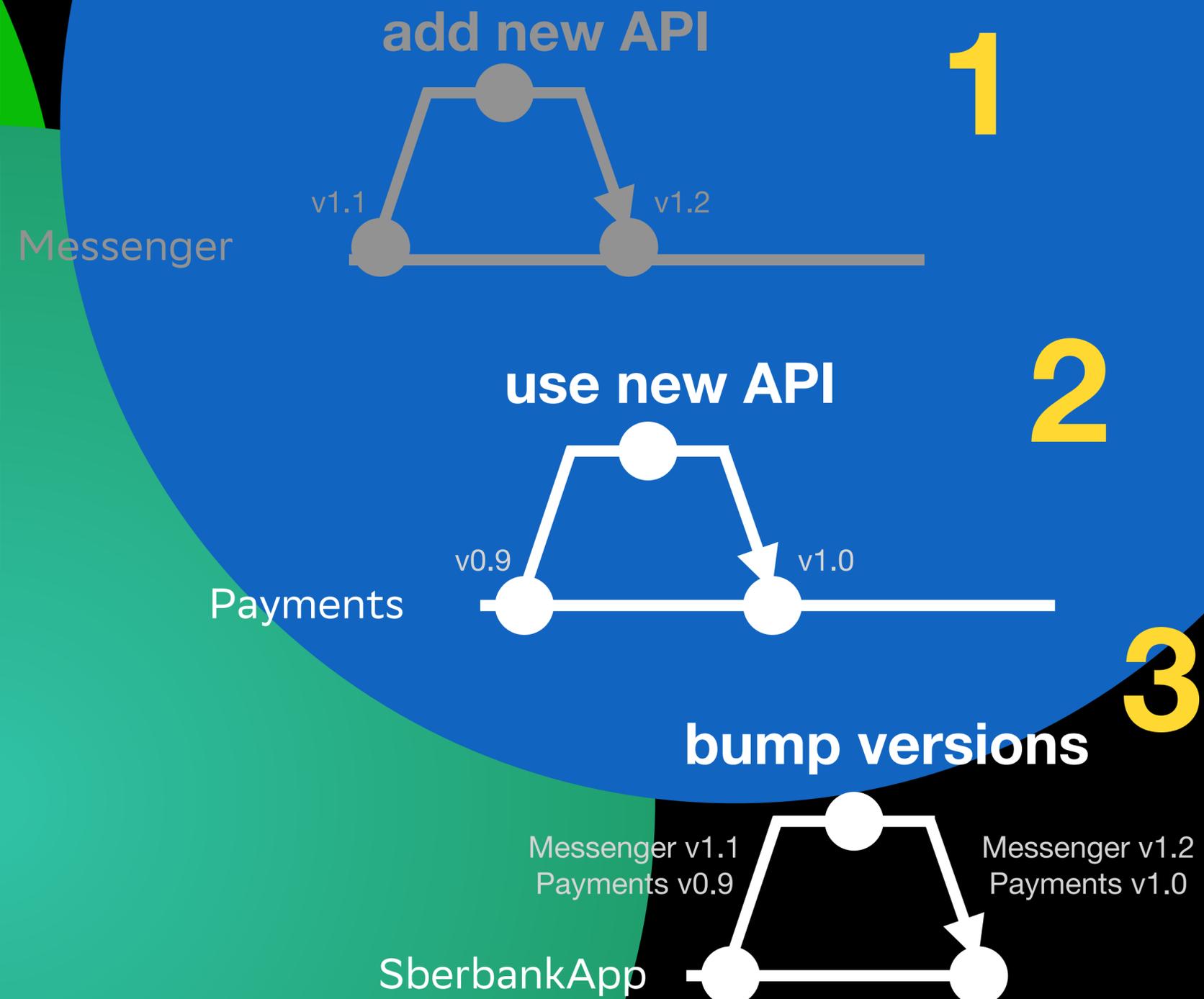
Мультирепозиторий...

Несовместимость транзитивных зависимостей

Нужно делать несколько ПРов:
вливать фичу, задеплоить, поднять версию

Затруднен дебаггинг

Сложно делать интеграционные изменения



Мультирепозиторий...

Несовместимость транзитивных зависимостей

Нужно делать несколько ПРов:
вливать фичу, задеплоить, поднять версию

Затруднен дебаггинг

Сложно делать интеграционные изменения

Нужно поддерживать совместимость API

```
@available(*, deprecated)
public func sendAnalytics(_ params: [String: String]? = nil) {
    let mParams: [String: String]
    if let params { mParams = params } else { mParams = [:] }
    print(mParams)
}

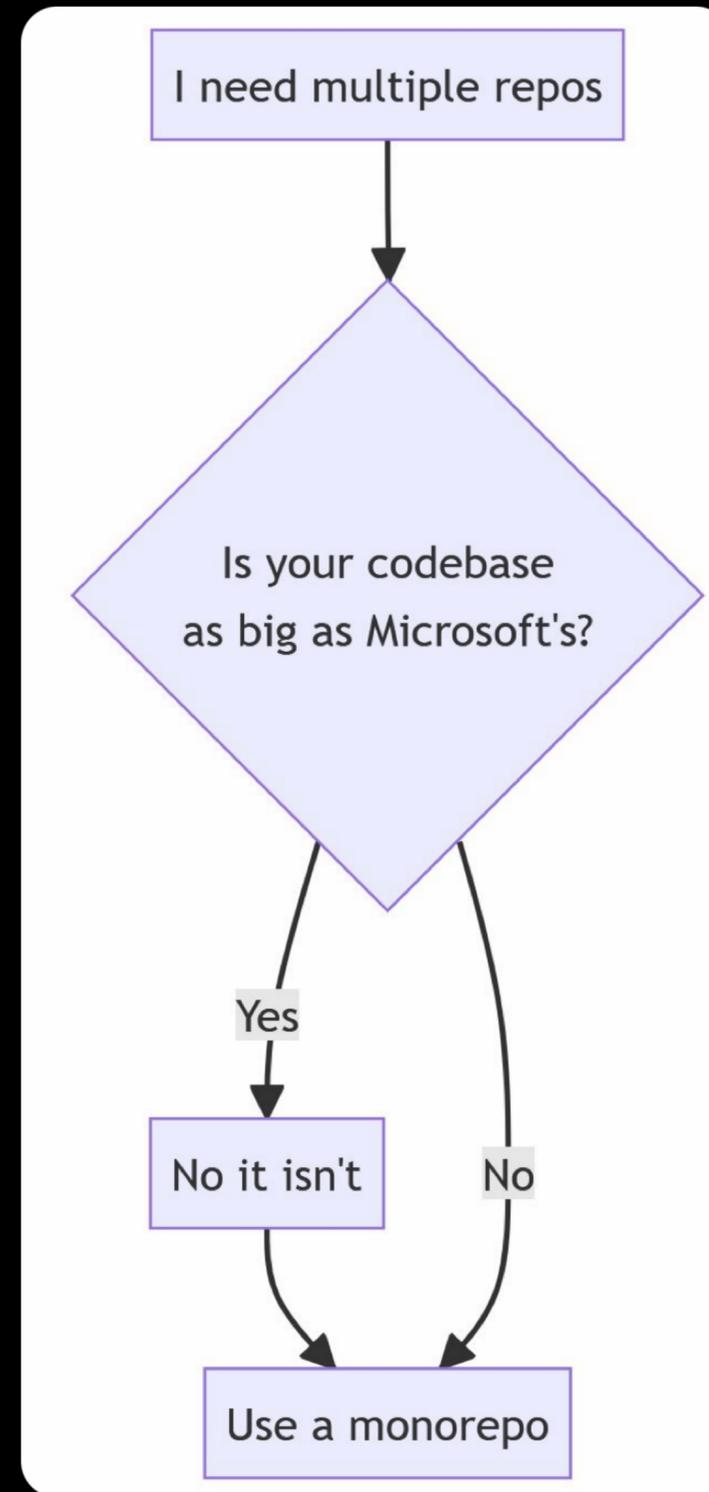
public func sendAnalytics(_ params: [String: String] = [:]) {
    let mParams: [String: String] = params
    print(mParams)
}
```

```
sendAnalytics(nil)
```



'sendAnalytics' is deprecated

I made an easy decision tree for anyone considering using git submodules or a monorepo.



Мультирепозиторий
==
Модульность



Монолит

Модульность

Мотивирует писать взаимосвязанный код

Пересечения в нейминге

В большом проекте сложно разобраться

Не требует дополнительных усилий

Проще переиспользовать код в рамках App

Требует поддержания консистентности

Циклические зависимости

Можно оптимизировать компиляцию

Мотивирует писать менее связанный код

Код можно переиспользовать в других App

Монолит



Модульность

Мотивирует писать взаимосвязанный код

Пересечения в нейминге

В большом проекте сложно разобраться

Не требует дополнительных усилий

Проще переиспользовать код в рамках App

Требует поддержания консистентности

Циклические зависимости

Можно оптимизировать компиляцию

Мотивирует писать менее связанный код

Код можно переиспользовать в других App

Монолит



Модульность

Мотивирует писать взаимосвязанный код

Пересечения в нейминге

В большом проекте сложно разобраться

Не требует дополнительных усилий

Проще переиспользовать код в рамках App

Требует поддержания консистентности

Циклические зависимости

Можно оптимизировать компиляцию

Мотивирует писать менее связанный код

Код можно переиспользовать в других App

Монолит



Модульность

Мотивирует писать взаимосвязанный код

Пересечения в нейминге

В большом проекте сложно разобраться

Не требует дополнительных усилий

Проще переиспользовать код в рамках App

Требует поддержания консистентности

Циклические зависимости

Можно оптимизировать компиляцию

Мотивирует писать менее связанный код

Код можно переиспользовать в других App

Монолит



Модульность

Мотивирует писать взаимосвязанный код

Пересечения в нейминге

В большом проекте сложно разобраться

Не требует дополнительных усилий

Проще переиспользовать код в рамках App

Требует поддержания консистентности

Циклические зависимости

Можно оптимизировать компиляцию

Мотивирует писать менее связанный код

Код можно переиспользовать в других App

Пора



Вы потратили много времени

Вы потратили много времени

Распилили код на модули



Messenger.swift



Payments.swift



Networking.swift



App



Messenger



Payments



Networking



App



Messenger.swift



Payments.swift

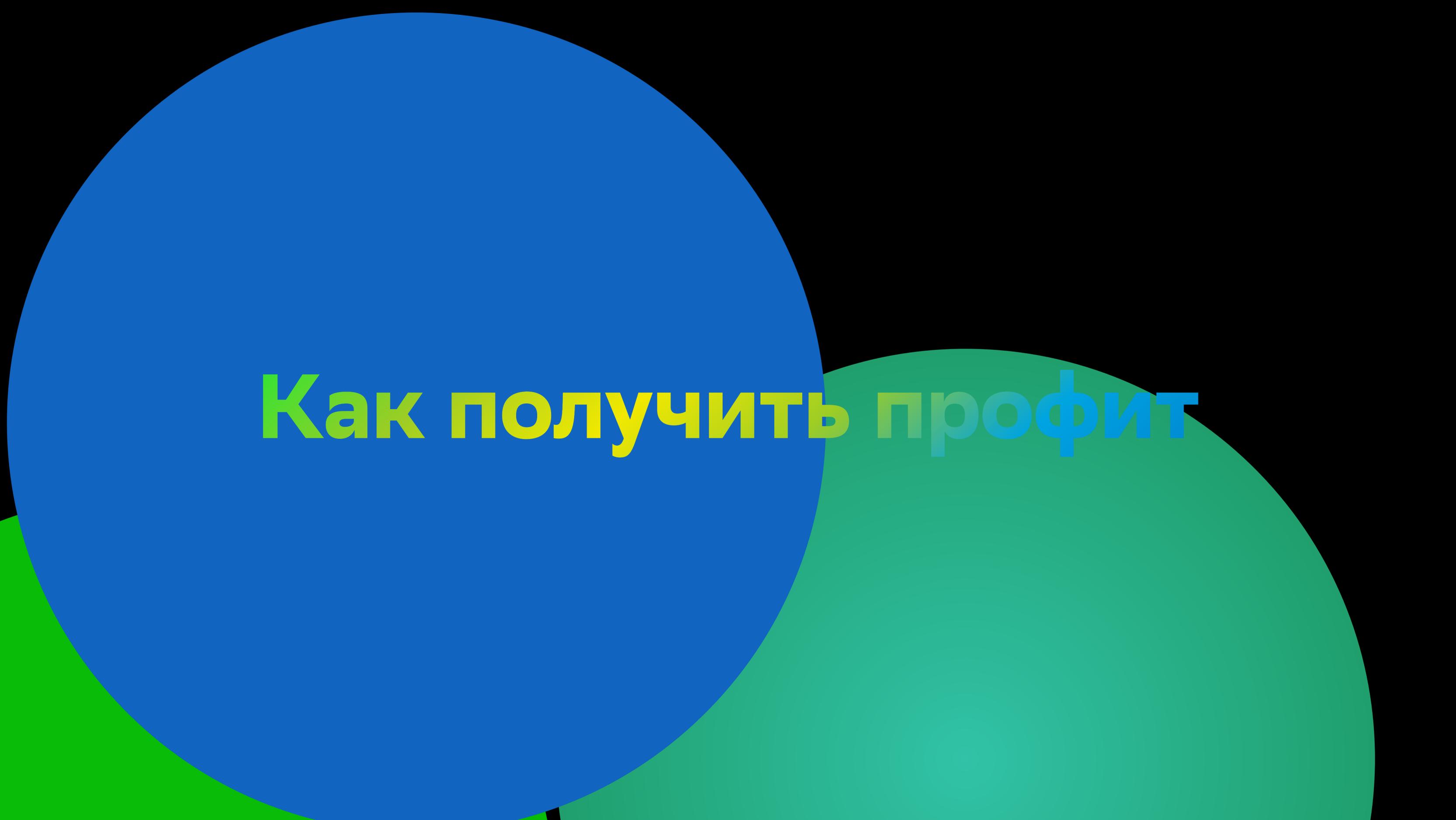


Networking.swift

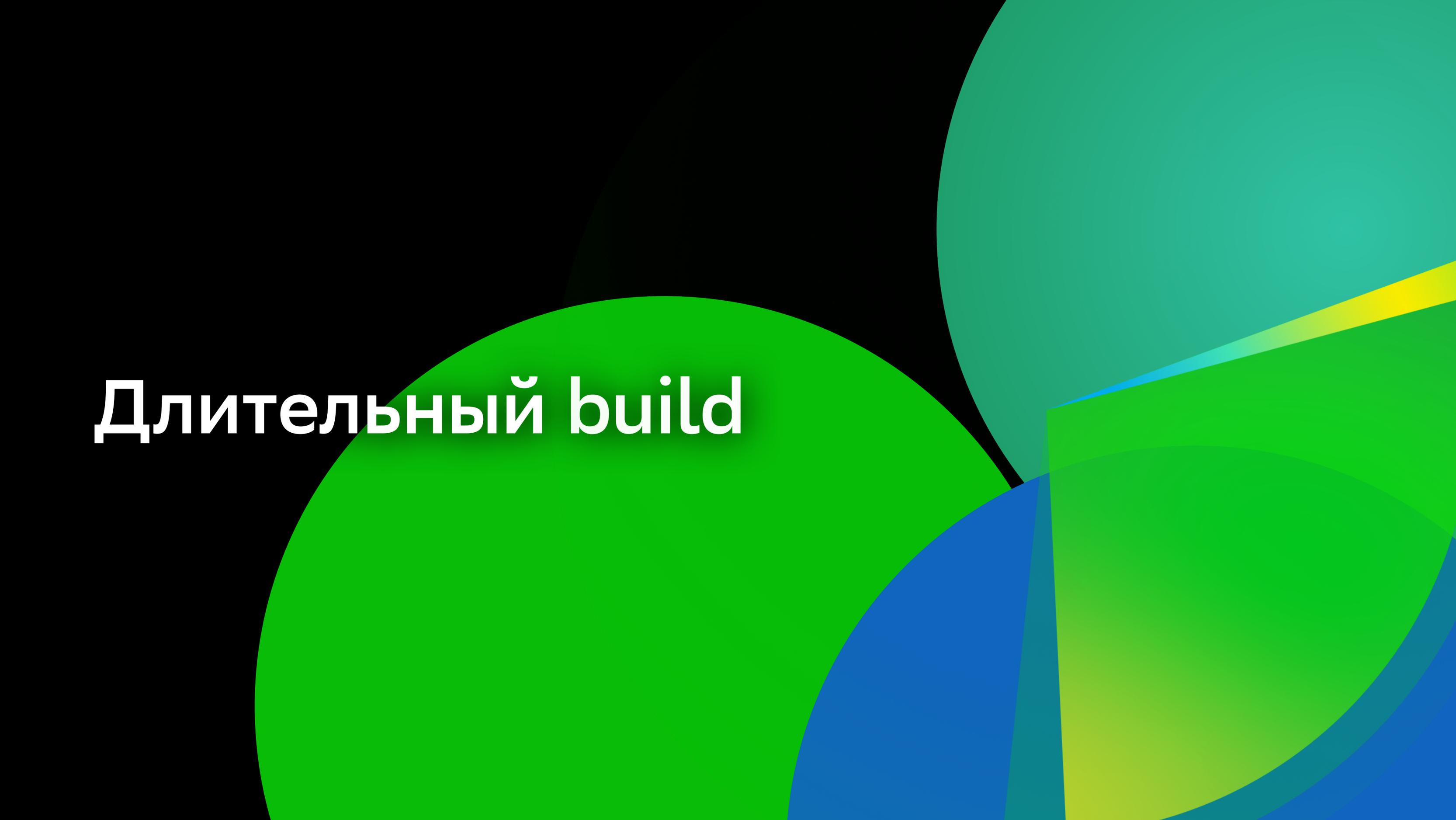
Вы потратили много времени

Распилили код на модули

Ничего не изменилось



Как получить профит



Длительный build

Ускорение Build

Bazel & Buck

Крупная миграция

Риск остаться без поддержки

Энтропия инструментов

Prebuild .framework-ов

Понятный для нас путь

Уже была модульность

“Нативный” способ

1
Выбор модулей
для работы

2

Build всего
что не нужно

Ускорение локальной сборки

3

Замена кода
на артефакты



Messenger



Payments



Networking



App



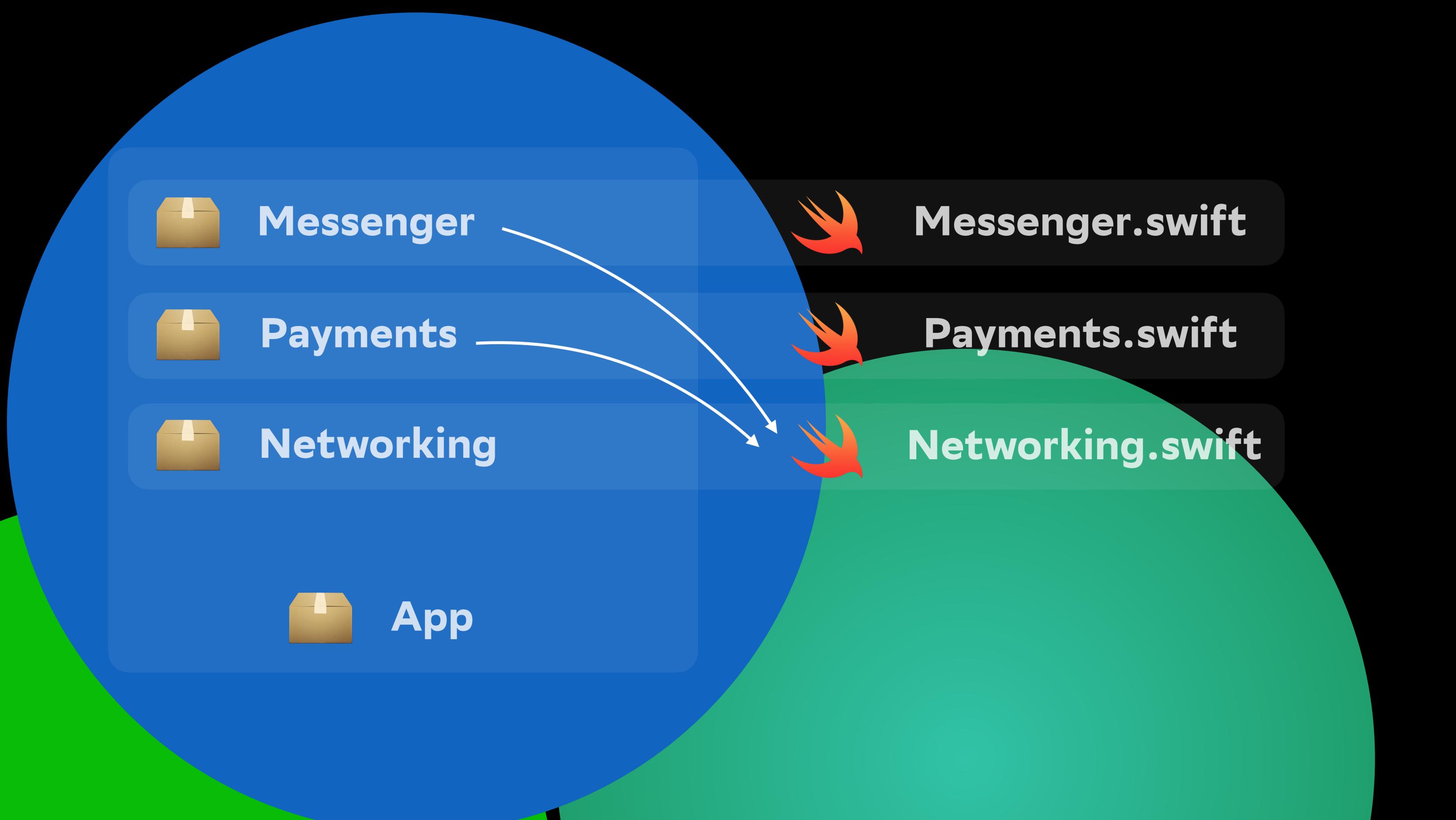
Messenger.swift



Payments.swift



Networking.swift





Messenger



Payments



Networking



App



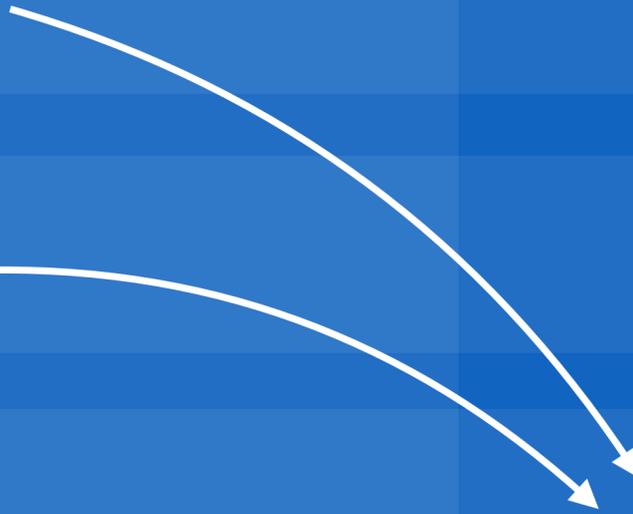
Messenger.swift



Payments.swift



Networking.framework

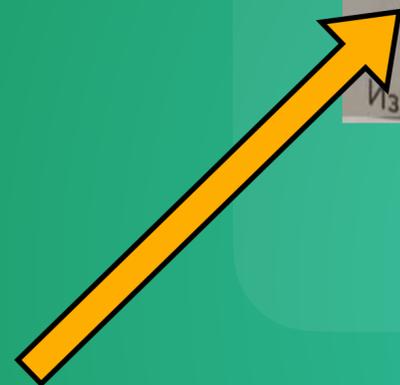
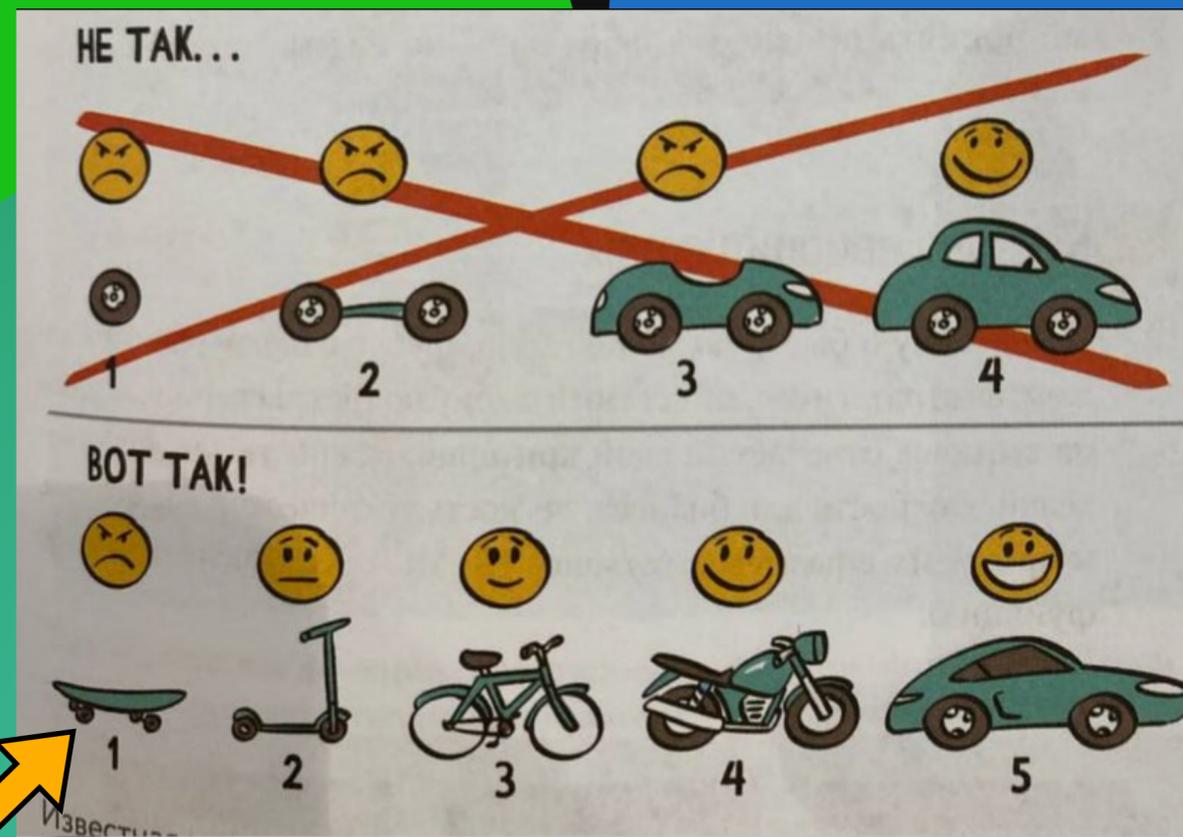


DEMO



Полная автоматизация





Мы находимся тут



Remote Cache



Messenger



Payments



Networking



App

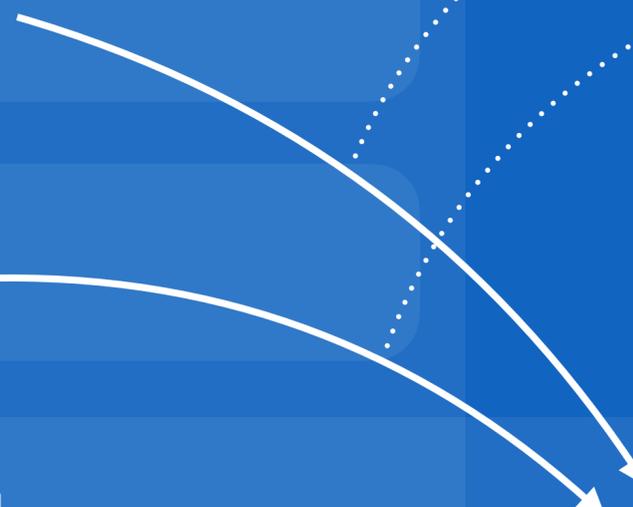
v1.0.1

v1.0.1

v1.0.1



Networking.framework



```
git log --pretty=format:'%H' -1 -- networking/sources
```

v 79f3dcec7c

v 79f3dcec7c

v 79f3dcec7c



Messenger



Payments



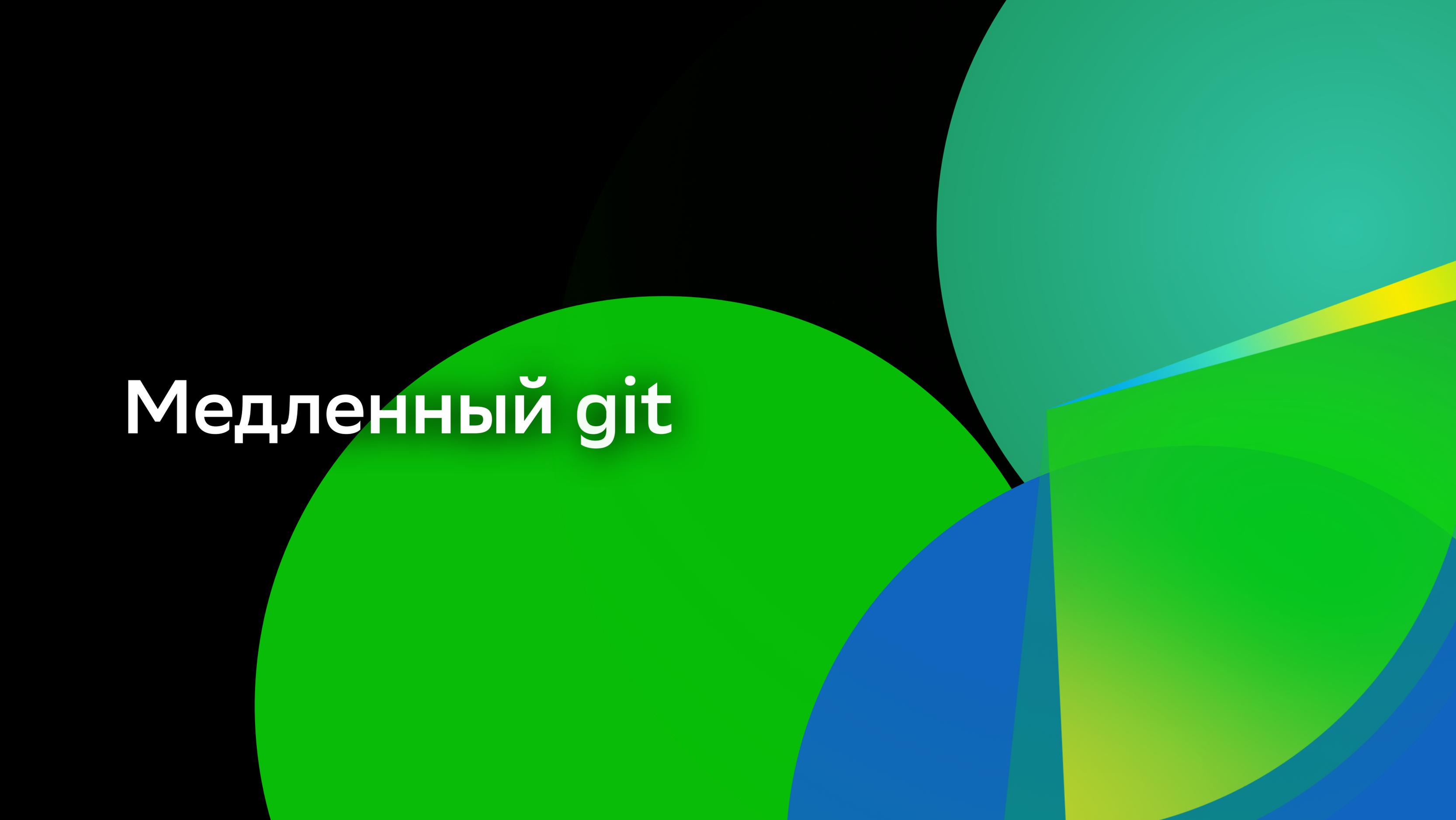
Networking



App



Networking.framework



Медленный git

Мультирепа

Монорепа

Неудобство при работе с несколькими репами

Гранулярная настройка пермишенов

Лучше отделяемость код

Быстрый clone, если не брать все исходники

Понятный git log

Крупный merge release -> develop

Высокая нагрузка на BitBucket/GitLab

Большой поток ПР-ов

Атомарность изменений

Низкий порог вхождения

Мультирепа

Монорепа

Неудобство при работе с несколькими репами

Гранулярная настройка пермишенов

Лучше отделяемость код

Быстрый clone, если не брать все исходники

Понятный git log

Крупный merge release -> develop

Высокая нагрузка на BitBucket/GitLab

Большой поток PR-ов

Атомарность изменений

Низкий порог вхождения

Мультирепа

Монорепа

Неудобство при работе с несколькими репами

Гранулярная настройка пермишенов

Лучше отделяемость код

Быстрый clone, если не брать все исходники

Понятный git log

Крупный merge release -> develop

Высокая нагрузка на BitBucket/GitLab

Большой поток PR-ов

Атомарность изменений

Низкий порог вхождения

Мультирепа

Монорепа

Неудобство при работе с несколькими репами

Гранулярная настройка пермишенов

Лучше отделяемость код

Быстрый clone, если не брать все исходники

Понятный git log

Крупный merge release -> develop

Высокая нагрузка на BitBucket/GitLab

Большой поток ПР-ов

Атомарность изменений

Низкий порог вхождения

Мультирепа

Монорепа

Неудобство при работе с несколькими репами

Гранулярная настройка пермишенов

Лучше отделяемость код

Быстрый clone, если не брать все исходники

Понятный git log

Крупный merge release -> develop

Высокая нагрузка на BitBucket/GitLab

Большой поток PR-ов

Атомарность изменений

Низкий порог вхождения



Remote Cache



Messenger



Payments



Networking



App

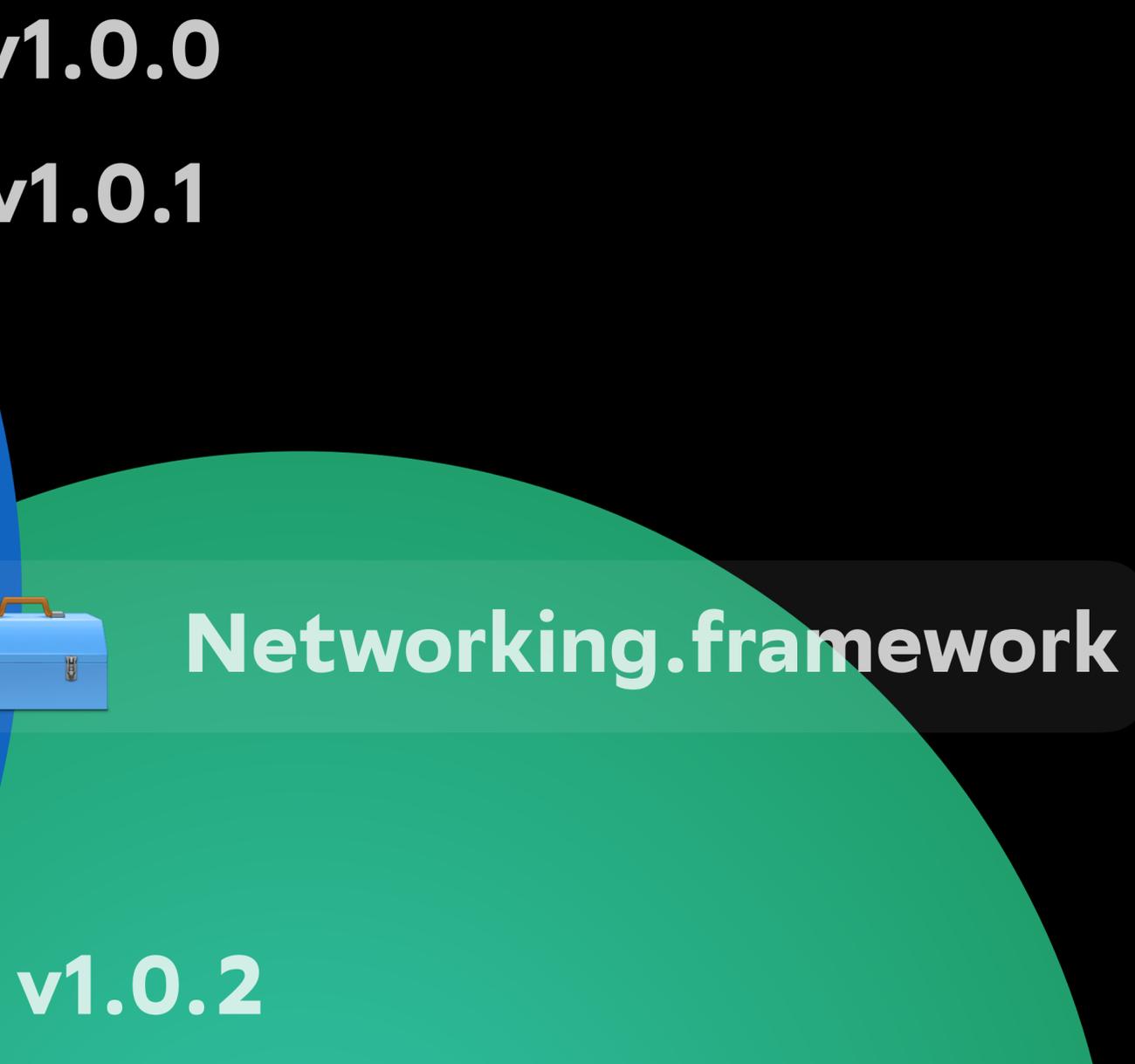
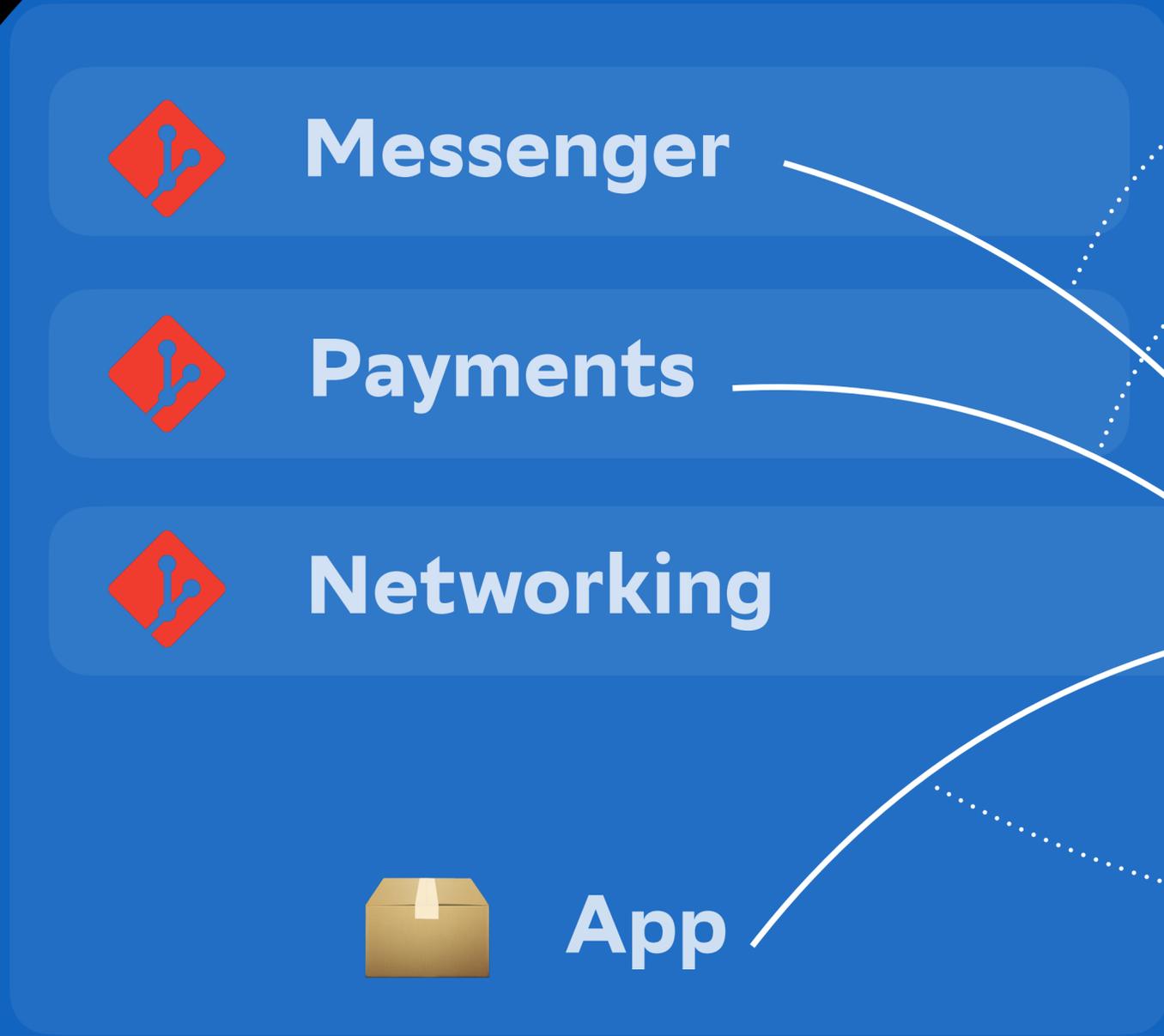
v1.0.0

v1.0.1



Networking.framework

v1.0.2



Синхронизация изменений

Networking



Messenger



Payments



RADAR



- Networking
- Messenger
- Payments

Синхронизация изменений

Networking



Messenger



Payments

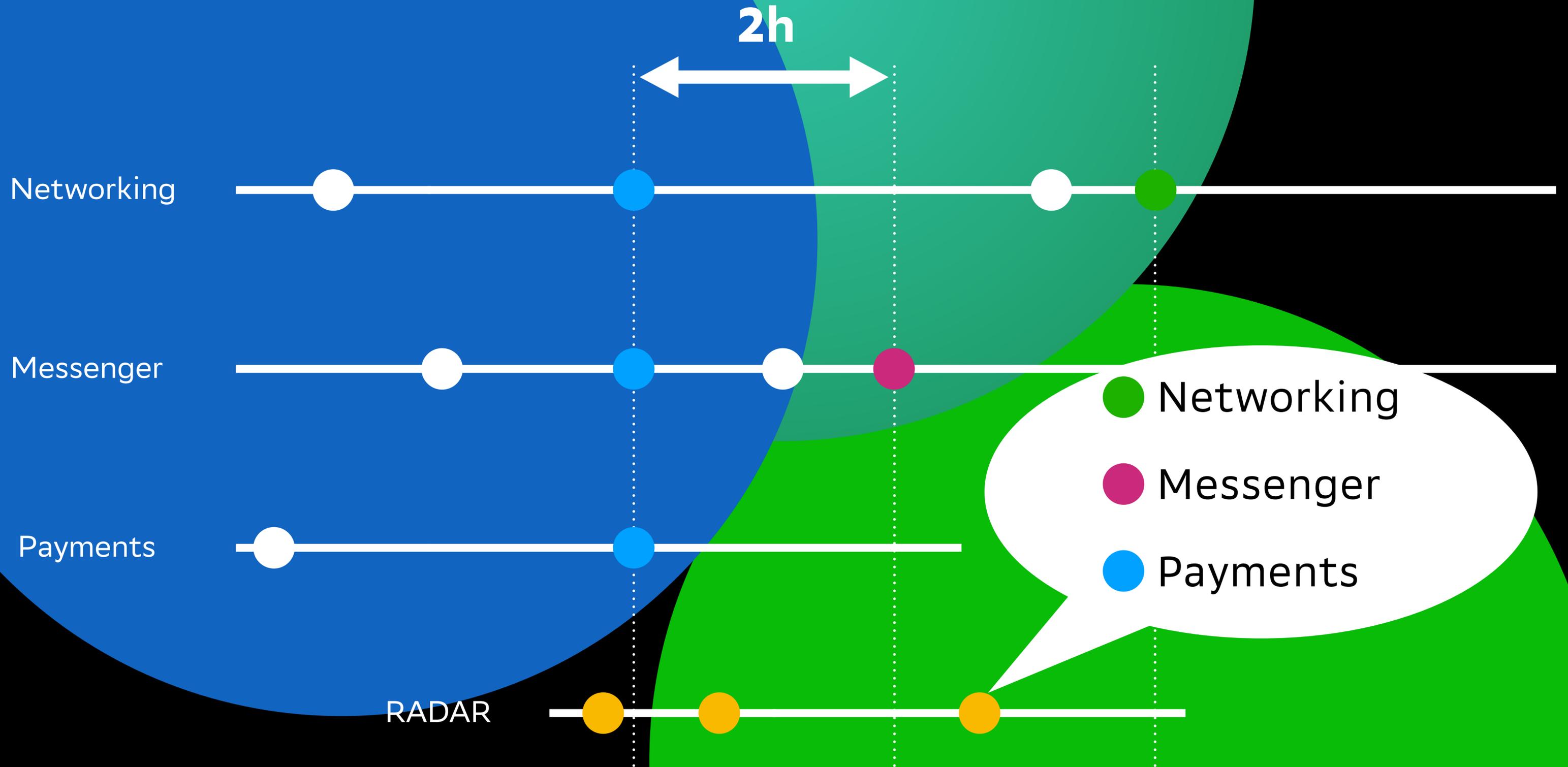


RADAR

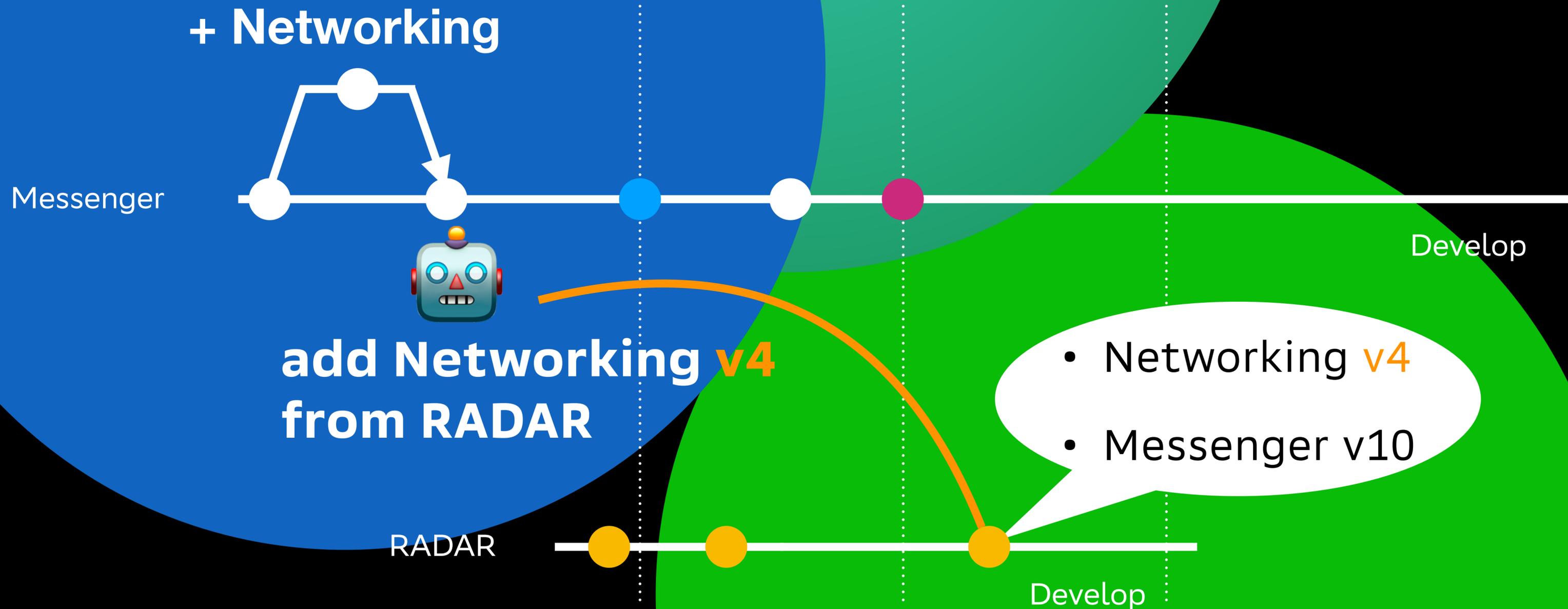


-  Networking
-  Messenger
-  Payments

Синхронизация изменений



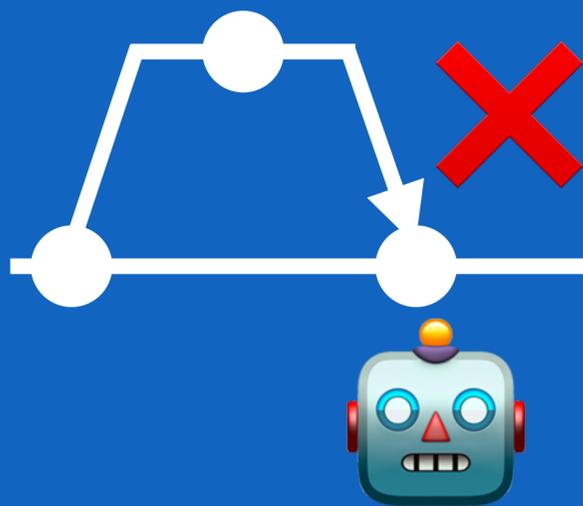
Взаимодействие с другими



Взаимодействие с другими

+ CryptoSwift

Messenger



Нужно сначала добавить в
общий список зависимостей

RADAR



• тут нет CryptoSwift

Develop

* Но есть свой велосипед на CommonCrypto

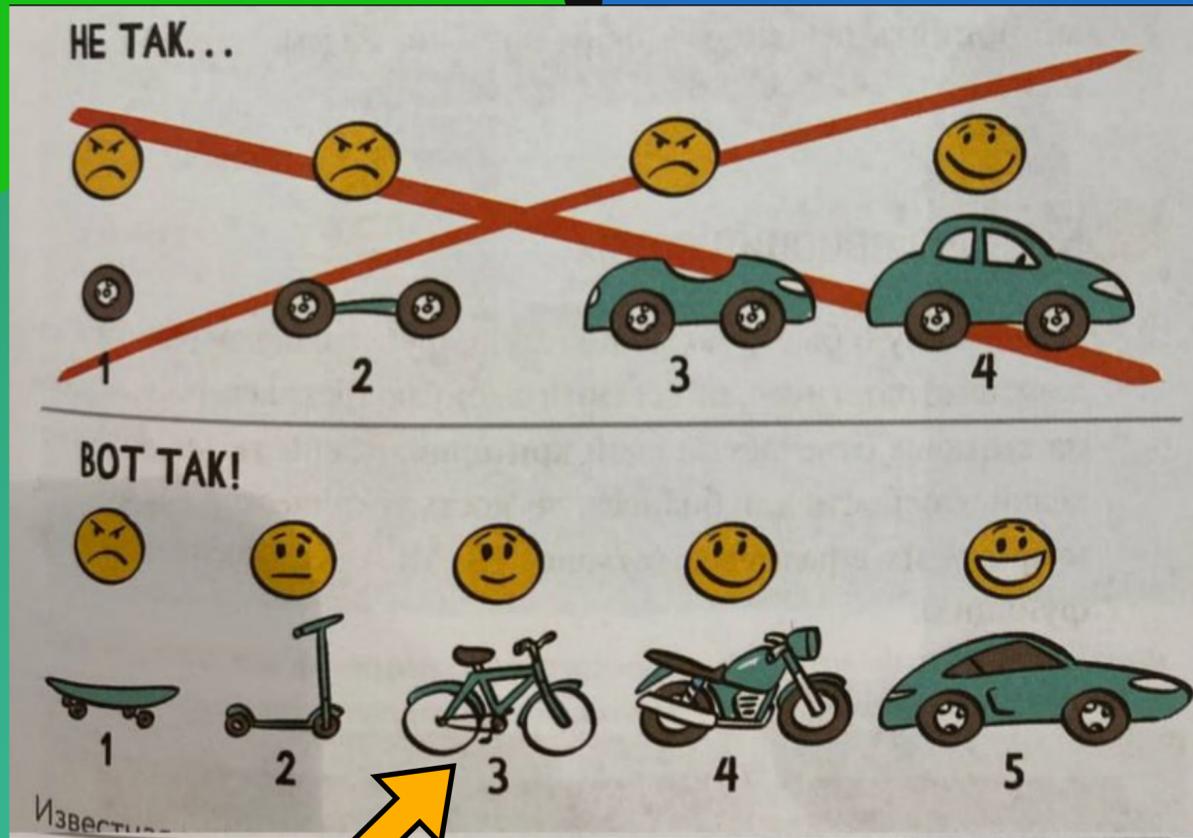
Develop

DEMO



[https://github.com/wowlocal/
sampleapp](https://github.com/wowlocal/sampleapp)

[https://github.com/wowlocal/
ios_bs](https://github.com/wowlocal/ios_bs)



Мы находимся тут

Что получили

- 1 pr для попадания изменений в следующую сборку, не нужно “поднимать версию”

Что получили

- 1 pr для попадания изменений в следующую сборку, не нужно “поднимать версию”
- Новые API становятся доступны каждые 2 часа

Что получили

- 1 pr для попадания изменений в следующую сборку, не нужно “поднимать версию”
- Новые API становятся доступны каждые 2 часа
- Нет более с резолвом зависимостей

Что получили

- 1 pr для попадания изменений в следующую сборку, не нужно “поднимать версию”
- Новые API становятся доступны каждые 2 часа
- Нет более с резолвом зависимостей
- Все разработчики живут на гарантированно совместимом состоянии

Что получили

- 1 pr для попадания изменений в следующую сборку, не нужно “поднимать версию”
- Новые API становятся доступны каждые 2 часа
- Нет более с резолвом зависимостей
- Все разработчики живут на гарантированно совместимом состоянии
- Контроль за зависимостями проекта в одном месте

Что получили

- 1 pr для попадания изменений в следующую сборку, не нужно “поднимать версию”
- Новые API становятся доступны каждые 2 часа
- Нет более с резолвом зависимостей
- Все разработчики живут на гарантированно совместимом состоянии
- Контроль за зависимостями проекта в одном месте
- Build Cache шэрится между разработчиками и используется в PR пайплайнах

Что получили

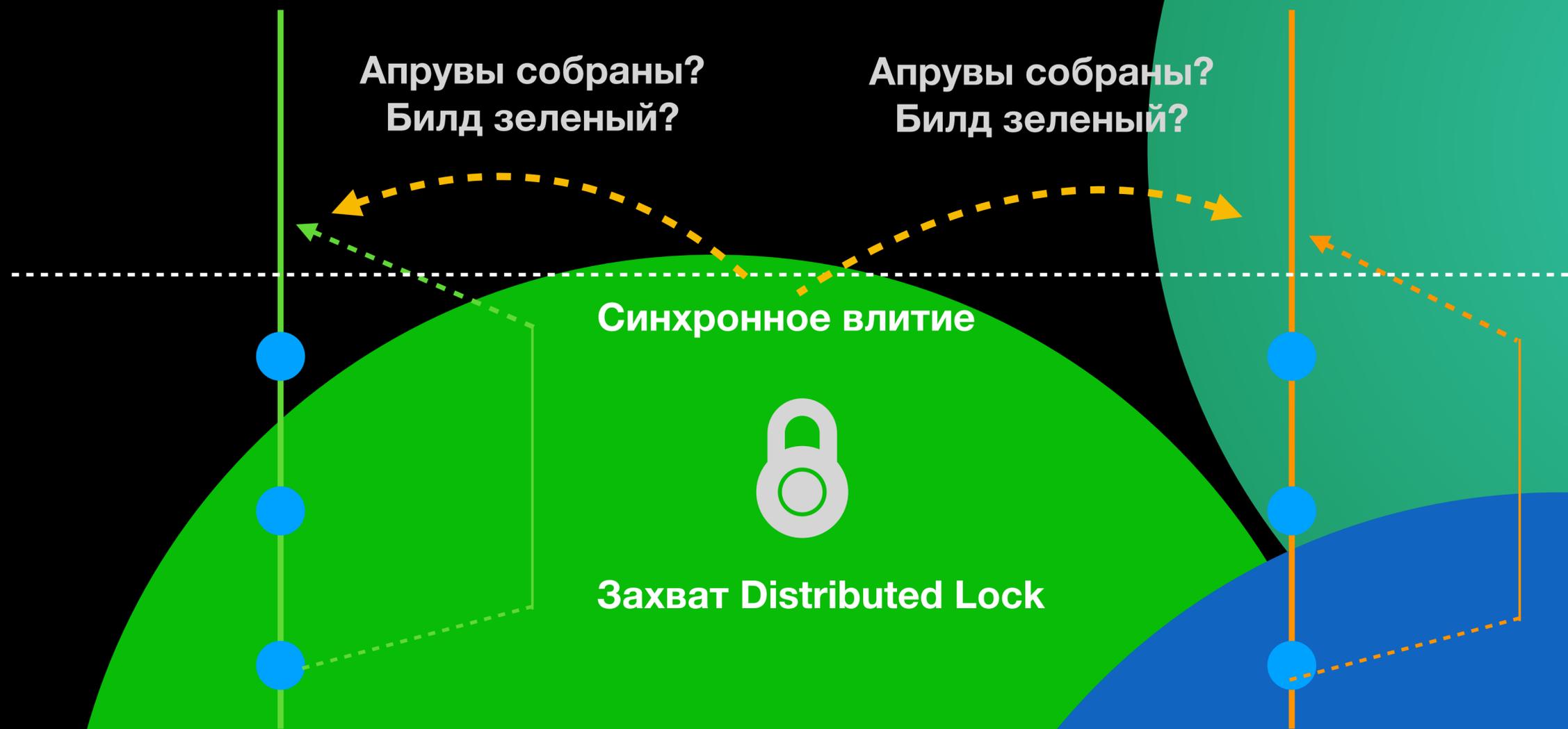
- 1 pr для попадания изменений в следующую сборку, не нужно “поднимать версию”
- Новые API становятся доступны каждые 2 часа
- Нет более с резолвом зависимостей
- Все разработчики живут на гарантированно совместимом состоянии
- Контроль за зависимостями проекта в одном месте
- Build Cache шэрится между разработчиками и используется в PR пайплайнах
- Быстрый git

Чего-то не хватает

Глобальный рефакторинг

☮ Imagine there is no Repos

Выполнение всех условий влития ПРа



- A** - DBIOSP-0000:@M Изменением названия публичного метода
- B** - DBIOSP-0000:@S Переименование всех использований

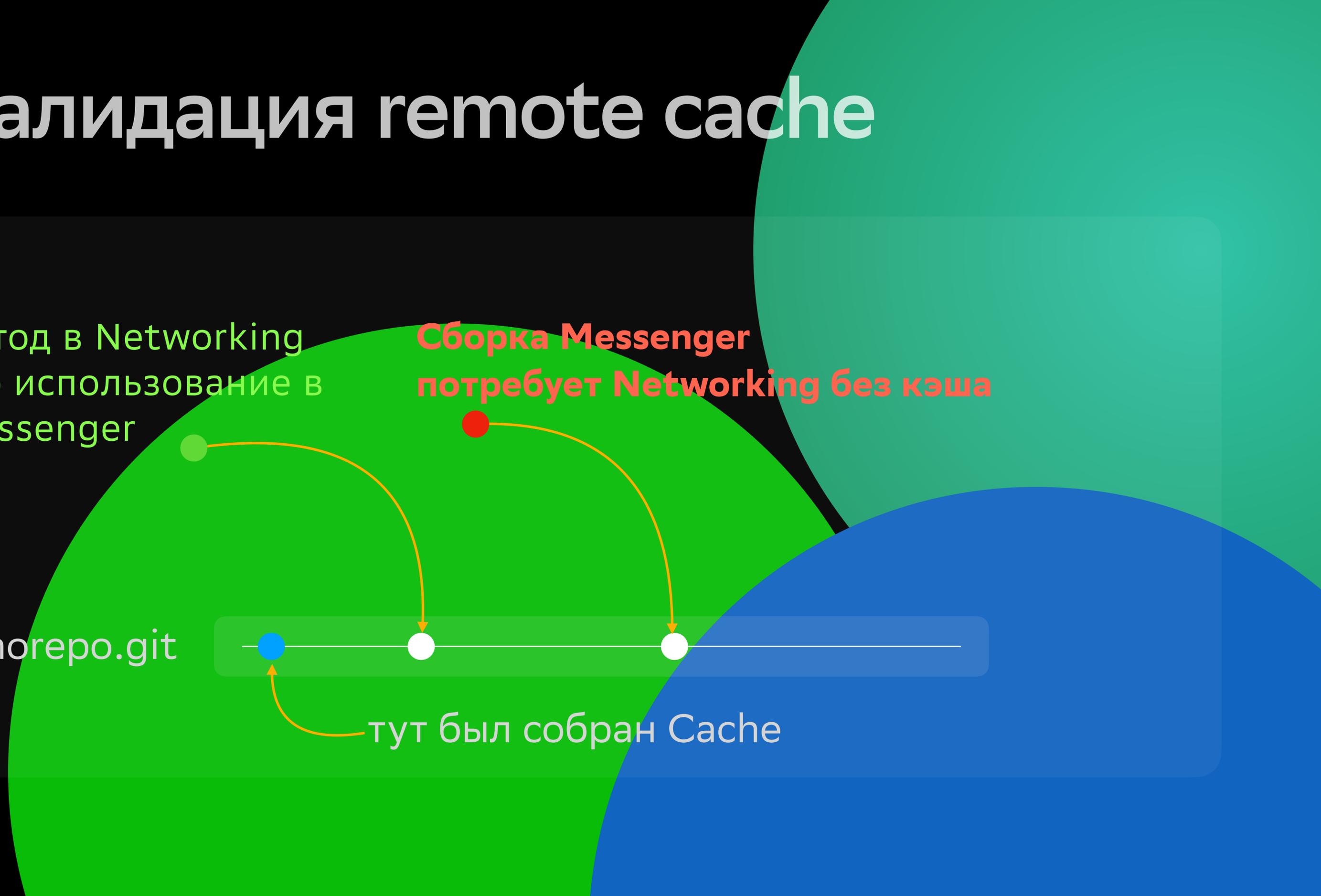
Инвалидация remote cache

- + метод в Networking
- + его использование в Messenger

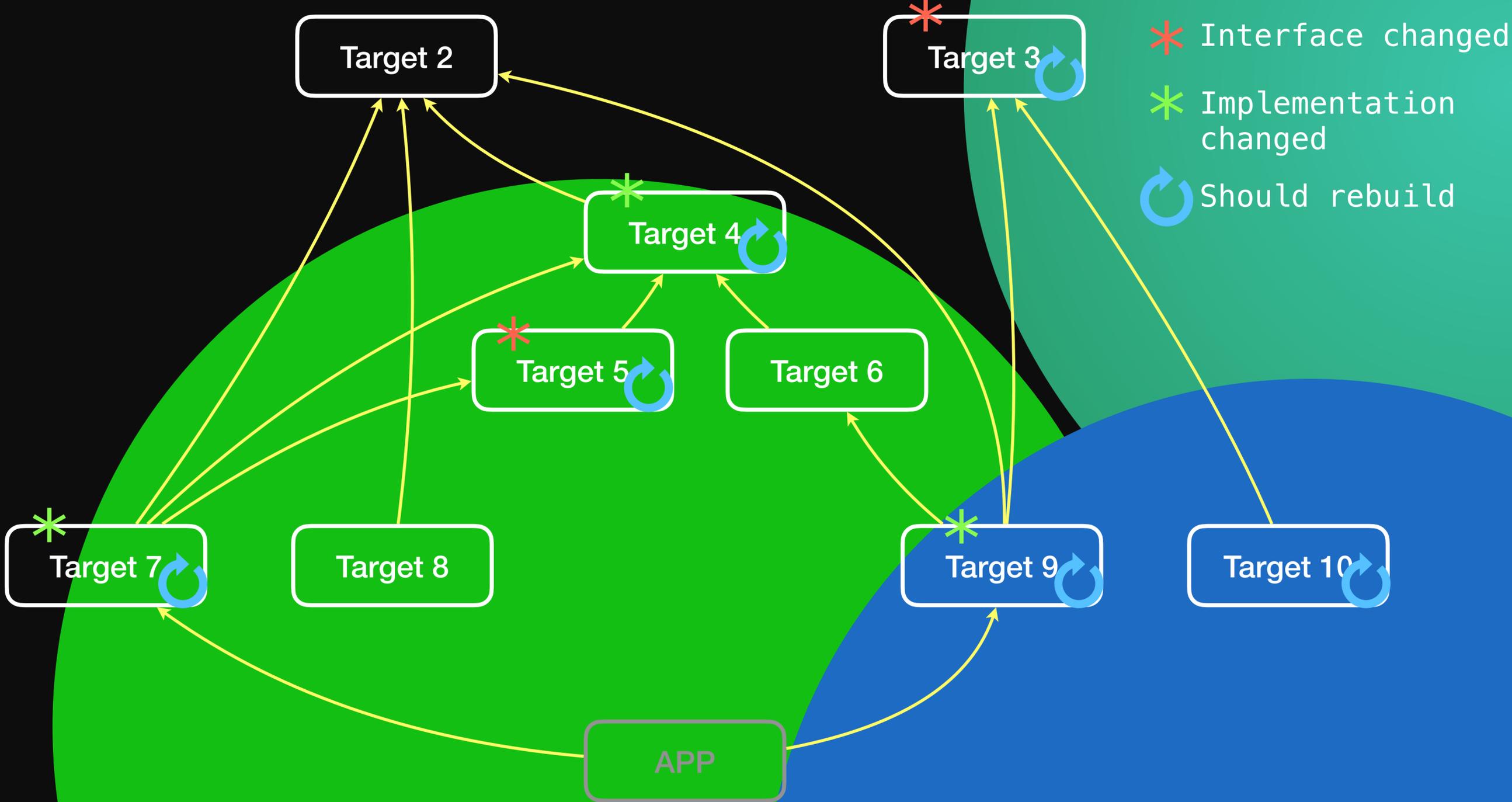
**Сборка Messenger
потребуется Networking без кэша**

Monorepo.git

тут был собран Cache



Импакт анализ





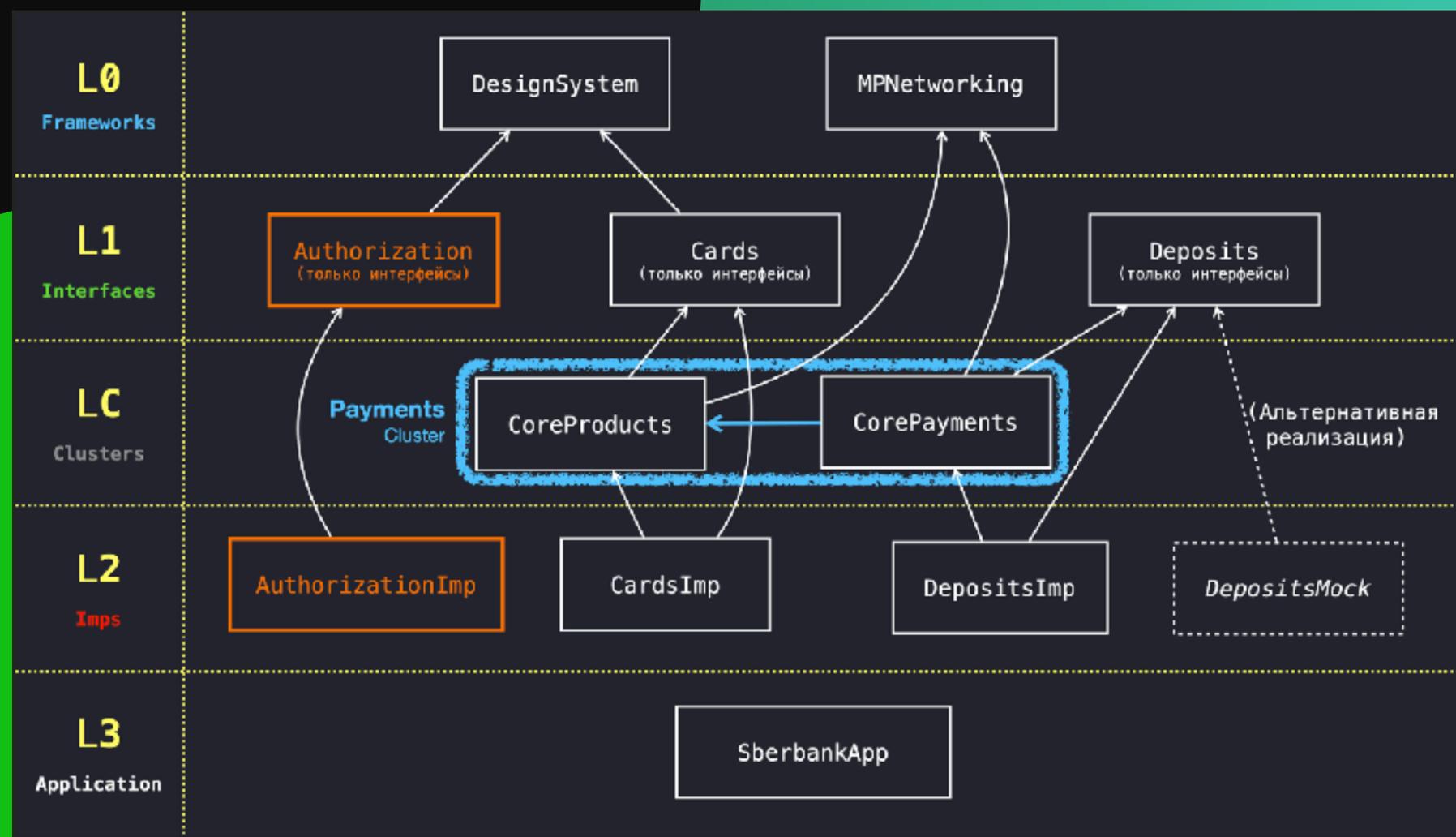
Переиспользовать нельзя перепроверять:

Баланс между скоростью и надежностью



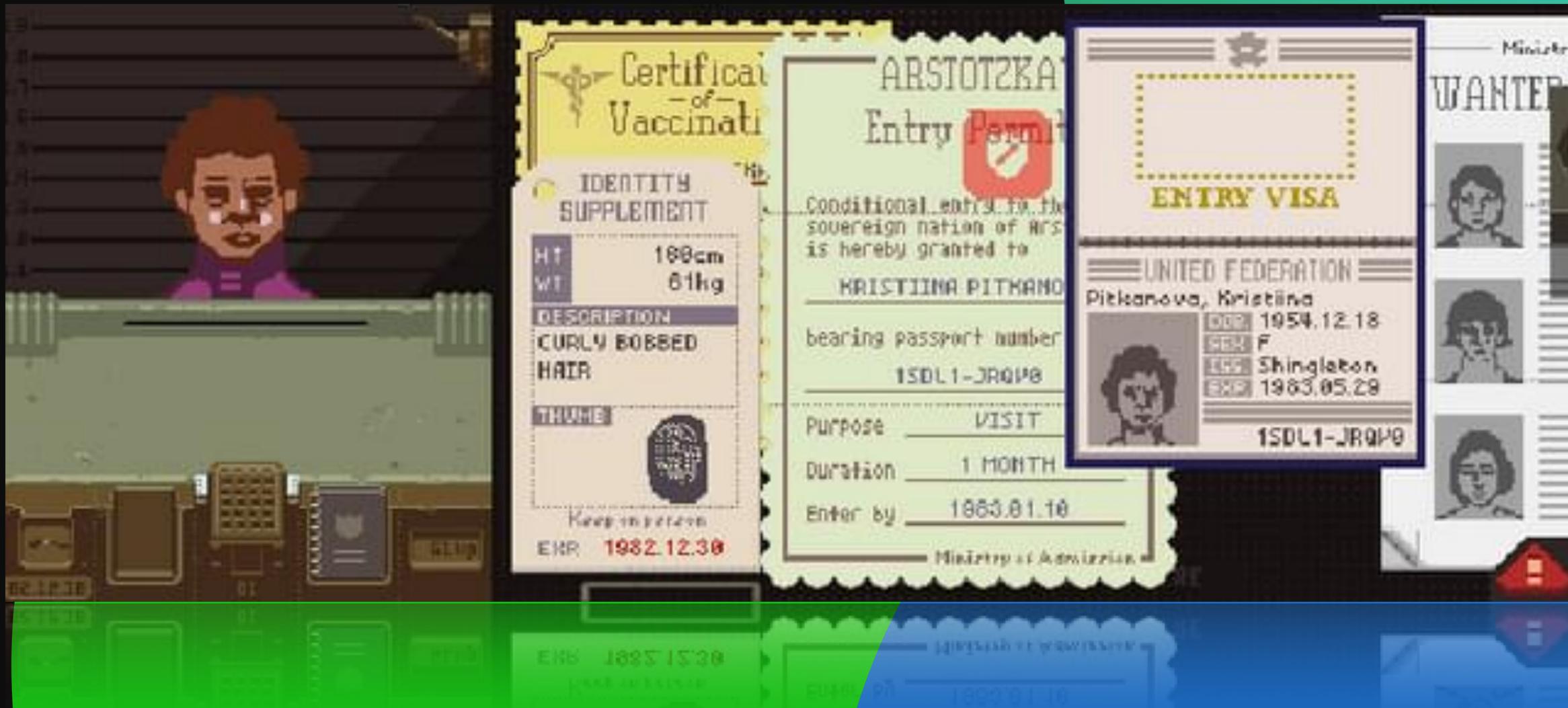
Вероника Макаровская
iOS-MobilePlatform

Архитектура



Граф растет преимущественно горизонтально

Валидация всего что возможно на СІ



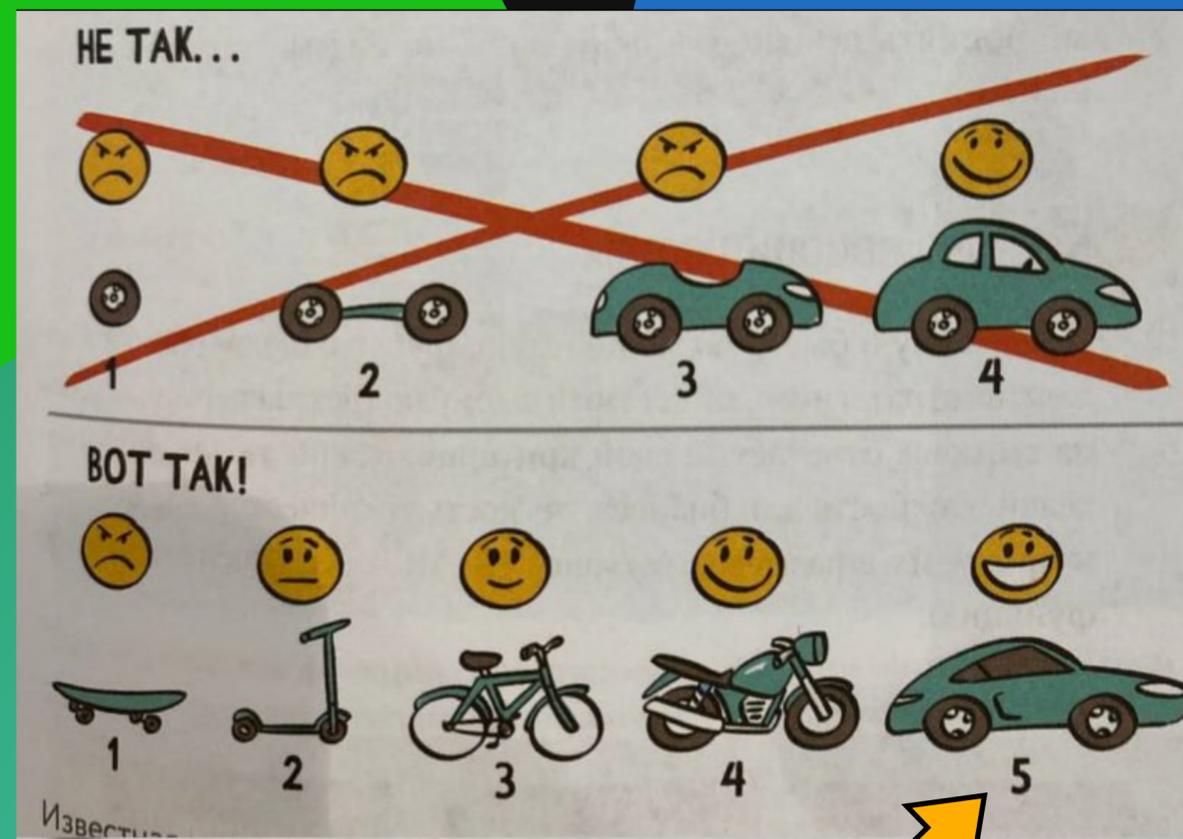
Интеграция инструментов

CI / CD

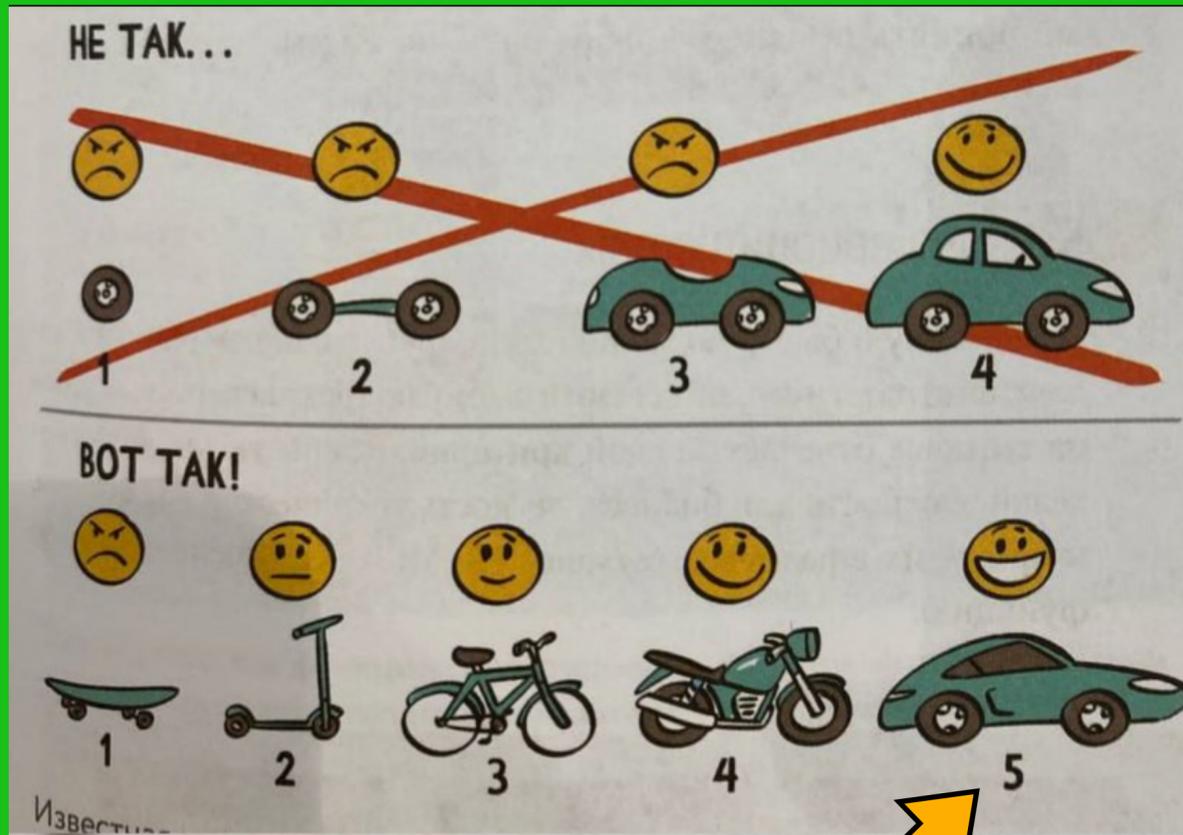
Скорость сборки

Архитектура

SCM



Как оказаться тут?



Как оказаться тут?

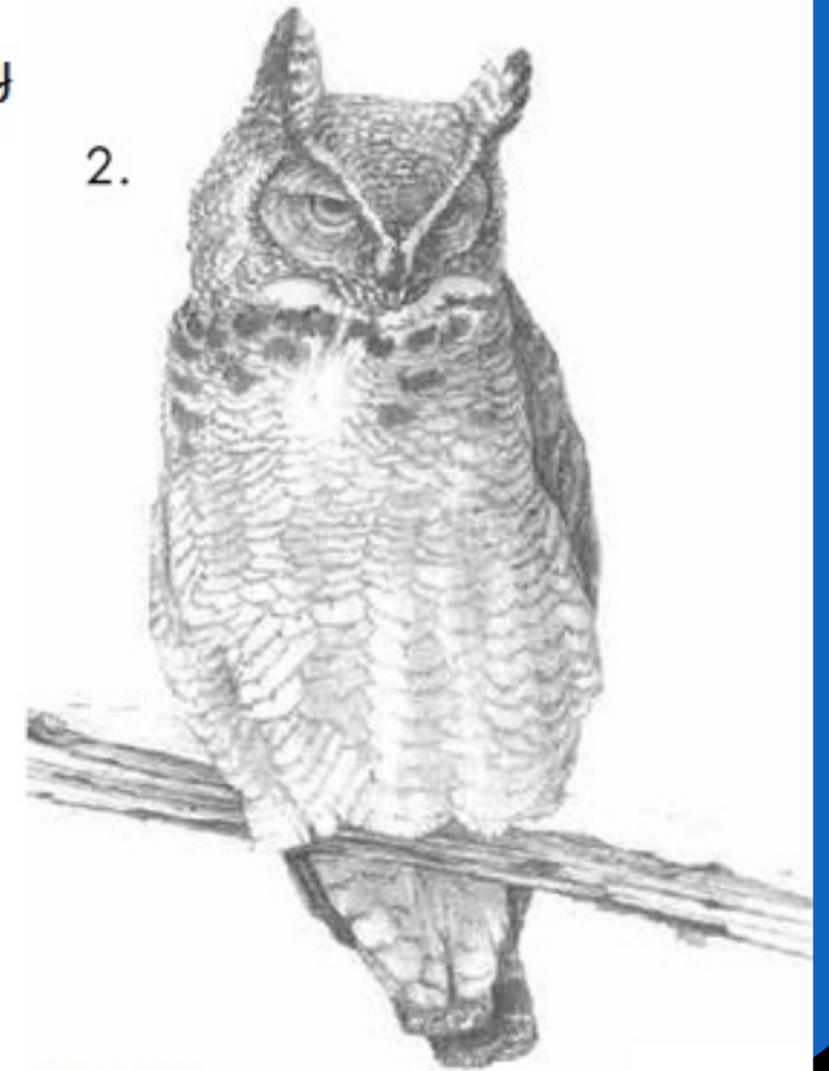
Как нарисовать сову

1.



1. Рисуем кружочки

2.



2. Рисуем остатоқ совы

Короткий путь



Приходите работать в Сбер :)



✔ Спасибо



за ваше внимание



 Контакты



Миша Харитончик

 Команда iOS-MobilePlatform Сбера

 tg: resultBuilder

