Вадим Салаватов

Реализация мультиплатформенного доступа к файловым хранилищам на языке Kotlin

Выпускная квалификационная работа

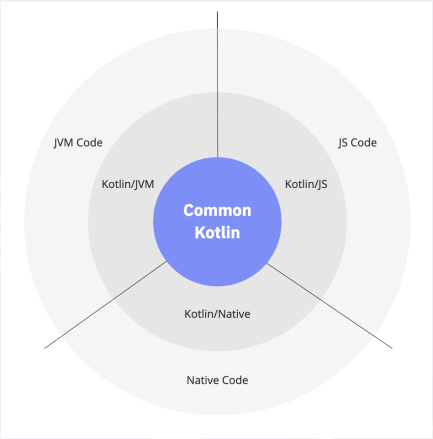
Научный руководитель: **(TODO:)** В. Н. Брагилевский

14 июня 2022

Факультет математики и компьютерных наук СПбГУ
Программа «Современное программирование»

Введение в предметную область

- Kotlin Multiplatform
- Файловые системы
- Хотим писать код в общем модуле
- Как работать с файловыми хранилищами в веб-браузере?



Обзор основных аналогов

- okio
  - \+ мультиплатформенность
  - \+ активно развивается
  - \+ `FileSystem`…
  - \- …но для JS требует Node.js

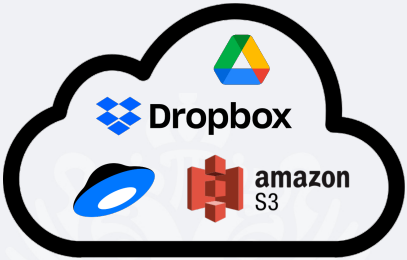
МКН

Обзор основных аналогов

- okio
  - \+ мультиплатформенность
  - \+ активно развивается
  - \+ `FileSystem`...
  - \- ...но для JS требует Node.js
- korlibs/korio
  - \+ мультиплатформенность
  - \+ интерфейс VFS, несколько реализаций
  - \- последнее обновление почти год назад
  - \- перегруженность VFS
  - ± для JS есть VFS на основе `localStorage`

Обзор основных аналогов

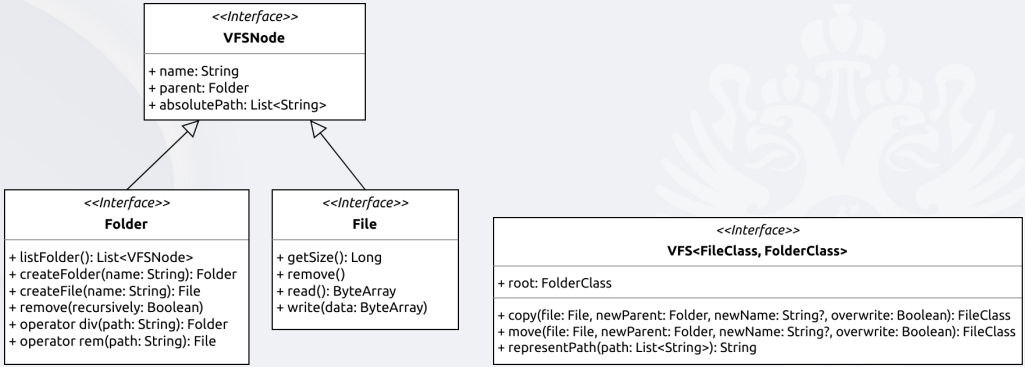
- okio
  - \+ мультиплатформенность
  - \+ активно развивается
  - \+ `FileSystem`...
  - \- ...но для JS требует Node.js
- korlibs/korio
  - \+ мультиплатформенность
  - \+ интерфейс VFS, несколько реализаций
  - \- последнее обновление почти год назад
  - \- перегруженность VFS
  - ± для JS есть VFS на основе `localStorage`



Постановка задачи

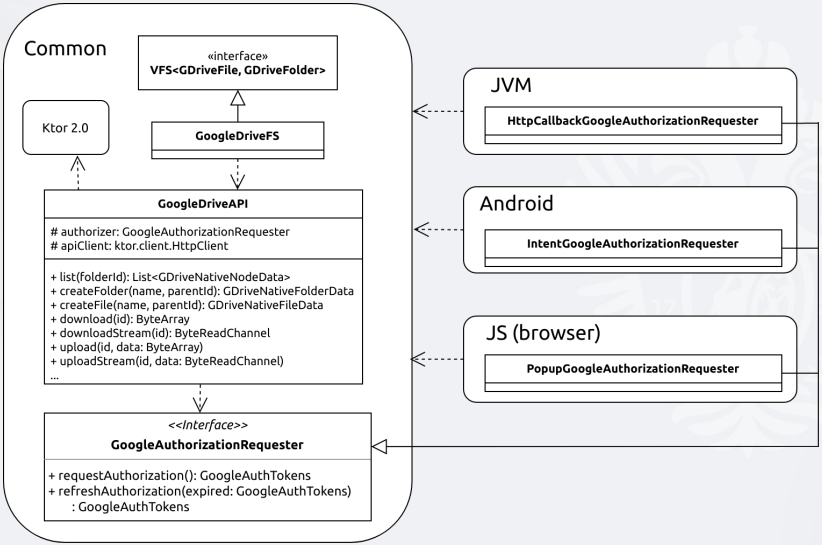
1. Разработать мультиплатформенную библиотеку для работы с файловыми хранилищами
2. Поддержать платформы JVM, Android, JS (browser)
3. Поддержать как минимум одно облачное хранилище
4. Обеспечить простую расширяемость как в плане поддержки новых хранилищ, так и в плане предоставляемой функциональности

Архитектура библиотеки



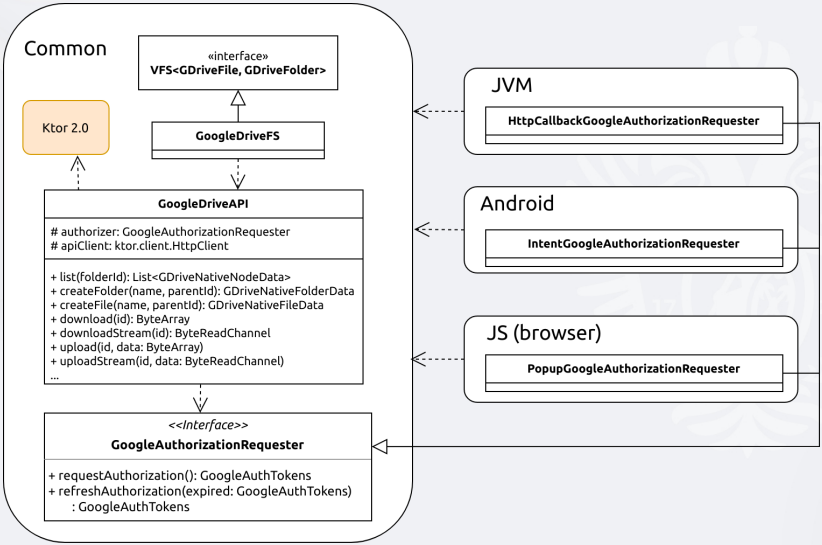
Все методы (кроме `representPath`) помечены `suspend`

GoogleDriveFS



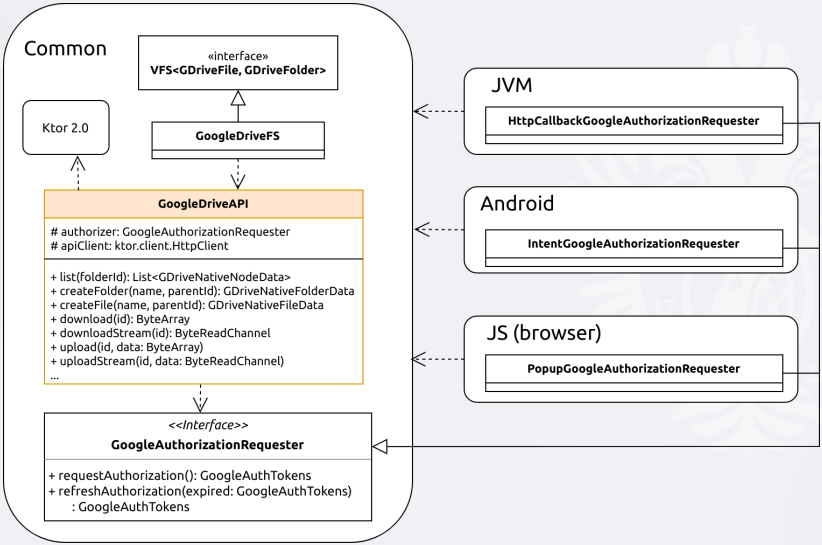
MKH

GoogleDriveFS



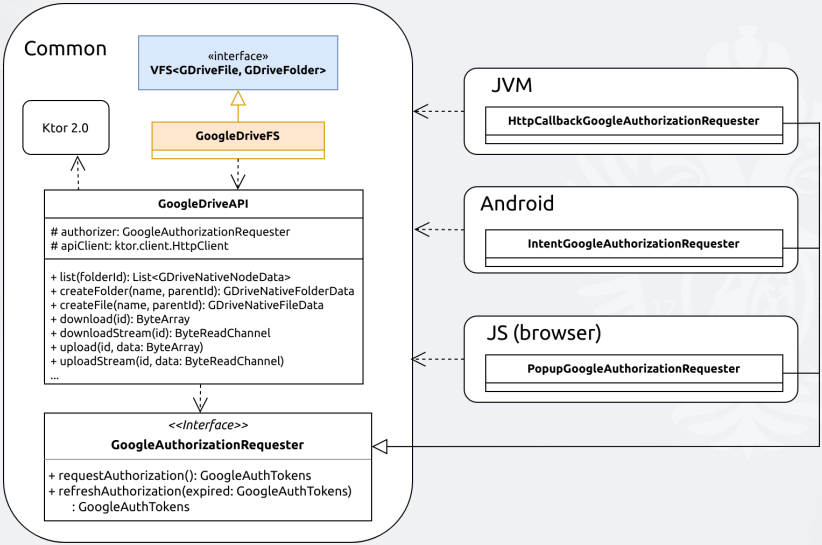
MKH

GoogleDriveFS



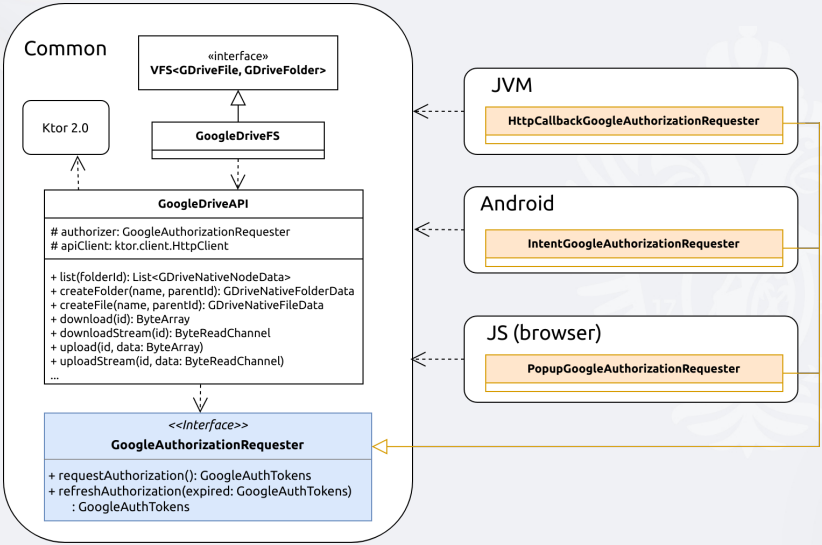
MKH

GoogleDriveFS



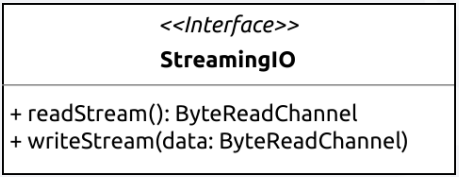
MKH

GoogleDriveFS



MKH

Расширения



SystemFS и GoogleDriveFS реализуют потоковое чтение и запись

МКН

Сводная таблица поддержанных возможностей

	Платформа			Расширения
Название	JVM	Android	JS (browser)	StreamingIO
SystemFS	да	да		да
GoogleDriveFS	да	да	да	да
SqliteFS		да		

МКН

Приложения на основе библиотеки

- Multieditor[1] — мультиплатформенный текстовый редактор, работающий на платформах JVM, Android, Web
- gdrive-cli[2] — интерфейс командной строки для работы с содержимым Google Drive, позволяет скачивать и загружать файлы в потоковом режиме

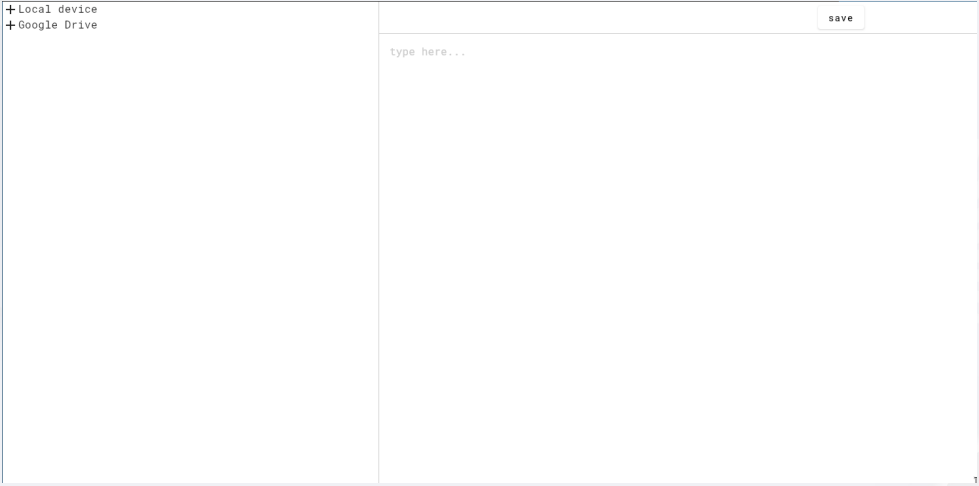
[1] https://github.com/vsalavatov/multieditor
[2] https://github.com/vsalavatov/gdrive-cli

МКН

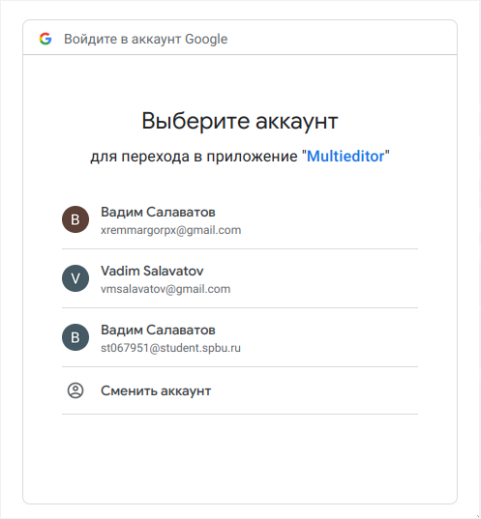
Multieditor

- Вся логика работы с файлами в общем модуле
- Код графических интерфейсов для JVM и Android написан с помощью Compose в общем для них модуле
- Код графического интерфейса для JS (browser) написан на Compose for Web

Multieditor JVM



Multieditor JVM



Multieditor JVM

Результаты

1. Реализована библиотека для мультиплатформенной работы с файловыми хранилищами
2. Поддержаны три хранилища, из которых одно облачное
3. На основе библиотеки реализовано мультиплатформенное приложение

---

vmsalavatov@gmail.com, @vsalavatov

https://github.com/vsalavatov/multifs

Дополнительные слайды

Пример. Общий код

```
typealias Storage = VFS<out File, out Folder>
suspend fun printFiles(folder: Folder) {
  for (node in folder.listFolder()) {
    when (node) {
      is File -> println("${node.name} at ${node.absolutePath}")
      is Folder -> printFiles(node)
    }
  }
}
suspend fun printStorages(storages: List<Storage>) {
  storages.forEach { printFiles(it.root) }
}
```

МКН

Пример. Клиентский код

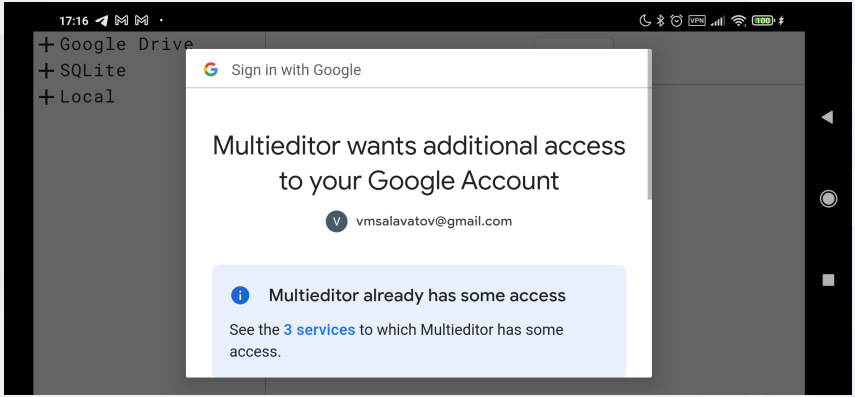
```
fun getStorages(): List<Storage> {
  val systemfs = SystemFS()
  val googleAuth = HttpCallbackGoogleAuthorizationRequester(
                     GoogleAppCredentials(CLIENT_ID, SECRET),
                     GoogleDriveAPI.Companion.DriveScope.General
                   )
  val gdrivefs = GoogleDriveFS(GoogleDriveAPI(googleAuth))
  return listOf(systemfs, gdrivefs)
}
suspend fun main() {
  printStorages(getStorages())
}
```

Пример. Расширения

```
suspend fun <T> upload(fs: VFS<out T, out Folder>)
        where T : File, T : StreamingIO {
  val file = (fs.root / "folder" / "subfolder" % "file.txt") as T
  val data = ByteReadChannel(
    ByteArray(10_000_000) { it.toByte() }
  )
  file.writeStream(data)
}
...
upload(gdrivefs)
```

МКН

Multieditor Android



MKH

Multieditor Android

Multieditor Web