

Алексей Минин

iOS-Core

Переезд на Swift 6: строим графы для успешной миграции

The logo consists of the lowercase letters 'hh' in a bold, sans-serif font, centered within a white circle. The background of the slide is a gradient from dark red on the left to light orange on the right.



Алексей Минин

iOS-разработчик в Core-команде

- Упрощаю жизнь коллегам из продуктовых команд
- Пять лет в iOS-разработке

~600

человек в ИТ, data, продукт

60+

команд в техническом департаменте

400+

стендов

AI

используется в 13 командах

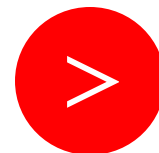
1 млн+

событий обрабатываем
и храним телеметрию со всех
сервисов без сэмплирования

99,98%

crashfree-сессий
мобильных приложений

О чём доклад?



Миграция на Swift 6: защита от датарейсов в многомодульном проекте

Новый релиз Swift 6

01

Проверки
для защиты
от датарейсов,
связанные со Swift
concurrency

02

После использования
получил несколько
тысяч ошибок
компиляции

03

Решил разобраться,
как правильно пойти к
процессу переезда на
Swift 6

04

Первые шаги и столкновения с реальностью

Многомодульность

Проект, о котором пойдёт речь, состоит из 160 модулей, что делает невозможным простое разделение фич по командам

Временные затраты

Разделение по код-овнерам оказалось длительным и требовало бы «костылей» и повторных проходов по измененным файлам

Модульная архитектура iOS-проекта

Два приложения:

соискатели

1

работодатели

2

Разделение на фичи позволяет командам независимо работать над разными частями приложения

Фичи: изолированные модули приложения

Фича —

это отдельный Xcode-проект с собственным кодом, ресурсами и тестами

Они подключаются к другим фичам и основному приложению, обеспечивая независимую разработку

Типы фичей:

- фундаментальные
- шаренные — для переиспользования кода
- продуктовые — специфичные для приложения

Tuist: генерация проектов из манифестов

Структура

1

Удобное управление структурой проекта

Генерация

2

Автоматическая генерация Xcode-проектов

Мердж

3

Предотвращение конфликтов при слиянии веток

Статья про Tuist



Миграция кода: подводные камни и последовательный подход

Учитывайте масштаб проекта
и потенциальные риски

Инструменты, `@preconcurrency import`
и `@unchecked sendable` не всегда
являются универсальным решением

Проблемы с Sendable и Strict concurrency checking

Несовместимость модулей

Включение Strict concurrency checking в одном модуле при отсутствии Sendable в другом может привести к конфликтам

01

Костыли и каскадные изменения

Необходимость в костылях для обхода проблем с Sendable увеличивает риск каскадных изменений

02

Эксперимент и его последствия

Неудачный опыт

Экспериментальный перевод одной из фич привел к созданию большого количества костылей

01

Временные решения

Появилась необходимость в удалении временных элементов (@preconcurrency import и других)

02

Увеличение трудозатрат

Пришлось бы возвращаться к коду повторно для удаления временных решений

03

Последовательность и обоснованность миграции

Важность последовательности

Для больших проектов важна последовательная миграция кода

Обоснование изменений

Каждый шаг миграции должен быть обоснован

Логика формирования сиграции Swift

Рекомендации Apple

Перевод начинается с «листьев» проекта для
минимизации каскадных изменений

Почему листья?

Листовые модули
минимизируют
влияние на проект

Снижение риска каскадных изменений в
проекте

План итераций

- 1 Разделение таргетов
- 2 Итерация за итерацией
Берем либо листья, либо зависящие таргеты
- 3 Последовательность
Учитываем реализованные фичи
на каждой итерации

Выбор таргетов вместо фич

Преимущества таргетов

- Избежание циклических зависимостей
- Возможность автоматизации анализа

Рассмотрение таргетов как атомарных единиц миграции

Предотвращение возникновения циклов зависимостей между фичами

Tuist как инструмент для генерации графа зависимостей

Таргеты

Создание графа по таргетам

01

Визуализация

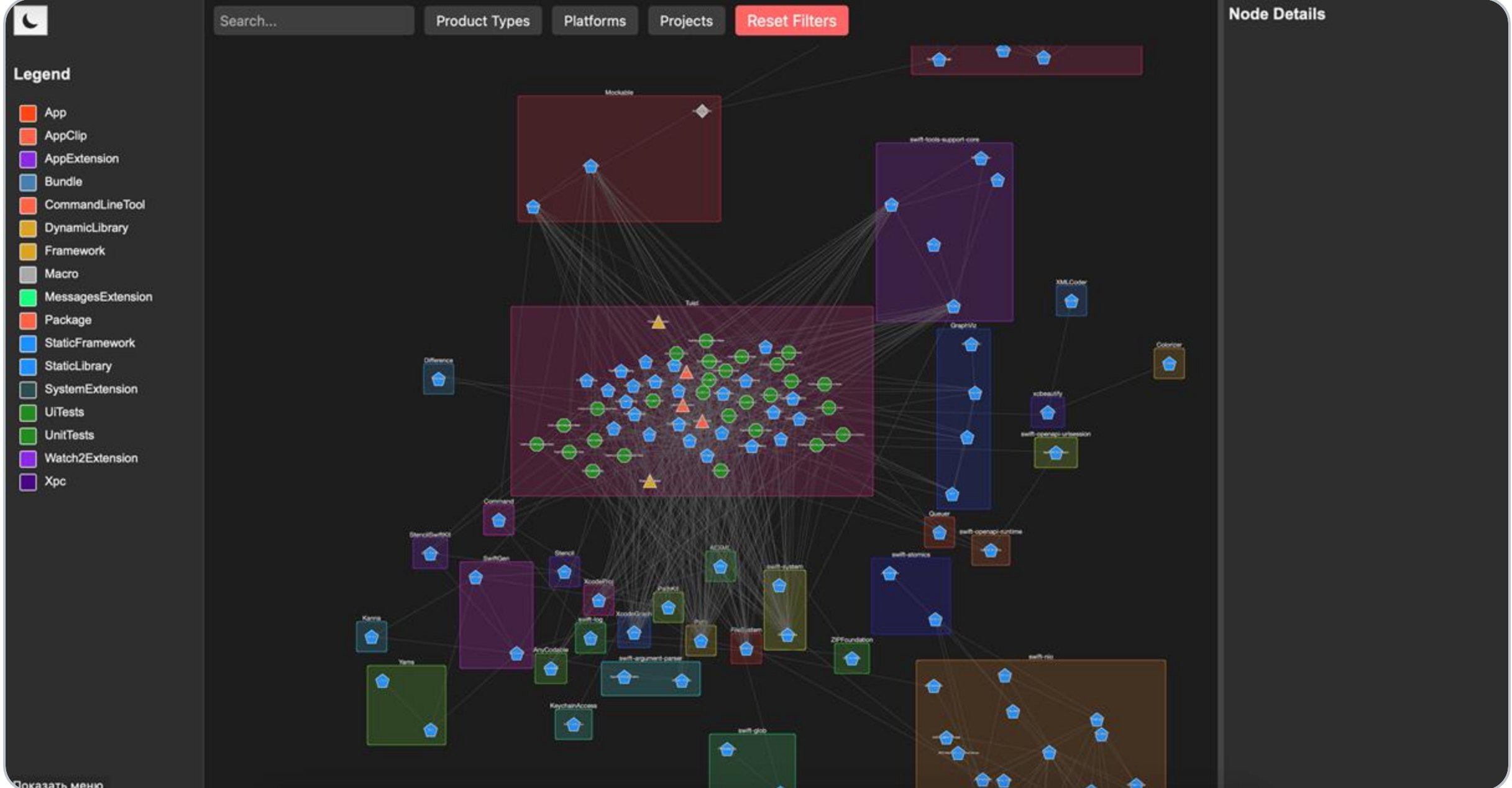
Удобный инструмент для построения графов

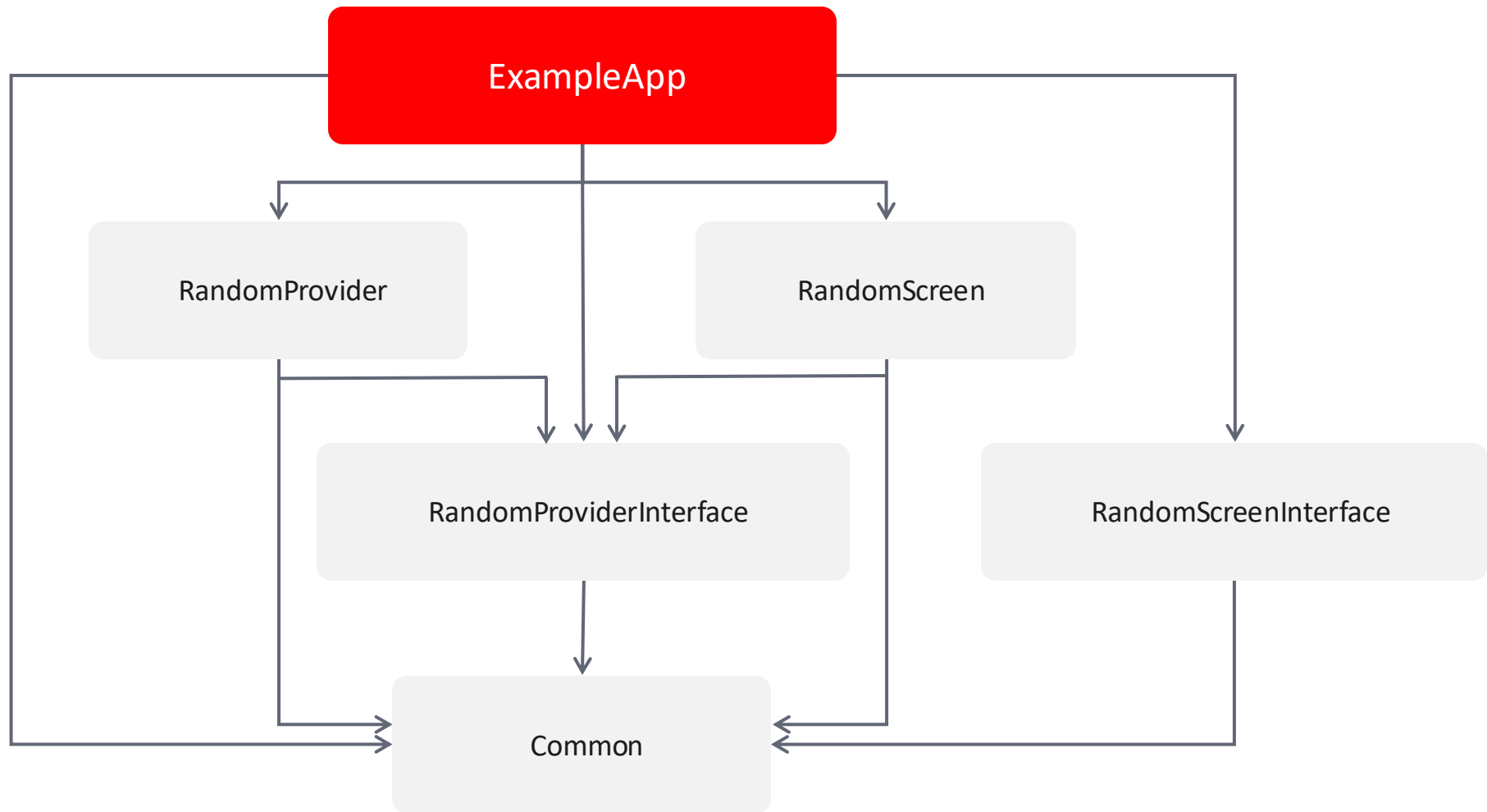
02

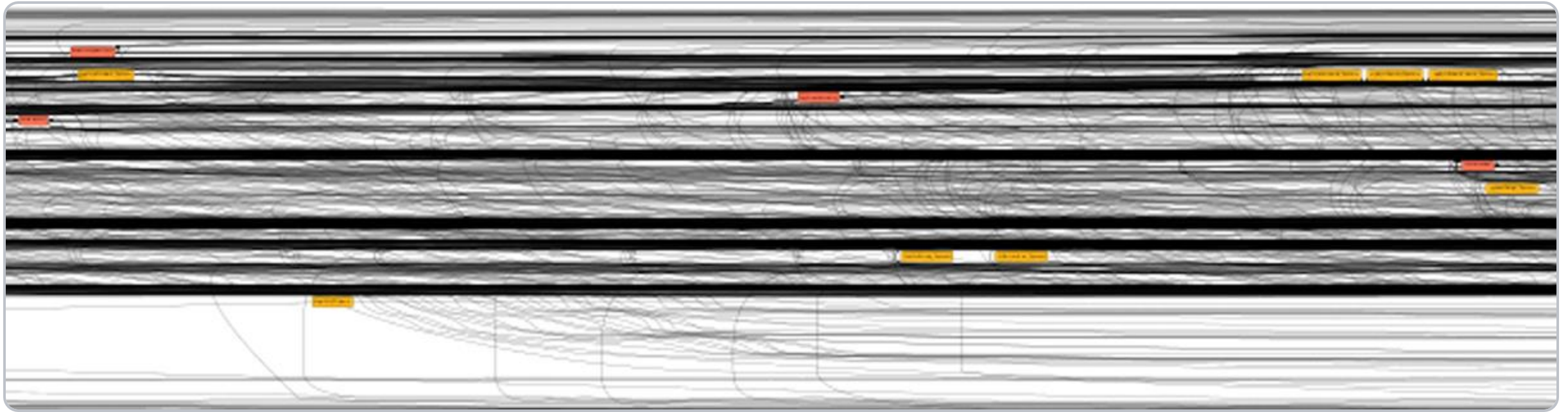
Автоматизация

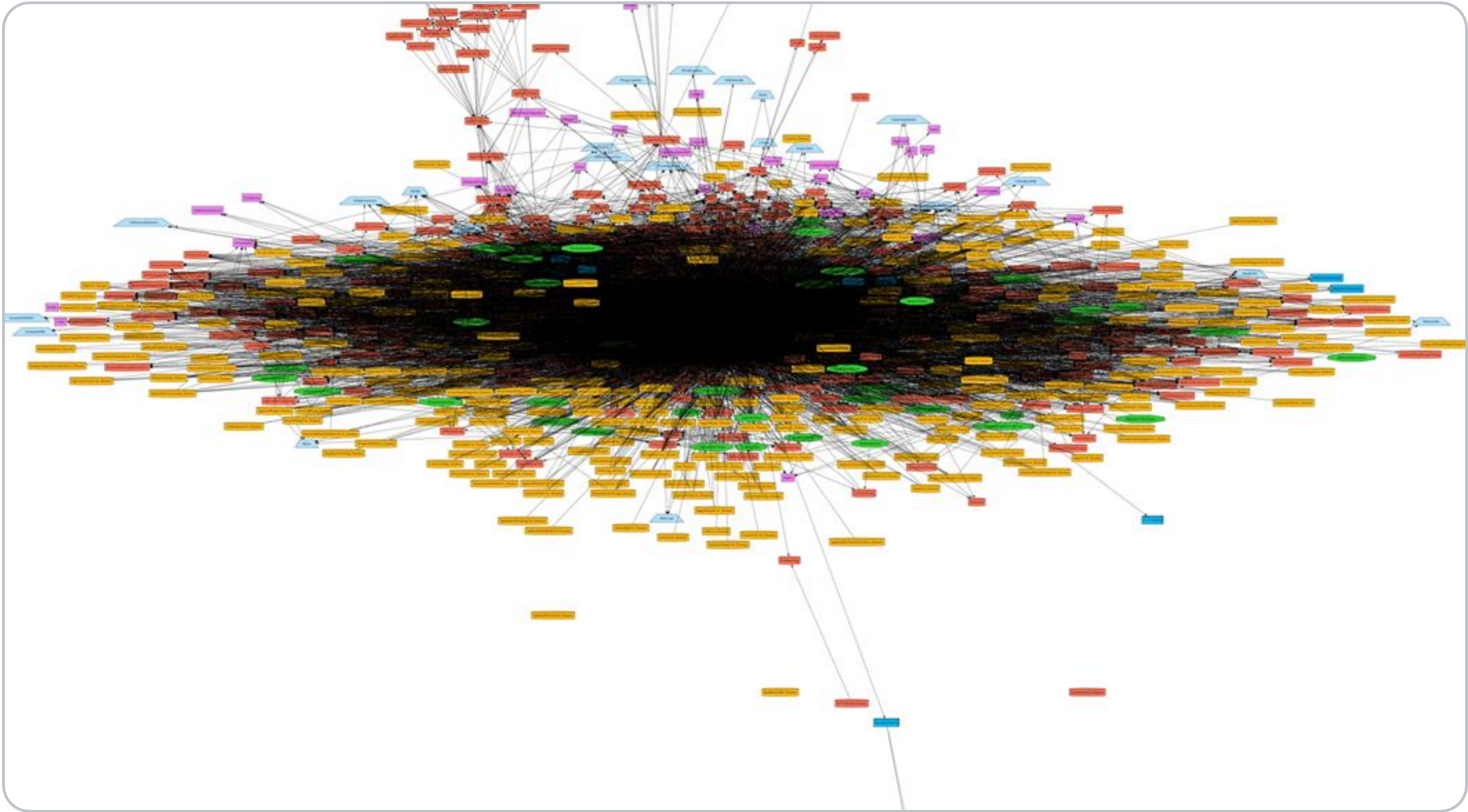
Автоматическое выявление зависимостей

03









Очистка графа от лишних зависимостей

Удаление

Автоматическое удаление динамических таргетов

01

Сохранение

Учет внешних зависимостей

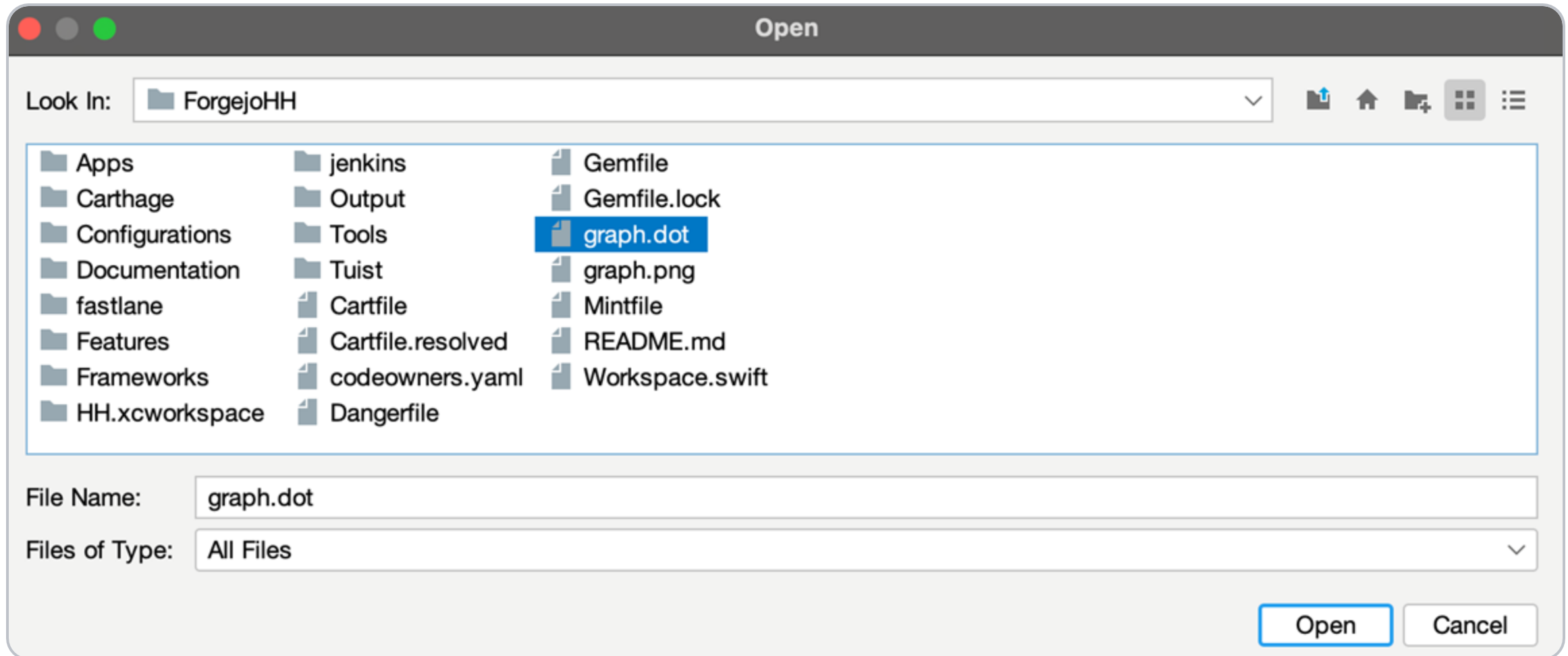
02

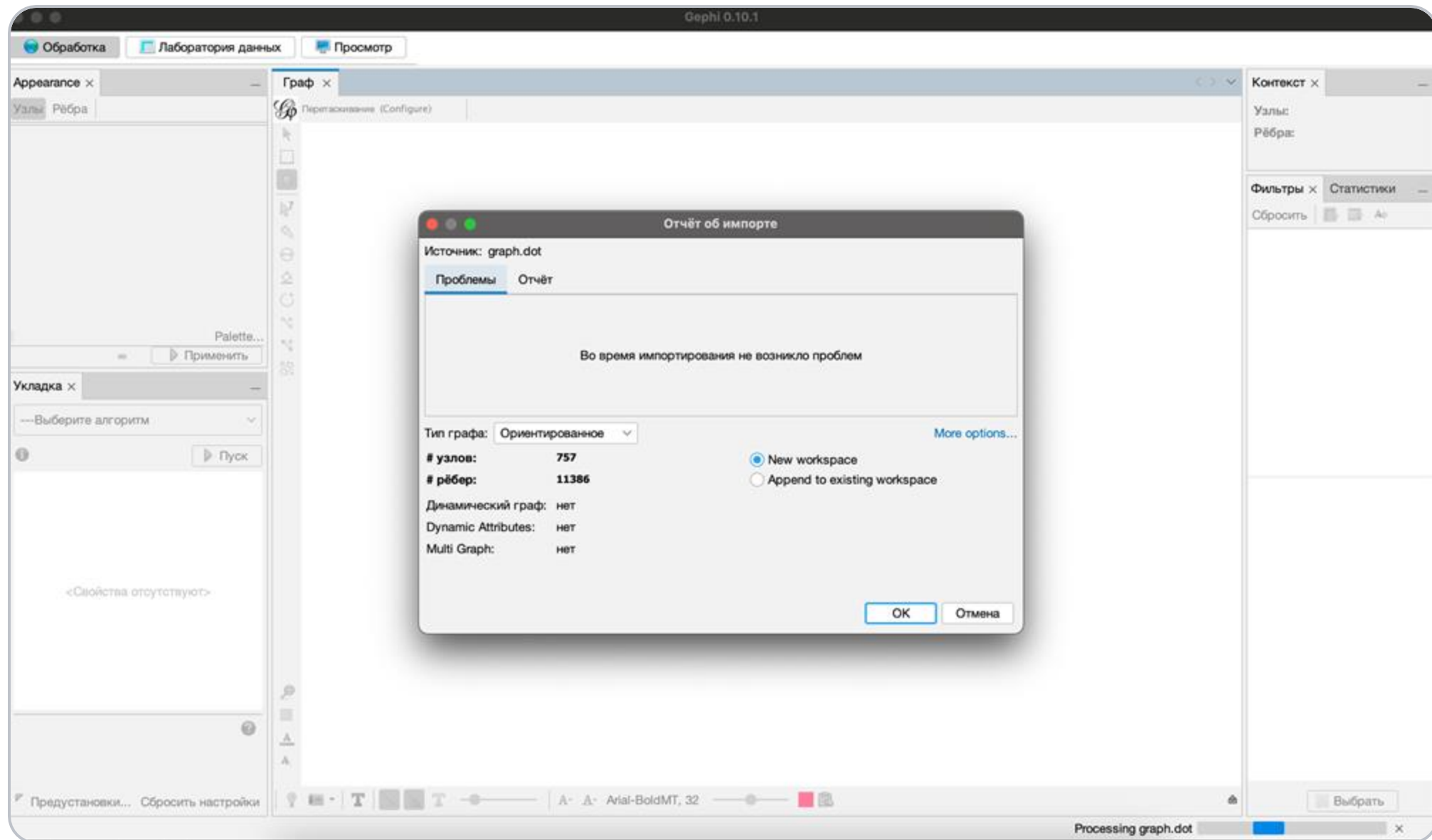
Исключение

Удаление системных библиотек

03

Самое интересное: начинаем визуализировать





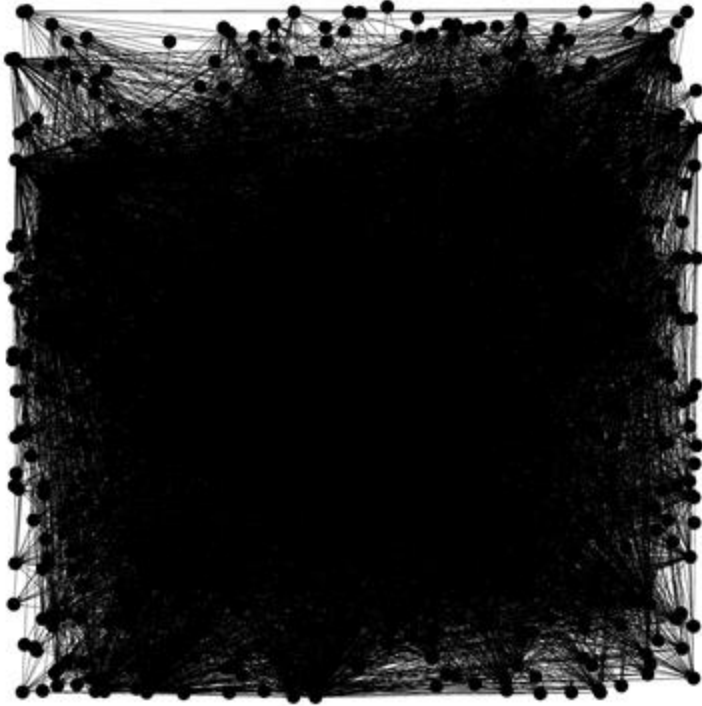
Gephi 0.10.1 - Проект 3

Обработка | Лаборатория данных | Просмотр | graph x

Ареанжжэ ×
Узлы Рёбра
Unique Partition Ranking
#c0c0c0
Применить

Граф ×
Перетаскивание (Configure)

Укладка ×
---Выберите алгоритм
Пуск
<Свойства отсутствуют>



Контекст ×
Узлы: 757
Рёбра: 11386
Ориентированный граф

Фильтры × | Статистики
Сбросить
Библиотека
Атрибуты
Динамика
Операторы
Рёбра
Топология
Сохранённые запросы


Наборы фильтров
Перетаскийте фильтр сюда

Выбрать

Предустановки... Сбросить настройки

Аrial-BoldMT, 32



Appearance x


Узлы Рёбра 

Unique Partition Ranking

Degree



Color:

Слайны...  


 Применить

Укладка x

---Выберите алгоритм

  Пуск

Граф x

 Перетаскивание (Configure)



Graph visualization area

Gephi 0.10.1 - Проект 3


Обработка | Лаборатория данных | Просмотр | graph x



Граф x

Узлы Рёбра

Unique Partition Ranking

Degree

Color: 

Слайды...  

Применить

Укладка x

---Выберите алгоритм

Пуск

<Свойства отсутствуют>




Предустановки... Сбросить настройки

Перетасование (Configure)

Контекст x

Узлы: 757
Рёбра: 11386
Ориентированный граф


Фильтры x | Статистики

Сбросить   

Библиотека

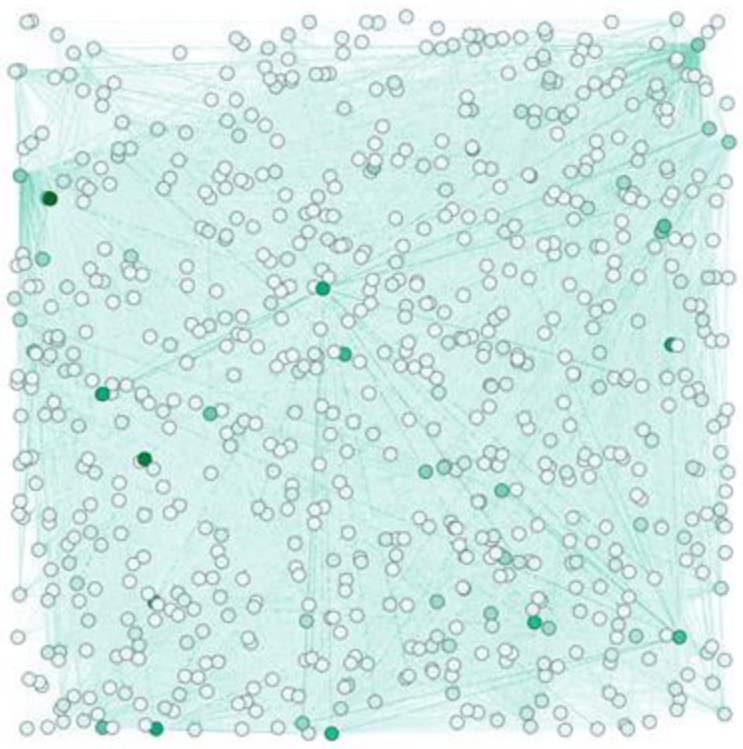
- Атрибуты
- Динамика
- Операторы
- Рёбра
- Топология
- Сохранённые запросы

Наборы фильтров

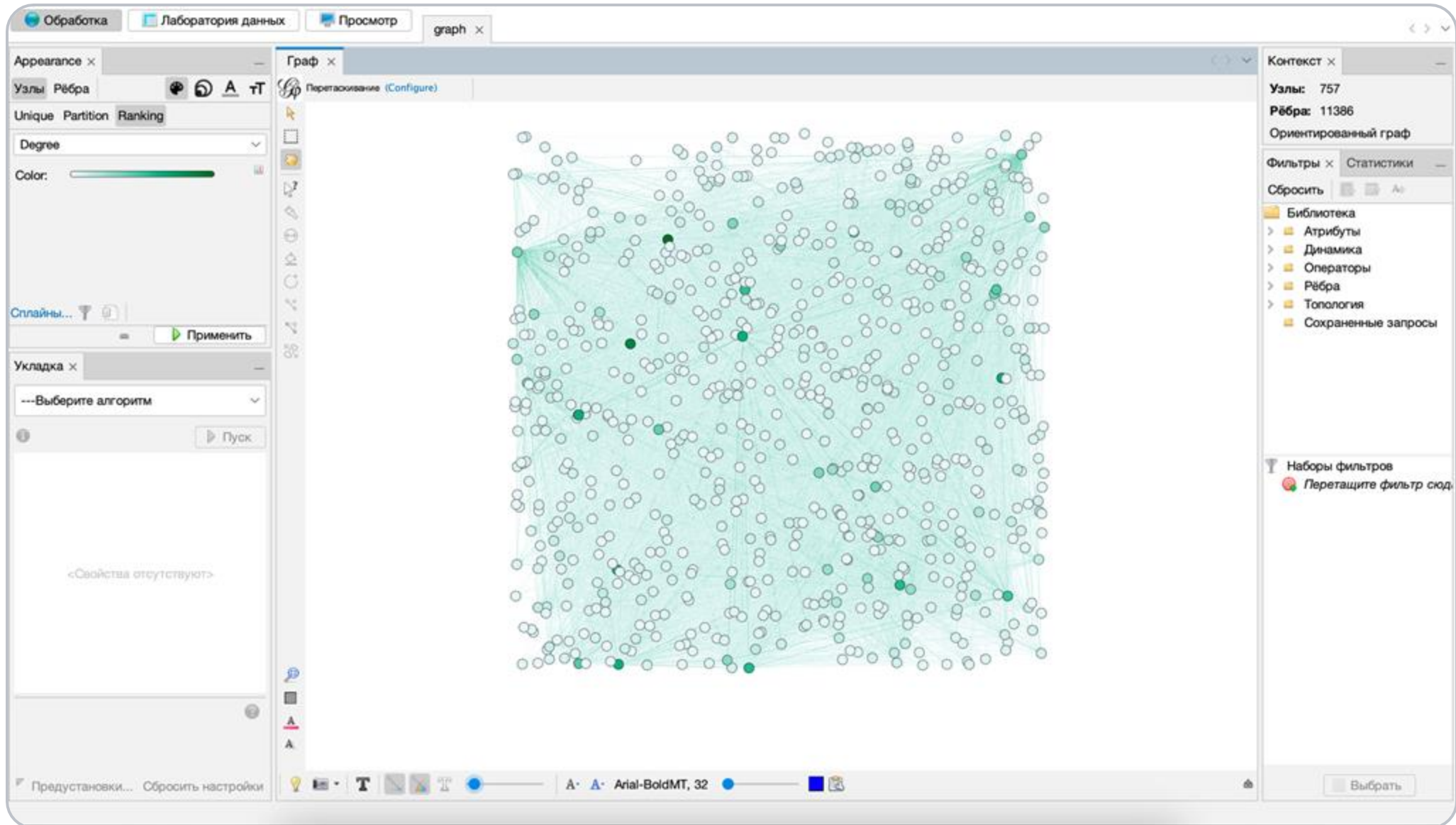
-  Перетасайте фильтр сюд.

Выбрать

Аrial-BoldMT, 32



The image shows a screenshot of the Gephi 0.10.1 software interface. The main window displays a network graph with 757 nodes and 11386 edges. The nodes are represented by small circles, and the edges are thin lines connecting them. The graph is densely connected, with many nodes having multiple neighbors. The interface is in Russian. On the left, there are panels for 'Appearance' (with 'Ranking' selected and 'Degree' as the ranking attribute) and 'Layout' (with a dropdown menu for selecting an algorithm). On the right, there is a 'Context' panel showing graph statistics and a 'Filters' panel with a library of filter types. The bottom of the window shows a toolbar with various icons and a font selection area.



Gephi 0.10.1 - Проект 3

Обработка | Лаборатория данных | Просмотр | graph x

Ареаланс x

Узлы Рёбра

Unique Partition Ranking

Degree

Color:

Слайды...

Применить

Укладка x

---Выберите алгоритм

Пуск

<Свойства отсутствуют>

Предустановки... Сбросить настройки

Граф x

Перетаскивание (Configure)

Контекст x

Узлы: 757
Рёбра: 11386

Ориентированный граф

Фильтры x | Статистики

Сбросить

Библиотека

- Атрибуты
- Динамика
- Операторы
- Рёбра
- Топология
- Сохранённые запросы

Наборы фильтров

Перетаскийте фильтр сюда

Выбрать

А Arial-BoldMT, 32

Применить

Укладка ×

Force Atlas

Пуск

Force Atlas	
Инерция	0.1
Сила отталкивания	200.0
Сила притяжения	10.0
Максимальное отдал	10.0
Автостабилизация	<input checked="" type="checkbox"/>
Сила автостаблизац	80.0
Чувствительность ав	0.2
Гравитация	30.0
Ослабление хабов	<input type="checkbox"/>
Учитывать размер	<input type="checkbox"/>
Скорость	1.0

Force Atlas

Предустановки... Сбросить настройки

Find

Обработка | Лаборатория данных | Просмотр | graph x

Граф x

Выделение мышью (Configure)

Узлы Рёбра

Appearance x

Unique Partition Ranking

Degree

Color:

Слайды... @

Применить

Укладка x

Force Atlas

Пуск

Force Atlas

Инерция	0.1
Сила отталкивания	5000.0
Сила притяжения	10.0
Максимальное оттал	10.0
Автостабилизация	<input checked="" type="checkbox"/>
Сила автостабилиз	80.0
Чувствительность ав	0.2
Гравитация	30.0
Ослабление хабов	<input type="checkbox"/>
Учитывать размер	<input type="checkbox"/>
Скорость	1.0

Force Atlas

Предустановки... Сбросить настройки

Контекст x

Узлы: 678
Рёбра: 11195
Ориентированный граф

Фильтры x Статистики

Сбросить @

Библиотека

- > Атрибуты
- > Динамика
- > Операторы
- > Рёбра
- > Топология
- Сохранённые запросы

Наборы фильтров

⚙️ Перетащите фильтр сюда.

Выбрать

А Arial-BoldMT, 32

Gephi 0.10.1 - Проект 3

Обработка | Лаборатория данных | Просмотр | graph x

Аппеаранс x

Узлы | Рёбра

Unique | Partition | Ranking

Degree

Color:

Слайды... | Применить

Укладка x

Force Atlas

Пуск

Force Atlas	
Инерция	0.1
Сила отталкивания	1000.0
Сила притяжения	200.0
Максимальное оттал	10.0
Автостабилизация	<input checked="" type="checkbox"/>
Сила автостабилизация	80.0
Чувствительность ав	0.2
Гравитация	30.0
Ослабление хабов	<input checked="" type="checkbox"/>
Учитывать размер	<input checked="" type="checkbox"/>
Скорость	1.0

Force Atlas

Предустановки... | Обросить настройки

Граф x

Выделение мышью (Configure)

Контекст x

Узлы: 649
Рёбра: 11111
Ориентированный граф

Фильтры x | Статистики

Сбросить

Библиотека

- Атрибуты
- Динамика
- Операторы
- Рёбра
- Топология
- Сохранённые запросы

Наборы фильтров

Перетяните фильтр сюда

Выбрать

А · Arial-BoldMT, 32

Gephi 0.10.1 - Проект 3

Обработка | Лаборатория данных | Просмотр | graph x

Аппаранс x

Узлы | Рёбра

Unique | Partition | Ranking

Degree

Color: [Color bar]

Слайды... [Icon] [Icon]

Применить

Укладка x

Укладка меток

Пуск

LabelAdjust

Скорость 1.0

Учитывать размер y:

Укладка меток

Предустановки... | Сбросить настройки

Граф x

Выделение мышью (Configure)

Контекст x

Узлы: 649

Рёбра: 11111

Ориентированный граф

Фильтры x | Статистики

Сбросить [Icon] [Icon] [Icon]

Библиотека

- > Атрибуты
- > Динамика
- > Операторы
- > Рёбра
- > Топология
- Сохранённые запросы

Наборы фильтров

Перетащите фильтр сюда

Выбрать

Gephi 0.10.1 - Проект 3

Обработка | Лаборатория данных | Просмотр | graph x | Copy of graph x

Аппеаранс x

Узлы Рёбра

Unique Ranking

Degree

Min size: 10 Max size: 60

Слайды... | Применить

Укладка x

Укладка меток

Пуск

LabelAdjust

Скорость 1.0

Учитывать размер y:

Укладка меток

Предустановки... Сбросить настройки

Граф x

Выделение мышью (Configure)

Контекст x

Узлы: 649

Рёбра: 11111

Ориентированный граф

Фильтры x | Статистики

Сбросить

Библиотека

- > Атрибуты
- > Динамика
- > Операторы
- > Рёбра
- > Топология
- Сохранённые запросы

Наборы фильтров

- Перетяните фильтр сюда.

Выбрать

Gephi 0.10.1 - Проект 3

Обработка Лаборатория данных Просмотр graph × Copy of graph ×

Таблица Данных ×

Узлы Ребра **Конфигурация** + Добавить узел + Добавить ребро Поиск/Замена Импортировать из CSV Экспортировать таблицу Другие действия Фильтр: Source

Source	Target	Type	Id	Label	Interval	Weight
ActionCards	DynamicSettingsCore	Ориентированное	23209			1.0
ActionCards	ResumeServices	Ориентированное	23210			1.0
ActionCards	ExperimentsCore	Ориентированное	23212			1.0
ActionCards	DesignSystem	Ориентированное	23213			1.0
ActionCards	FloatingCardCore	Ориентированное	23214			1.0
ActionCards	AVFoundation	Ориентированное	23215			1.0
ActionCards	SharedServicesCore	Ориентированное	23216			1.0
ActionCards	ApplicantAnalyticsEvents	Ориентированное	23217			1.0
ActionCards	ApplicantProfileCore	Ориентированное	23218			1.0
ActionCards	Nivelir	Ориентированное	23219			1.0
ActionCards	State	Ориентированное	23220			1.0
ActionCards	ApplicantResumeListCore	Ориентированное	23221			1.0
ActionCards	ApplicantStorageCore	Ориентированное	23222			1.0
ActionCards	WebCore	Ориентированное	23223			1.0
ActionCards	NegotiationsCore	Ориентированное	23224			1.0
ActionCards	CoreUser	Ориентированное	23225			1.0
ActionCards	CallServicesCore	Ориентированное	23226			1.0
ActionCards	CallsCore	Ориентированное	23227			1.0
ActionCards	EmployerReviewsCore	Ориентированное	23228			1.0
ActionCards	ActionCardsCore	Ориентированное	23229			1.0
ActionCards	Core	Ориентированное	23230			1.0
ActionCards	Theme	Ориентированное	23231			1.0
ActionCards	ApplicantNotifications	Ориентированное	23232			1.0
ActionCards	Perception	Ориентированное	23233			1.0
ActionCards	VacancySearchCore	Ориентированное	23234			1.0
ActionCards	CollectionView	Ориентированное	23235			1.0
ActionCards	ApplicantHomeCore	Ориентированное	23236			1.0
ActionCards	ResumeProfileBuilderCore	Ориентированное	23237			1.0
ActionCards	Module	Ориентированное	23238			1.0
ActionCards	APIServices	Ориентированное	23239			1.0
ActionCards	Routing	Ориентированное	23240			1.0

Добавить столбец
 Объединить столбцы
 Удалить столбец
 Очистить значения в столбце
 Скопировать значения в другой столбец
 Заполнить столбец значением
 Создать копию колонки
 Создание столбца с логическими значениями с помощью регулярного выражения
 Создание столбца с помощью поиска по маске, заданной регулярным выражением

Гephi 0.10.1 - Проект 3

Обработка | Лаборатория данных | Просмотр | graph x | Copy of graph x

Настройки предпросмотра x | Предпросмотр x

Наборы настроек

По умолчанию

Управление отрисовщиками

Настройки

Узлы

Fixed Border Width	<input checked="" type="checkbox"/>
Толщина обводки	1.0
Цвет обводки	darker
Прозрачность	100.0
Per-Node Opacity	<input type="checkbox"/>

Метки узлов

Отображать метки	<input checked="" type="checkbox"/>
Шрифт	Arial 12 Простой тек
Пропорциональный р	<input type="checkbox"/>
Цвет	custom [0,0,0]
Укорачивать метки	<input type="checkbox"/>
Максимальное число	30
Толщина обводки	0.0
Цвет обводки	custom [255,255,2..
Прозрачность обводк	80.0
Отображать рамки м	<input type="checkbox"/>
Цвет рамки меток	parent
Прозрачность рамки	100.0

Рёбра

Отображать рёбра	<input checked="" type="checkbox"/>
Толщина	1.0
Масштабировать вес	<input type="checkbox"/>
Min. rescaled weight	0.1
Max. rescaled weight	1.0
Цвет	mixed
Прозрачность	100.0

Детальность предпрос...

Экспорт: SVG/PDF/PNG

Обновит

Цвет фона Сбросить масштаб - +

Получившаяся последовательность миграции как артефакт

```
1 #1,DictionaryCoder,Levitan
2 #2,Core,Theme
3 #3,State
4 #4,HRMobileHHRichPush,HRMobileJTBRichPush,Networking,UI
5 #5,DesignSystem,NetworkingTesting
6 #6,APIServices,AlertCore,ApplicantResumeListCore,ArtifactSelectionCore,DateTimePickerCore,ExperimentsCore,FloatingCardCore,HHRProCore,Module,PhoneRegionPickerCore,ResumeUI,SharedModels,TextAreaCore,TextInputCo
7 #7,APIServicesTesting,ApplicantProfileCore,Authorization,DreamJobCore,EmployerAuthorizationCore,JobSearchStatusCore,Maps,MovieCaptureCore,PaidServicesCore,ResumeExpertServicesCore,SharedServicesCore,SuggestCo
8 #8,APIDictionaries,AnalyticsServices,ApplicantExpertMarketplaceCore,CallServicesCore,DynamicSettings,Experiments,Routing,SharedComponents,SharedServices,WrongUserRole
9 #9,APIDictionariesTesting,AboutUs,ActionList,AddressMap,Alert,AnalyticsServicesTesting,ApplicantEmployerCore,AreaServices,ArtifactSelection,CallServices,ChatServices,DateTimePicker,EmployerAddressServices,Emp
10 #10,ApplicantServices,ApplicationFeedback,Calls,Captcha,DreamJob,EmployerAPIModels,EmployerAddressList,EmployerAddressPicker,EmployerAddressUI,EmployerProfileCore,EmployerUI,KeySkillsServices,Markdown,Notific
11 #11,ActionCardsCore,ApplicantAuthorizationCore,ApplicantCareerCore,ApplicantStorage,Articles,CatalogSelection,ChatDesignSystem,EditAdditionalLanguages,EmployerAPIModelsTesting,EmployerAddress,EmployerAutosear
12 #12,AccountSettings,ApplicantCallTracking,ApplicantPhoneVerification,ApplicantServicesTesting,ApplicantVacancyCore,ChatsUI,DebugMenu,EmployerHiringManager,EmployerNegotiationsCore,EmployerProfile,EmployerVaca
13 #13,ApplicantAutosearch,ApplicantBlacklist,ApplicantChats,ApplicantFavorites,ApplicantResumeViews,ApplicantVacancy,Chats,EmployerApplicantComplaint,EmployerAuthorization,EmployerHiringManagerTesting,EmployerH
14 #14,ActionCards,ApplicantCareer,ApplicantOnboardingCore,ApplicantResume,AutoResponse,EmployerResumeUI,KeySkillsCore,NegotiationActionsCore,Negotiations,PaidServices,ResumeEditPositionSelectionCore,ResumeMerge
15 #15,ApplicantAuthorization,ApplicantExpertMarketplace,ApplicantHome,ApplicantLaunch,ApplicantOnboarding,ApplicantProfile,ApplicantResumeList,EmployerChats,EmployerMassActions,EmployerResumeServices,EmployerRe
16 #16,ApplicantEmployer,EmployerAutosearch,EmployerCallTracking,EmployerFavorites,EmployerFeedback,EmployerNegotiations,EmployerPayment,EmployerResume,EmployerResumeContacts,EmployerRichPush,EmployerVacancy,Res
17 #17,HRMobileHH,HRMobileJTB,ResumeExpertServices,ResumeSearchTesting
18 #18,ApplicantHH,ApplicantJTB,ApplicantZP
19 Суммарное кол-во таргетов для миграции: 226
```

Последовательность миграции: 18 итераций

Параллелизация

Модули в одной итерации
не зависят друг от друга

Последовательность

Переход к следующей итерации
только после завершения текущей

Оптимизация генерации файлов

Изменения таргетов

Проект может меняться: добавление, удаление

Скрипт автоматизации

Свой скрипт для очистки и регенерации файлов

Экономия времени

Уменьшение трудозатрат на рутинные операции

```
mint run kowalski analysis --root-path ./ --work-space HH.xcworkspace
```

```
🌱 Finding latest version of kowalski  
🌱 Running kowalski master...  
Analyzing migration path to Swift 6
```

```
Level 0:
```

```
Core  
Theme
```

```
Level 1:
```

```
DesignSystem  
ExperimentsCore  
MetroSelectCore  
State  
UI
```

```
Level 2:
```

```
ApplicantProfileCore  
ApplicantResumeListCore  
ArtifactSelectionCore  
CoreTesting  
DateTimePickerCore  
DesignSystemTesting  
FloatingCardCore  
Module  
Networking  
PhoneRegionPickerCore  
PresenceGoalsCore  
ResumeExpertServicesCore  
ResumeUI  
SwiftUICompatibility  
TextAreaCore  
TextInputCore
```

Итоги

Генерируем граф через Tuist

Минимизация рисков

Стратегия листьев снижает риски миграции

Изолированные изменения

Фокус на изоляции изменений в коде

Автоматизируем процессы

Стараемся автоматизировать рутинные вещи, такие как поиск путей графа

План миграции с точки зрения организации

- 1 Автоматизация рутинных процессов
- 2 Шаринг знаний
- 3 Построение коммуникации
- 4 Технический релиз

Проблемы Static var для хранения инстансов

```
5
6 import Foundation
7
8 public enum Logger {
9
10     // Можно, так как всего один раз настраиваем loggingService
11     nonisolated(unsafe) private static var loggingService: CoreLoggingService?
12
13     @MainActor
14     public static func setup(loggingService: CoreLoggingService) {
15         Self.loggingService = loggingService
16     }
17
18     private static func shouldLog(untilDate: Date?) -> Bool {
19         guard let untilDate else {
20             return true
21         }
22         return Date() < untilDate
23     }
24
```

UserDefault

```
5
6 import Foundation
7
8 final class HHUserDefaultsStorage {
9
10     private enum Constants {
11         static let arrayKeyPrefix = "array_"
12         static let arraySeparator = ";^4*"
13     }
14
15     private let suiteName: String?
16
17     private let decoder: CommonDecoder
18     private let encoder: CommonEncoder
19
20     private var userDefaults: UserDefaults {
21         if let suiteName {
22             UserDefaults(suiteName: suiteName)!
23         } else {
24             UserDefaults.standard
25         }
26     }
27
28     init(
29         suiteName: String? = nil,
30         decoder: CommonDecoder,
31         encoder: CommonEncoder
32     ) {
```

Sendable и Hashable надо подружить

```
struct AnyHashableAndSendable: @unchecked Sendable, Hashable {  
    private let wrapped: AnyHashable  
  
    init(wrapped: some Hashable & Sendable) {  
        self.wrapped = .init(wrapped)  
    }  
}
```

Модульная архитектура iOS-проекта

1

итерация

Фундаментальные, сложные
модули

53

таргета

Регрессы
технические релизы

Полезные ссылки

Tuist: Xcode-проект на стероидах

[читать](#)

iOS

[читать](#)

Как плести виртуальные сети
с помощью Gephi

[читать](#)

Методы укладки Gephi. Force Atlas
и Force Atlas 2

[читать](#)