

avito.tech

# **“Go, please”: language server ПОД МИКРОСКОПОМ**

**GopherCon Russia 2020**  
**09.08.2020**



Илья Данилкин  
@nezorflame

# Содержание

Language Server Protocol	01
LSP и Go	02
gopls	03
GoLand vs VSCode	04
Демонстрация работы	05
Итоги	06

Секция 01

# Language Server Protocol

# Language Server Protocol

# Language Server Protocol

- **открытый** протокол

# Language Server Protocol

- **открытый** протокол
- основан на JSON-RPC

# Language Server Protocol

- **открытый** протокол
- основан на JSON-RPC
- **клиент** - редактор кода / IDE

# Language Server Protocol

- **открытый** протокол
- основан на JSON-RPC
- **клиент** - редактор кода / IDE
- **сервер** - language server



# Language Server Protocol

- **открытый** протокол
- основан на JSON-RPC
- **клиент** - редактор кода / IDE
- **сервер** - language server
- фокус на фичах редактора вместо языка

# Language Server Protocol

- **открытый** протокол
- основан на JSON-RPC
- **клиент** - редактор кода / IDE
- **сервер** - language server
- фокус на фичах редактора вместо языка
- много языков - одна среда разработки

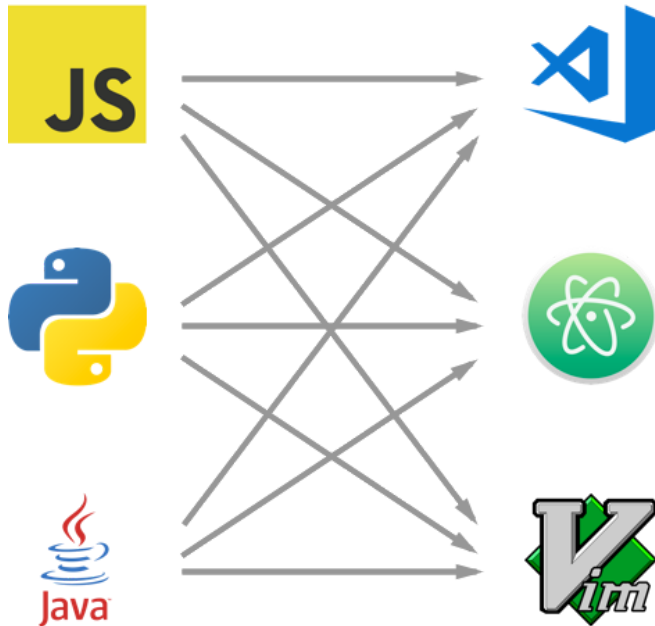
# Language Server Protocol

- **открытый** протокол
- основан на JSON-RPC
- **клиент** - редактор кода / IDE
- **сервер** - language server
- фокус на фичах редактора вместо языка
- много языков - одна среда разработки

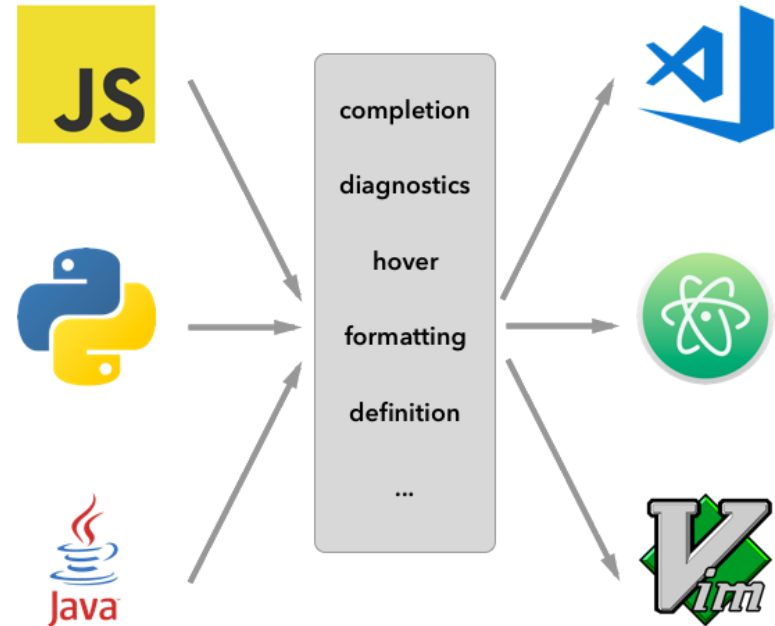


# Language Server Protocol

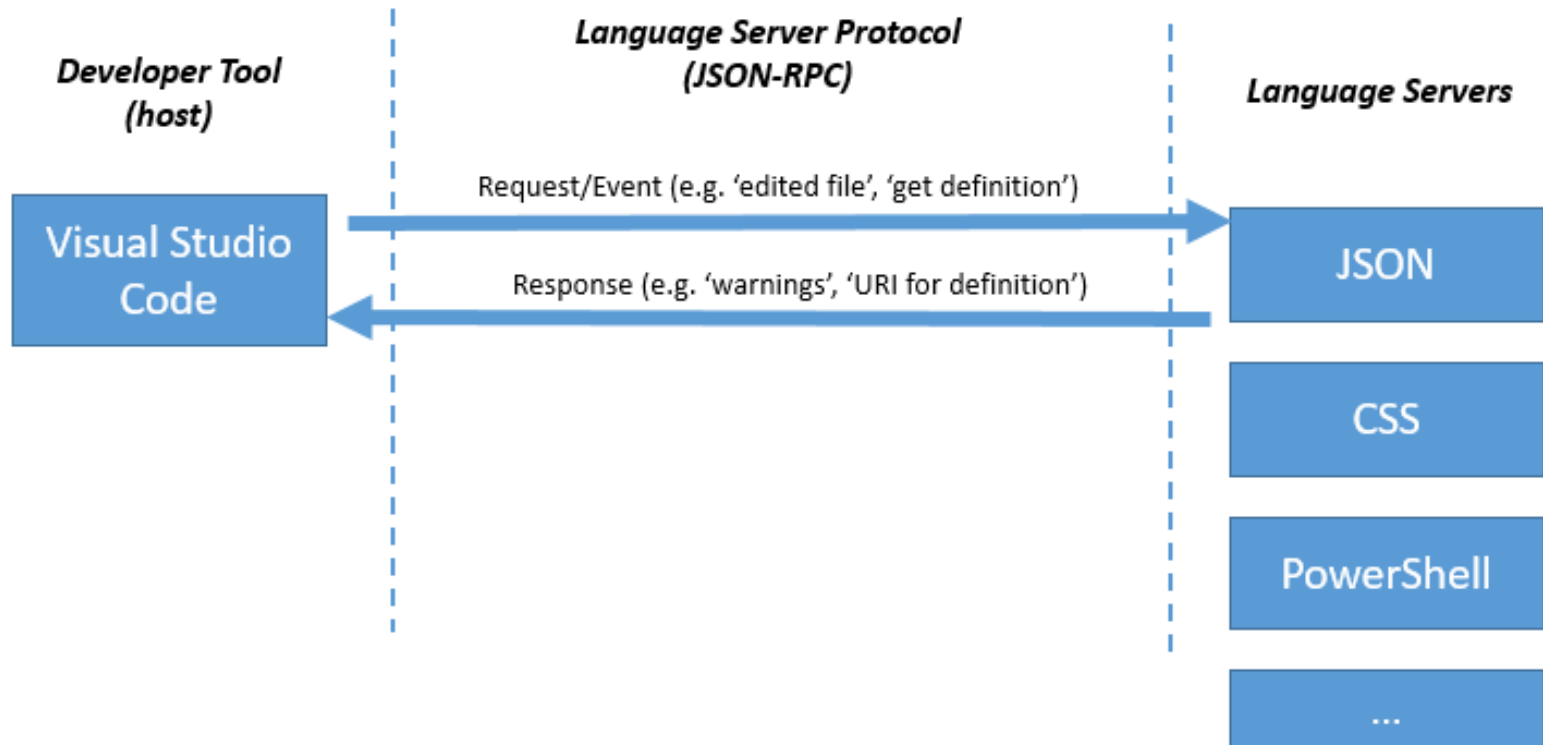
NO LSP



LSP



# Language Server Protocol



# LSIF

# LSIF

Language **S**erver **I**ndex **F**ormat

# LSIF

Language **S**erver **I**ndex **F**ormat

- надстройка над LSP



# LSIF

## Language **S**erver **I**ndex **F**ormat

- надстройка над LSP
- отображение информации без checkout'a

# LSIF

## Language **S**erver **I**ndex **F**ormat

- надстройка над LSP
- отображение информации без checkout'a
- уже есть для нескольких языков

# LSIF

## Language **S**erver **I**ndex **F**ormat

- надстройка над LSP
- отображение информации без checkout'a
- уже есть для нескольких языков

Подробности в [блоге](#) и на

<https://lsif.dev/>

# Language Server

# Language Server

- имплементация фич языка

# Language Server

- имплементация фич языка
- объединение массы утилит в одной

# Language Server

- имплементация фич языка
- объединение массы утилит в одной
- кэширование

# Language Server

- имплементация фич языка
- объединение массы утилит в одной
- кэширование
- нет зависимости от IDE



# Language Server

- имплементация фич языка
- объединение массы утилит в одной
- кэширование
- нет зависимости от IDE

LSP-протокол - интерфейс

Language Server - его реализация

Секция 02

# LSP и Go

# Sourcegraph

# Sourcegraph

- платформа для разработчиков

# Sourcegraph

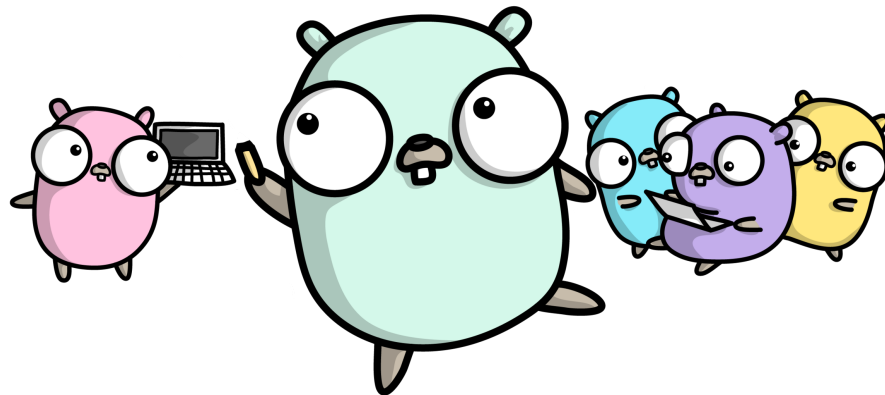
- платформа для разработчиков
- навигация, анализ, ревью кода и алерты

# Sourcegraph

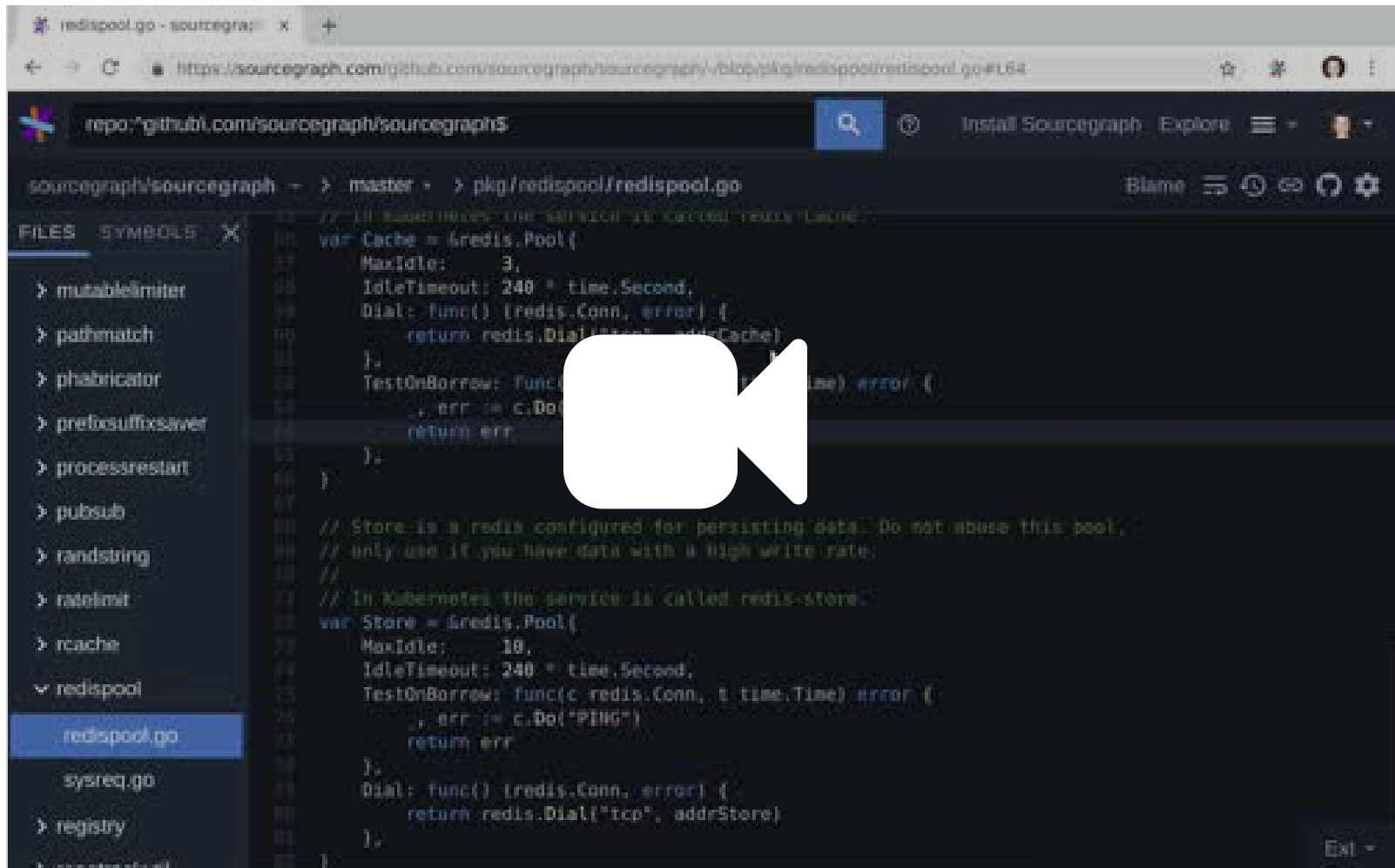
- платформа для разработчиков
- навигация, анализ, ревью кода и алерты
- под капотом - language server

# Sourcegraph

- платформа для разработчиков
- навигация, анализ, ревью кода и алерты
- под капотом - language server
- первый LSP для Go



# ...и все было хорошо...



The screenshot shows a web browser displaying the Sourcegraph interface for a repository named 'sourcegraph/sourcegraph'. The URL in the address bar is 'https://sourcegraph.com/github.com/sourcegraph/sourcegraph/-/blob/pkg/redispool/redispool.go#L64'. The file 'redispool.go' is selected in the left sidebar. The code in the main editor area is as follows:

```
11 // In Kubernetes the service is called redis-cache.
12
13 var Cache = &redis.Pool{
14     MaxIdle:     3,
15     IdleTimeout: 240 * time.Second,
16     Dial: func() (redis.Conn, error) {
17         return redis.Dial("tcp", addrCache)
18     },
19     TestOnBorrow: func(c redis.Conn, t time.Time) error {
20         _, err := c.Do("PING")
21         return err
22     },
23 }
24
25 // Store is a redis configured for persisting data. Do not abuse this pool,
26 // only use if you have data with a high write rate.
27 //
28 // In Kubernetes the service is called redis-store.
29
30 var Store = &redis.Pool{
31     MaxIdle:     10,
32     IdleTimeout: 240 * time.Second,
33     TestOnBorrow: func(c redis.Conn, t time.Time) error {
34         _, err := c.Do("PING")
35         return err
36     },
37     Dial: func() (redis.Conn, error) {
38         return redis.Dial("tcp", addrStore)
39     },
40 }
```



# ...до появления Go modules

Robby O'Connor on 3 Sep 2018



Things need to be updated to support go 1.11, currently, things don't work correctly in go 1.11.



13



tamasfe mentioned this issue on 5 Sep 2018

**Go 1.11 imports of local packages in a module aren't recognized #1909**



**go-langserver + modules failing to complete with "expected selector or type assertion, found 'return'"**

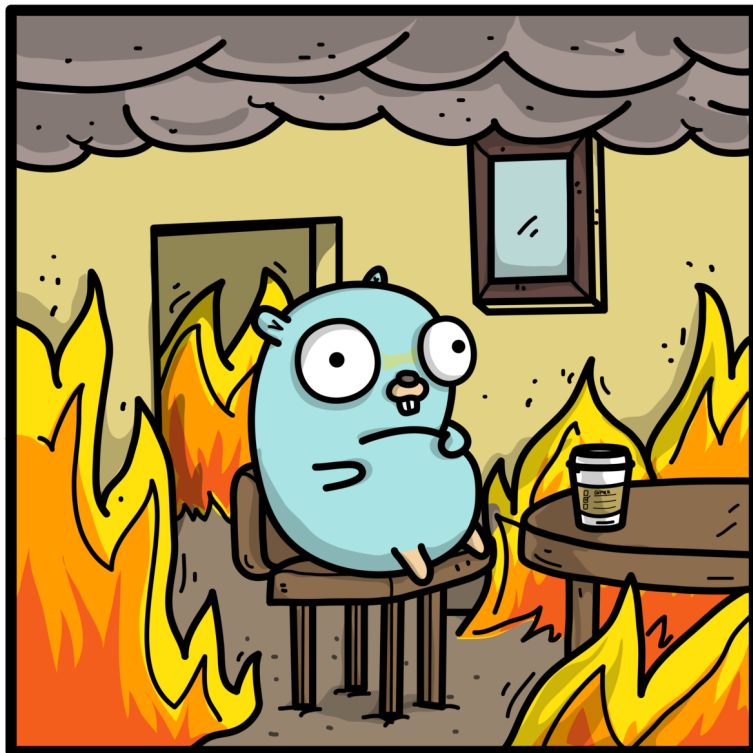
#2053 by alethomas was closed on 31 Oct 2018



**go-langserver using an insane amount of cpu**

#1994 by primalmotion was closed on 26 Aug 2019

needs more info



# bingo

# bingo

- fork **sourcegraph-go**

# bingo

- форк **sourcegraph-go**
- поддержка модулей

# bingo

- форк **sourcegraph-go**
- поддержка модулей
- портированы фичи из **guru**:  
rename, go to implementation, etc.

Секция 03

**gopls**

**Go, please!**



# Go, please!

- официальный LSP от Go core team

# Go, please!

- официальный LSP от Go core team
- ЖИВЕТ В [golang.org/x/tools](https://golang.org/x/tools)

# Go, please!

- официальный LSP от Go core team
- ЖИВЕТ В [golang.org/x/tools](https://golang.org/x/tools)
- "one tool to rule them all"

# Go, please!

- официальный LSP от Go core team
- ЖИВЕТ В [golang.org/x/tools](https://golang.org/x/tools)
- "one tool to rule them all"
- поддержка фич языка **до** релиза нового функционала (привет, модули!)

# Go, please!

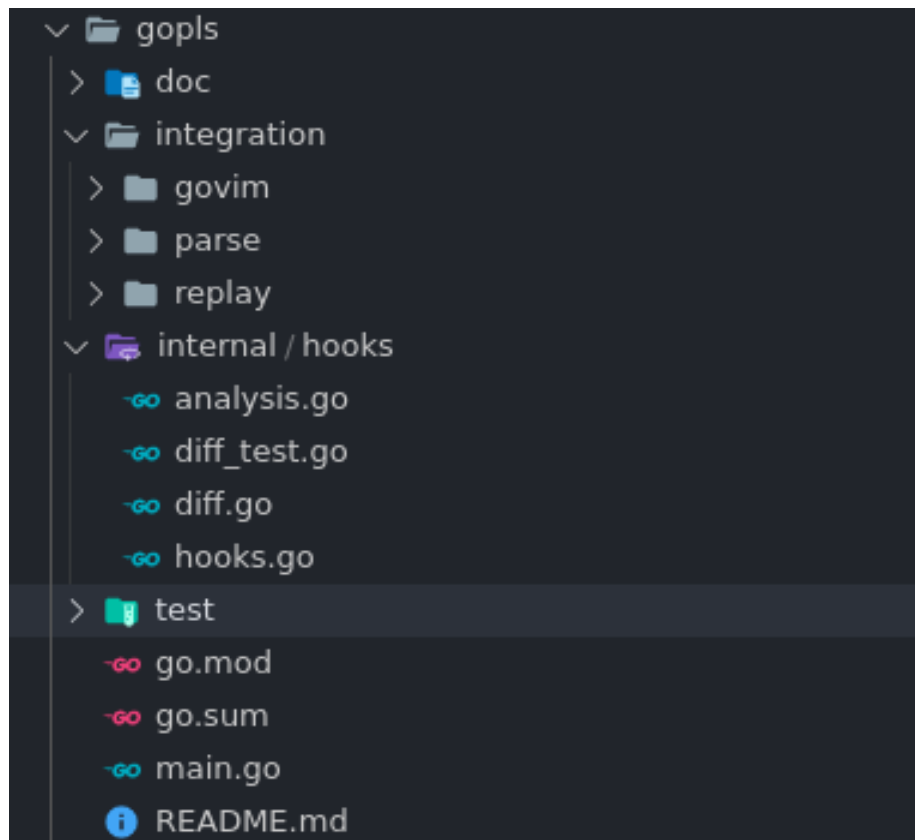
- официальный LSP от Go core team
- ЖИВЕТ В [golang.org/x/tools](https://golang.org/x/tools)
- "one tool to rule them all"
- поддержка фич языка **до** релиза нового функционала (привет, модули!)

GopherCon Russia 2019:

[Go, pls stop breaking my editor](#)

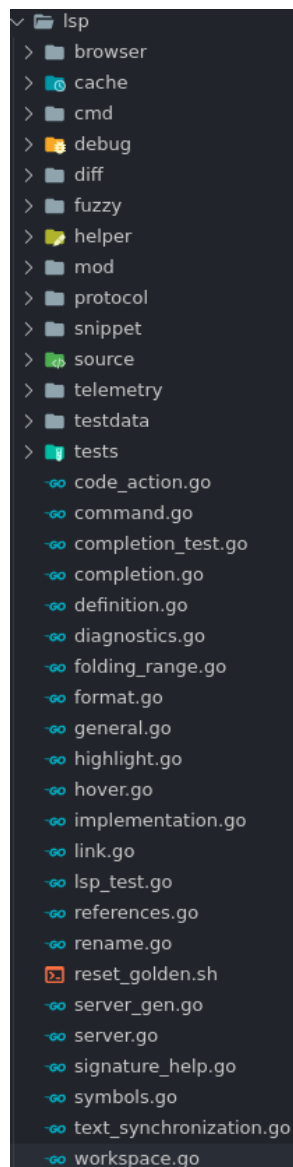
# gopls

Небольшой  
бинарник...



# gopls

...и большая  
библиотека



A screenshot of a code editor showing the file tree for the gopls project. The tree is expanded to show the 'tests' directory, which contains a list of Go source files and one shell script. The files are listed with their full names, including the '.go' extension. The shell script 'reset\_golden.sh' is highlighted with a blue background.

- lsp
  - browser
  - cache
  - cmd
  - debug
  - diff
  - fuzzy
  - helper
  - mod
  - protocol
  - snippet
  - source
  - telemetry
  - testdata
  - tests
    - code\_action.go
    - command.go
    - completion\_test.go
    - completion.go
    - definition.go
    - diagnostics.go
    - folding\_range.go
    - format.go
    - general.go
    - highlight.go
    - hover.go
    - implementation.go
    - link.go
    - lsp\_test.go
    - references.go
    - rename.go
    - reset\_golden.sh
    - server\_gen.go
    - server.go
    - signature\_help.go
    - symbols.go
    - text\_synchronization.go
    - workspace.go

# gopls

## Пакет

gopls

internal/lsp

internal/lsp/cache

internal/lsp/cmd

internal/lsp/debug

internal/lsp/protocol

internal/lsp/source

internal/span

internal/memoize

internal/jsonrpc2

## Описание

исполняемый файл, плагины и тесты

основной пакет для обработки сообщений

кэш

обработка командной строки

вспомогательные функции для дебага

обработка функций протокола

реализация фич сервера

управление адресацией файлов в пакете

кэш вызова функций

реализация JSON RPC2



# Структура

**Cache → Session → View**

# Структура

Cache ➡ Session ➡ View

- **Кэш:** глобальная информация  
(файловая система, ее содержимое и т.п.)

# Структура

Cache ➡ Session ➡ View

- **Кэш:** глобальная информация  
(файловая система, ее содержимое и т.п.)
- **Сессия:** подключение к редактору / IDE  
(открытые файлы / overlays)

# Структура

Cache ➡ Session ➡ View

- **Кэш:** глобальная информация  
(файловая система, ее содержимое и т.п.)
- **Сессия:** подключение к редактору / IDE  
(открытые файлы / overlays)
- **Отображение:** конфигурация, маппинг  
на пакеты

# Feature list

- Автокомплит
- Переход к определению / имплементации
- Документация по наведению
- Ссылки
- Рефакторинг (WIP)

Текущий [статус](#), [v1.0 milestone](#) (61 issue)

# Поддержка редакторов

- VSCode
- Vim / Neovim
- Emacs
- Acme
- Sublime Text
- Atom

Секция 04

**GoLand**

**VS**

**VSCode**

# GoLand



# GoLand

- полноценная "классическая" IDE

# GoLand

- полноценная "классическая" IDE
- родилась из IDEA + плагин для Go

# GoLand

- полноценная "классическая" IDE
- родилась из IDEA + плагин для Go
- проприетарная

# GoLand

- полноценная "классическая" IDE
- родилась из IDEA + плагин для Go
- проприетарная
- Java

# GoLand

- полноценная "классическая" IDE
- родилась из IDEA + плагин для Go
- проприетарная
- Java



# Возможности

# Возможности

- полная совместимость с модулями

# Возможности

- полная совместимость с модулями
- продвинутый рефакторинг



# Возможности

- полная совместимость с модулями
- продвинутый рефакторинг
- поддержка всех фич дебаггера

# Возможности

- полная совместимость с модулями
- продвинутый рефакторинг
- поддержка всех фич дебаггера
- кэш всего GORATH

# Возможности

- полная совместимость с модулями
- продвинутый рефакторинг
- поддержка всех фич дебаггера
- кэш всего GOPATH
- полноценный SQL клиент

# Возможности

- полная совместимость с модулями
- продвинутый рефакторинг
- поддержка всех фич дебаггера
- кэш всего GOPATH
- полноценный SQL клиент
- HTTP-клиент

# Возможности

- полная совместимость с модулями
- продвинутый рефакторинг
- поддержка всех фич дебаггера
- кэш всего GOPATH
- полноценный SQL клиент
- HTTP-клиент
- расширяемость

# VSCode

# VSCode

- **открытый** текстовый редактор

# VSCode

- **открытый** текстовый редактор
- быстрый



# VSCode

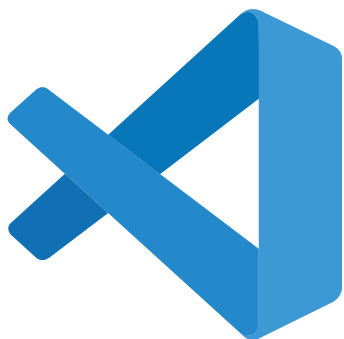
- [открытый](#) текстовый редактор
- быстрый
- Go: [vscode-go](#) + gopls

# VSCode

- **открытый** текстовый редактор
- быстрый
- Go: **vscode-go** + gopls
- Electron

# VSCode

- [открытый](#) текстовый редактор
- быстрый
- Go: [vscode-go](#) + gopls
- Electron



# Возможности

# Возможности

- расширяемость

# Возможности

- расширяемость
- удаленная работа (тонкий клиент + сервер)

# Возможности

- расширяемость
- удаленная работа (тонкий клиент + сервер)
- глобальные и локальные настройки

# Возможности

- расширяемость
- удаленная работа (тонкий клиент + сервер)
- глобальные и локальные настройки
- IntelliSense (расширенный автокомплит)



# Возможности

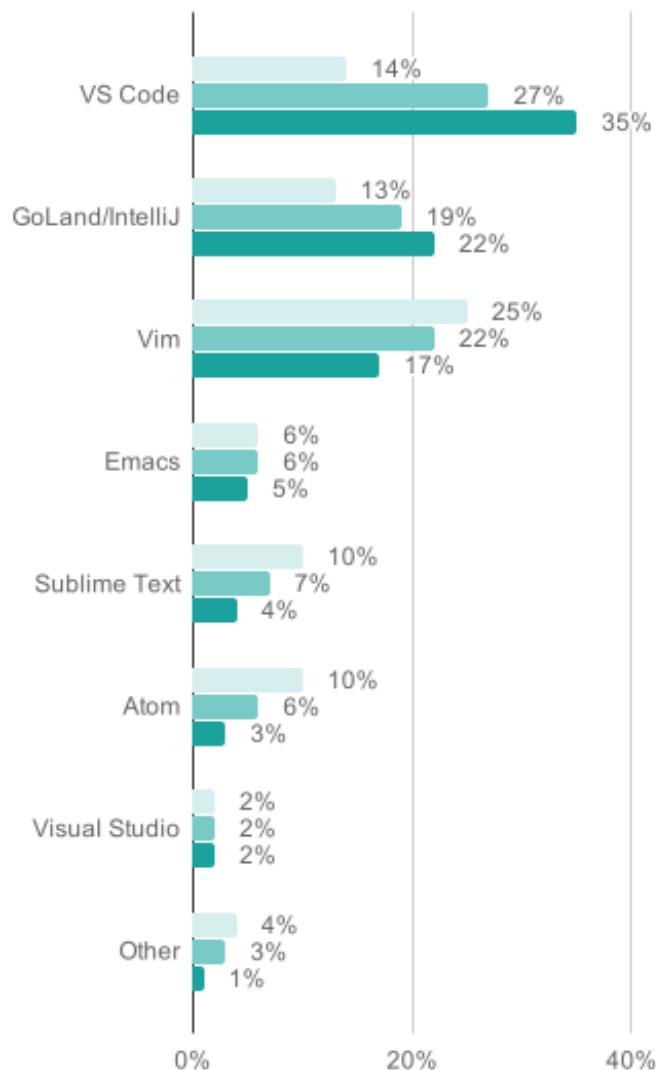
- расширяемость
- удаленная работа (тонкий клиент + сервер)
- глобальные и локальные настройки
- IntelliSense (расширенный автокомплит)

**vscode-go** - [расширение](#) от Microsoft  
[nezorflame/vscode-config](#)

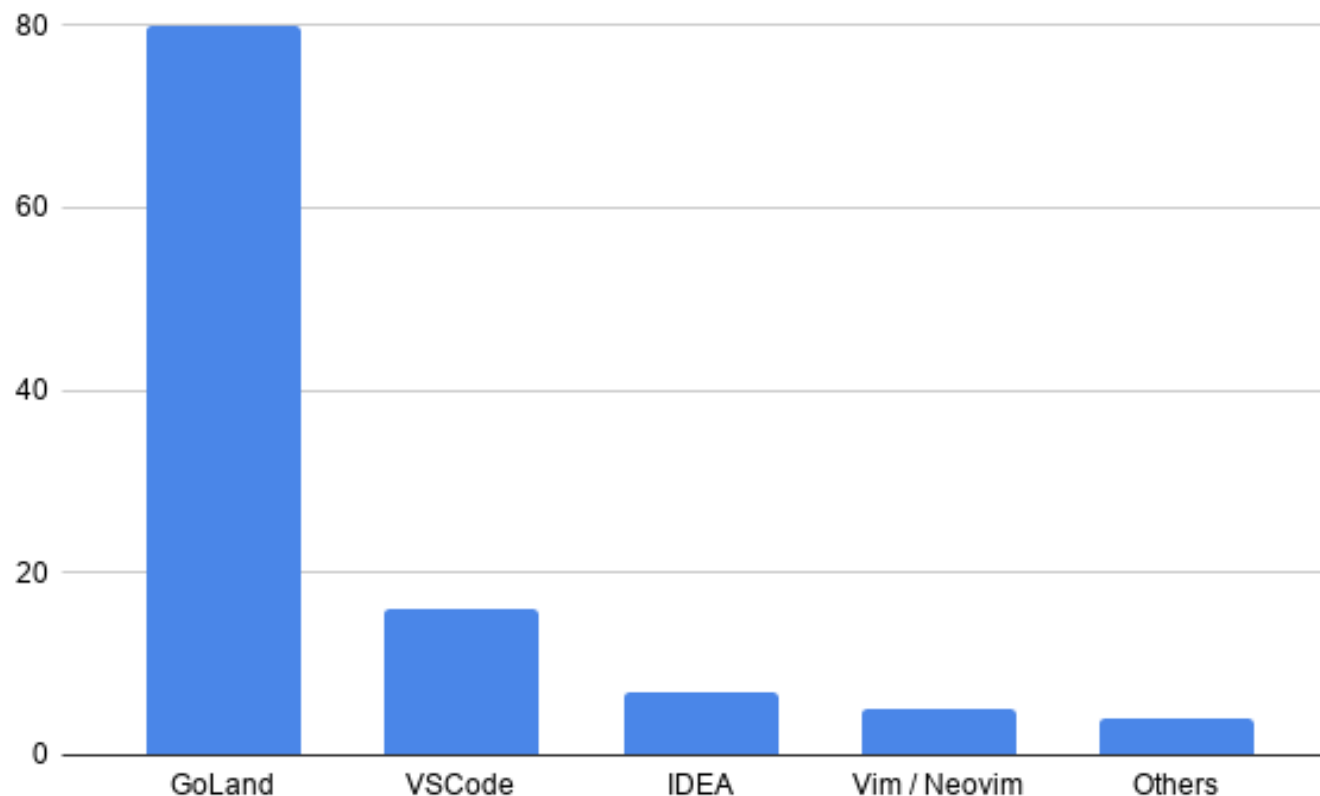
# Сравнение фич

Feature	GoLand	VSCode
Автокомплит	✓	✓
Быстрые фиксы	✓	⚠
Рефакторинг	✓	⚠
Документация по наведению	✓	✓
Кодогенерация	✓	✓
Определение рекурсий	✓	✗
Поиск использования	✓	✓
Поиск имплементации	✓	⚠
Работа с go.mod-файлом	⚠	✓

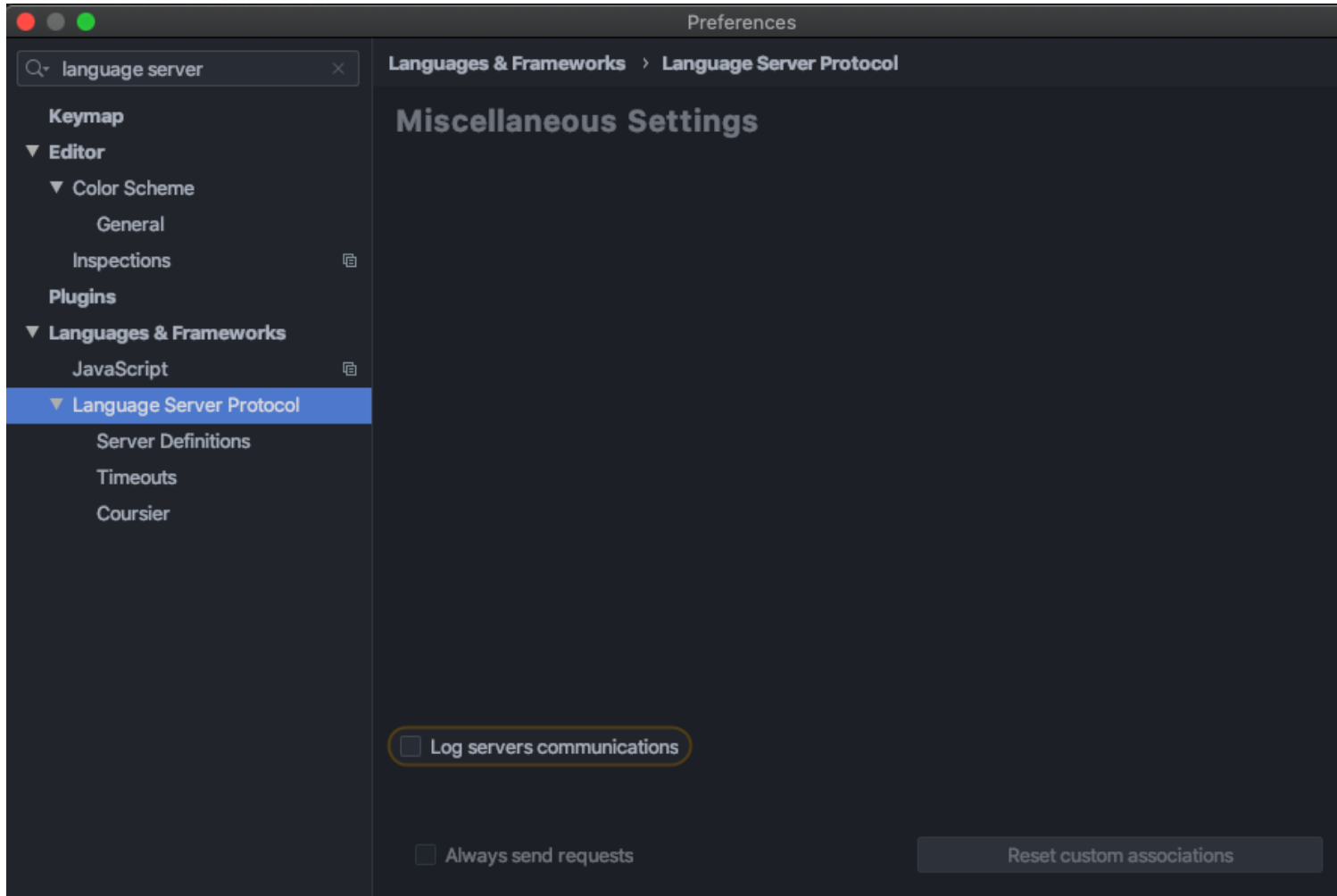
# Популярность в мире



# Использование в Avito



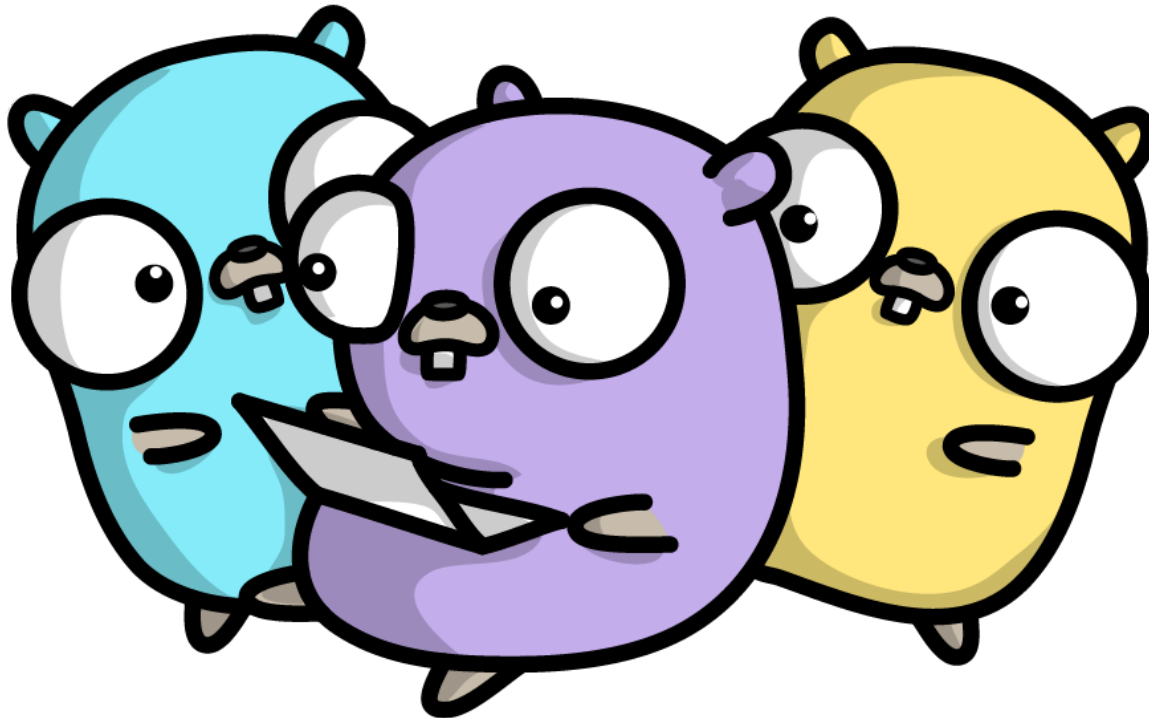
# GoLand + LSP! (soon?)



Секция 05

# Feature Demo

# Feature Demo



Секция 06

# ИТОГИ



# Итоги

# Итоги

- LSP упрощает поддержку языков в IDE / редакторах




# Итоги

- LSP упрощает поддержку языков в IDE / редакторах
- 1 IDE - all languages

# Итоги

- LSP упрощает поддержку языков в IDE / редакторах
- 1 IDE - all languages
- Для Go: gopls  
(PR'ы приветствуются!)

# Итоги

- LSP упрощает поддержку языков в IDE / редакторах
- 1 IDE - all languages
- Для Go: gopls  
(PR'ы приветствуются!)
- GoLand vs VSCode =   

# Полезные ссылки

- [Language Server Protocol](#)
- [Langserver.org](#)
- [Go, pls stop breaking my editor](#)
- [gopls user documentation](#)
- [Rules of contribution](#)
- [nezorflame/vscode-config](#)

**avito.tech**

# Вопросы?

**GopherCon Russia 2020**  
**09.08.2020**



Илья Данилкин  
@nezorflame