



# 云服务 x IDE 可以碰撞出哪些火花?

程柳锋(lewischeng)

# 分享大纲

行业云上 IDE 产品分析

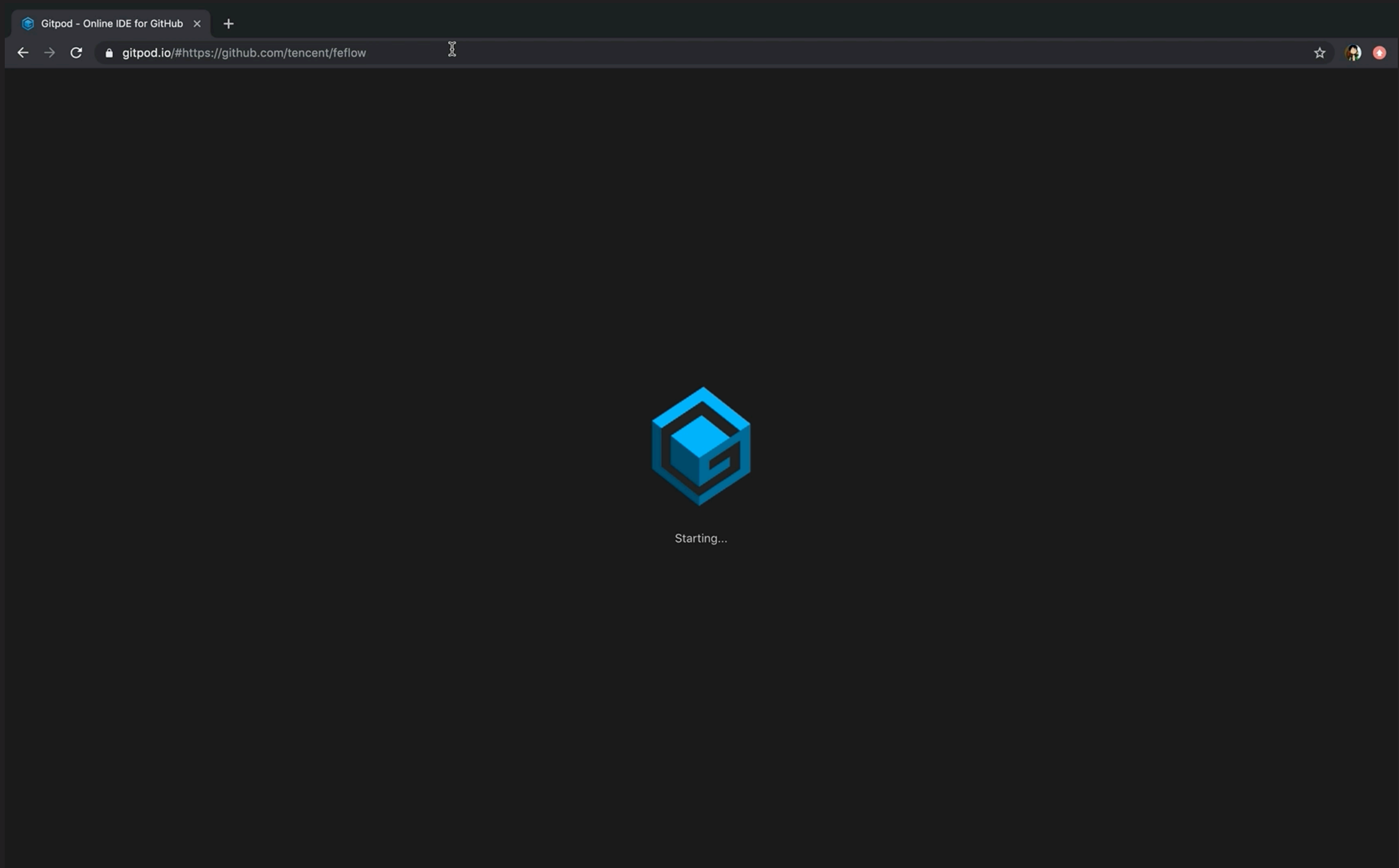
云上 IDE 的优势和架构

VSCode 的源码分析

# 行业云上 IDE 产品分析

# 方式一：云开发体验(Gitpod)

Gitpod: 是一款 Github 的在线 IDE



# 方式二：编译和运行VSCode OSS版

## 1. 克隆代码

```
$ git clone https://github.com/microsoft/vscode.git
```

## 2. 安装依赖

```
$ yarn
```

## 3. 构建

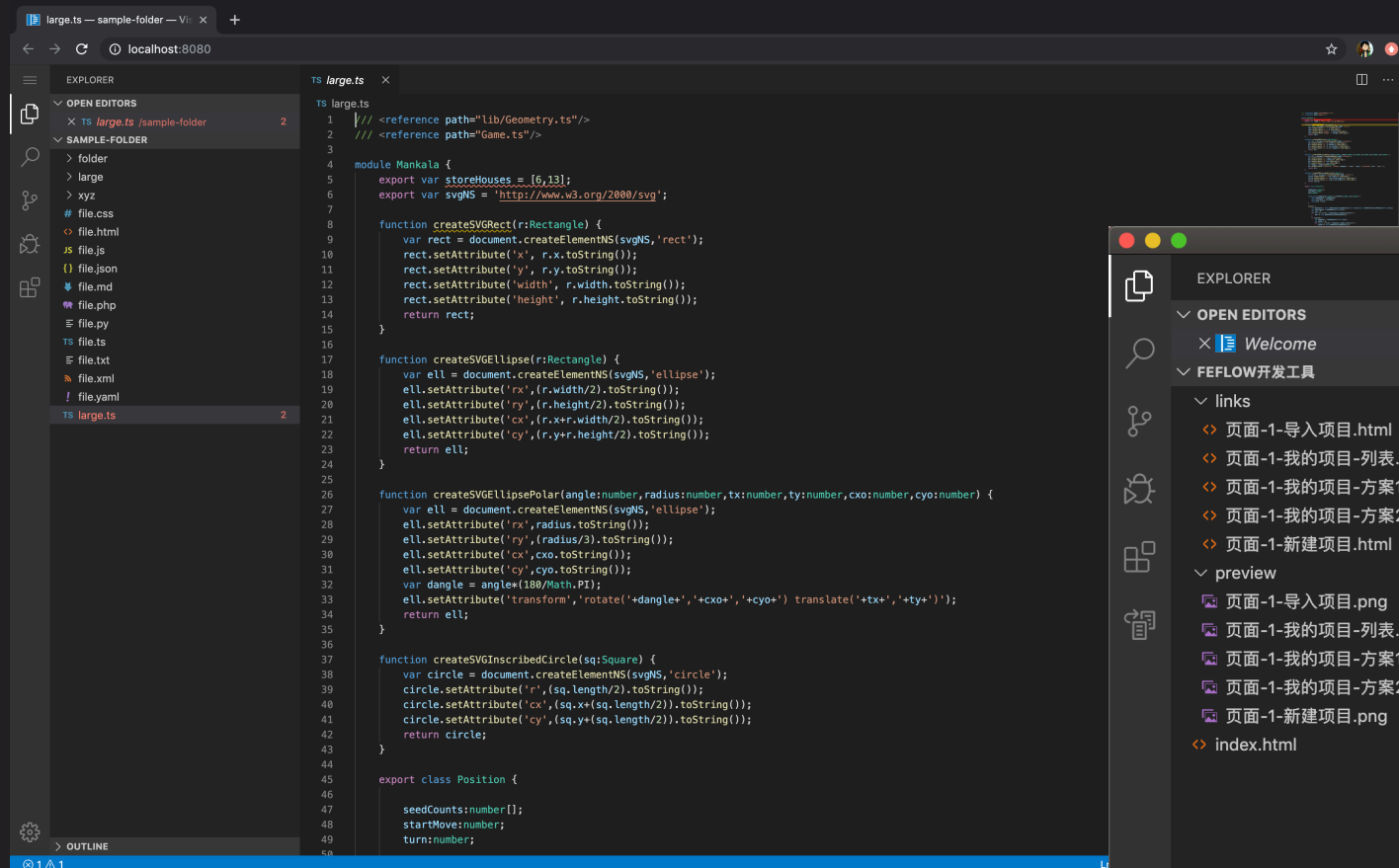
```
$ yarn watch
```

## 4. 运行

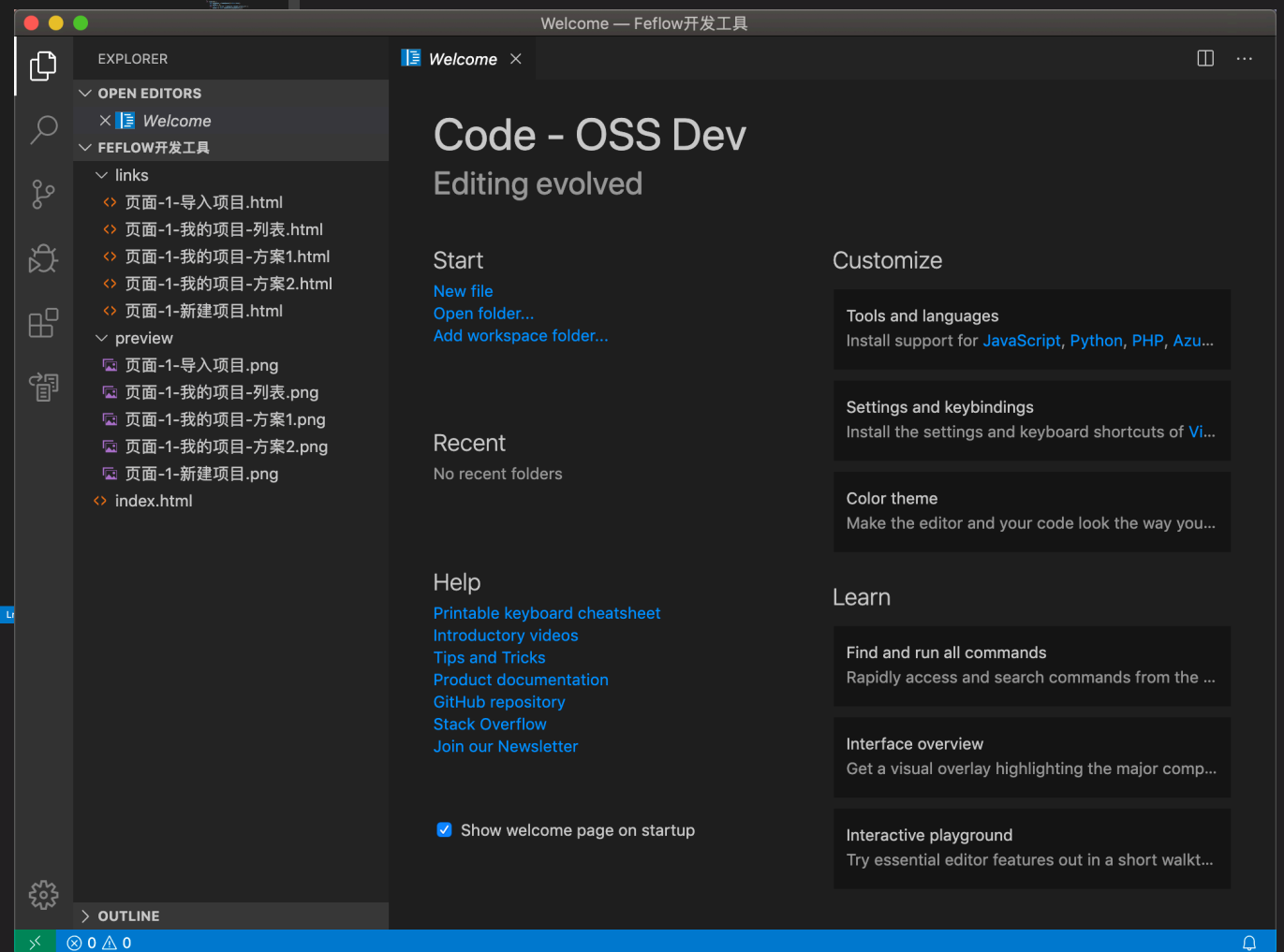
```
$ yarn web          # 运行 web 版本
```

```
$ ./scripts/code.sh # 运行 Electron 桌面版
```

# 方式二：运行效果



Web 版



桌面版

# 行业云产品 IDE 有哪些?

腾讯云: Cloud Studio

华为云: Cloud IDE

阿里云: 云 IDE

亚马逊: AWS Cloud9

谷歌云: Shell editor

Gitpod

CodeEnvy

StackBlitz

CodeSandbox

WebAssembly Studio

DrRacket

更多资源: <https://github.com/styfle/awesome-online-ide>

# IDE 特性和场景对比

**Monaco Editor:** 微软 Monaco workbench 项目 成长出来的 web 编辑器，基于浏览器，和 VSCode 很大一部分代码（monaco-editor-core）共用。浏览器版本

**VSCode:** 微软 Monaco workbench演变而来，基于 Electron，桌面平台

**Theia:** 由Eclipse Che团队研发的，借鉴 VSCode 的设计理念，和 VSCode UI比较像。特点是：Github、Git 集成比较好，支持 Pull request、CR 等功能。兼容 VSCode 扩展

**ACE Editor:** AWS Cloud9使用的主要 IDE



# Theia

## Theia - VS Code based online IDE

Your  
VS Code  
Extensions

Go Lang

Java

...

Your Extension

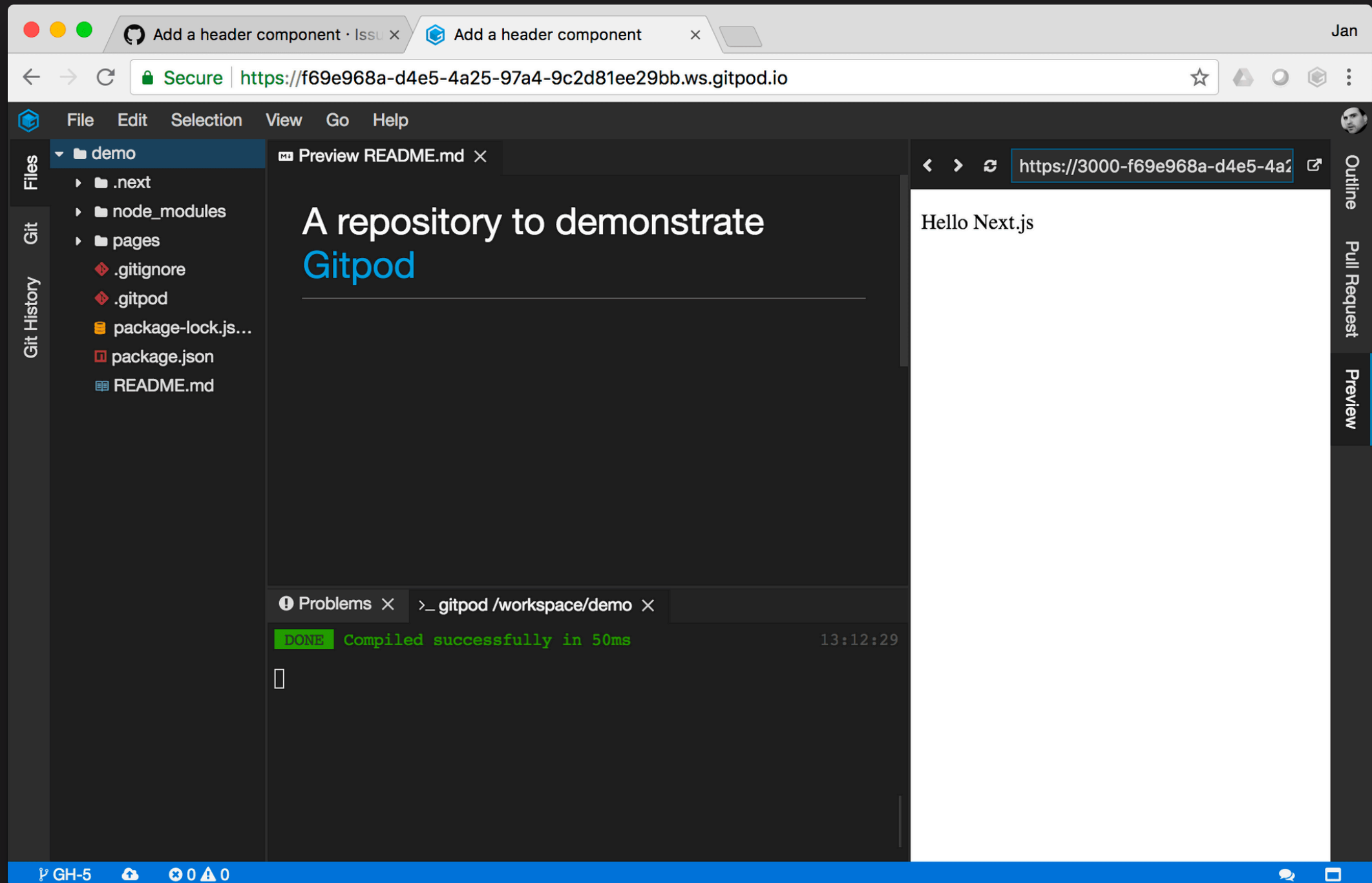
Editor

Preview

Git Integration

Linux Terminals

# Theia 的 Preview 能力



# Theia 的 PR 能力

The screenshot displays the Theia IDE interface within a web browser. The browser's address bar shows a secure connection to a Gitpod workspace. The IDE's menu bar includes File, Edit, Selection, View, Go, and Help. On the left sidebar, the 'Files' panel shows three open files: header.js, about.js, and index.js. The 'index.js' file is currently selected and its content is visible in the main editor. The code defines an 'Index' component that imports 'Header' and renders a paragraph 'Hello Gitpod!'. Below the editor, a terminal window shows a successful compilation message. On the right sidebar, the 'Pull Request' panel is active, showing a 'New' button, a list of 'Added headers', and a 'Create pull request' button. The status bar at the bottom indicates the current file is 'Ln 9, Col 1' in a 'JavaScript' file with 4 spaces.

```
1 import Header from '../components/header'
2
3 const Index = () => (
4   <div>
5     <Header />
6     <p>Hello Gitpod!</p>
7   </div>
8 )
9
10 export default Index
11
```

>\_ gitpod /workspace/demo X

DONE Compiled successfully in 104ms 13:16:09

JanKoehnlein/demo GH-5 0↓0↑ 0⚠0

Ln 9, Col 1 Spaces: 4 JavaScript

# Theia 的 CR 能力

The screenshot displays the Theia IDE interface for a code review. A modal dialog titled "Submit your 1 pending comment" is open, allowing the user to provide feedback. The dialog includes a text input field with the placeholder "Looks good to me" and three radio button options: "Comment" (selected), "Approve", and "Request changes". Each option has a brief description of its function. At the bottom of the dialog are "Cancel review" and "Submit review" buttons.

The background interface shows a code editor with a diff view. The left sidebar contains a "Files" panel, a "Git" panel, and a "Git History" panel. The right sidebar contains a "Pull Request" panel with details about the review, including the reviewer's name (JinKoehnlein), the commit hash (352b082), and the review status (Reviewed a few...). The bottom status bar shows the current file (index.js), the line and column (Ln 1, Col 1), the number of spaces (Spaces: 4), and the language (JavaScript).

# 云上 IDE 的优势和架构

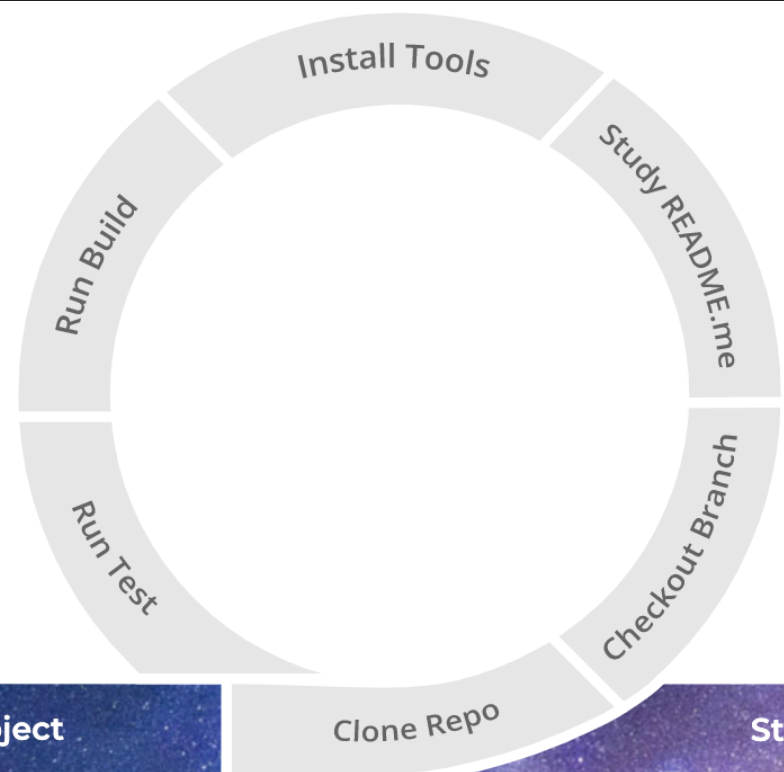
# 开发方式变革

云 IDE 下的开发方式

Choose Project

Start Coding

原始开发方式



Choose Project

Clone Repo

Start Coding

# 云 IDE 的优势

**自动配置开发环境：**新项目无需折腾开发环境、依赖安装等

**一次性开发环境：**不会造成对本机系统的全局污染，可以随时创建

**基于浏览器：** 平板、电脑、手机等多设备均可编程

**统一开发规范：** 统一管控内置插件、规范易于集成

# VSCode 背后的技术



Monaco Editor



TS

UI  
(Electron: HTML/CSS/JS)

Editor 特性  
(Monoco Editor)

Brain  
(智能提示、工具、调试)



# VSCode 的组件

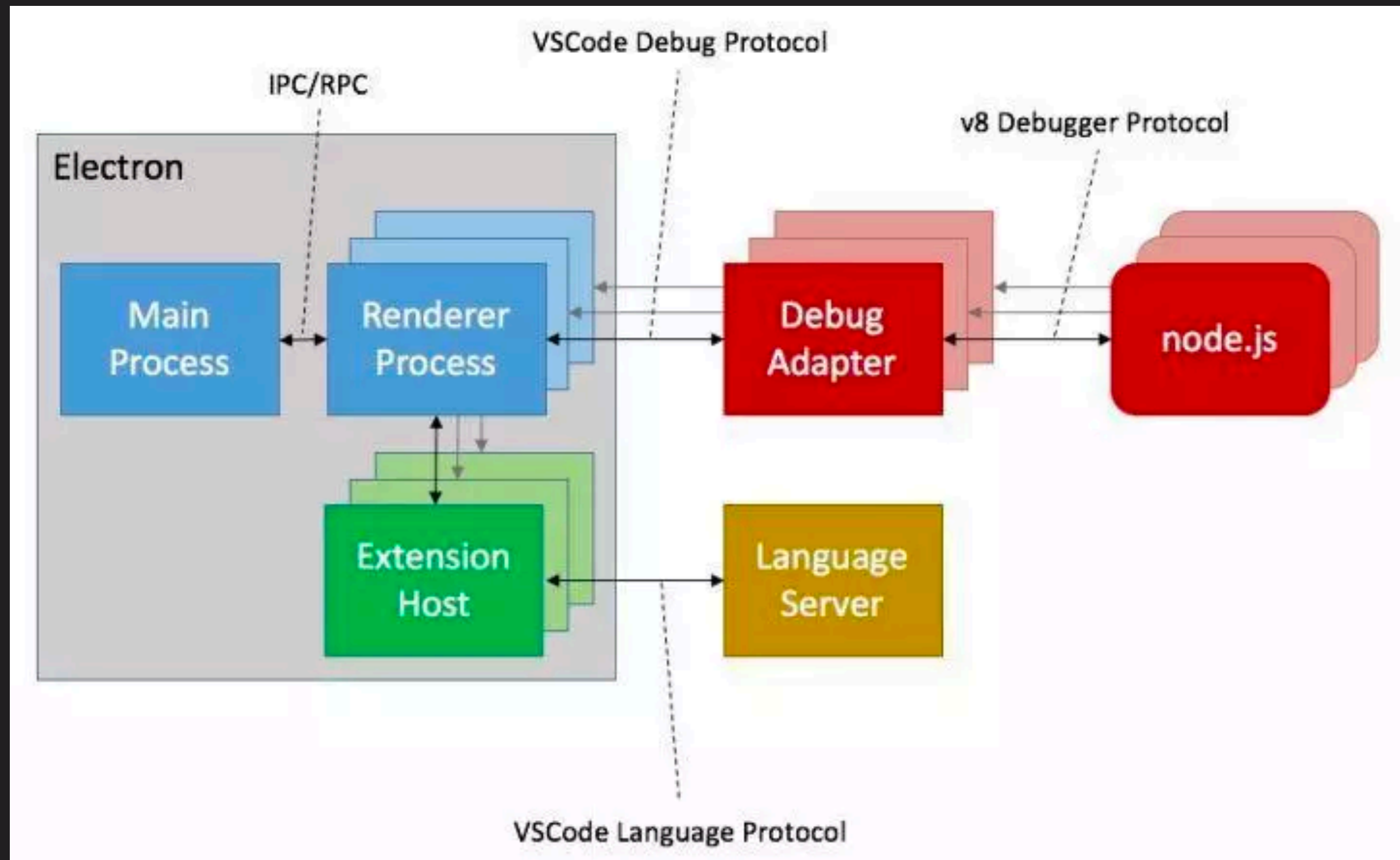
Editor (UI)

Frameworks: 提供语言服务

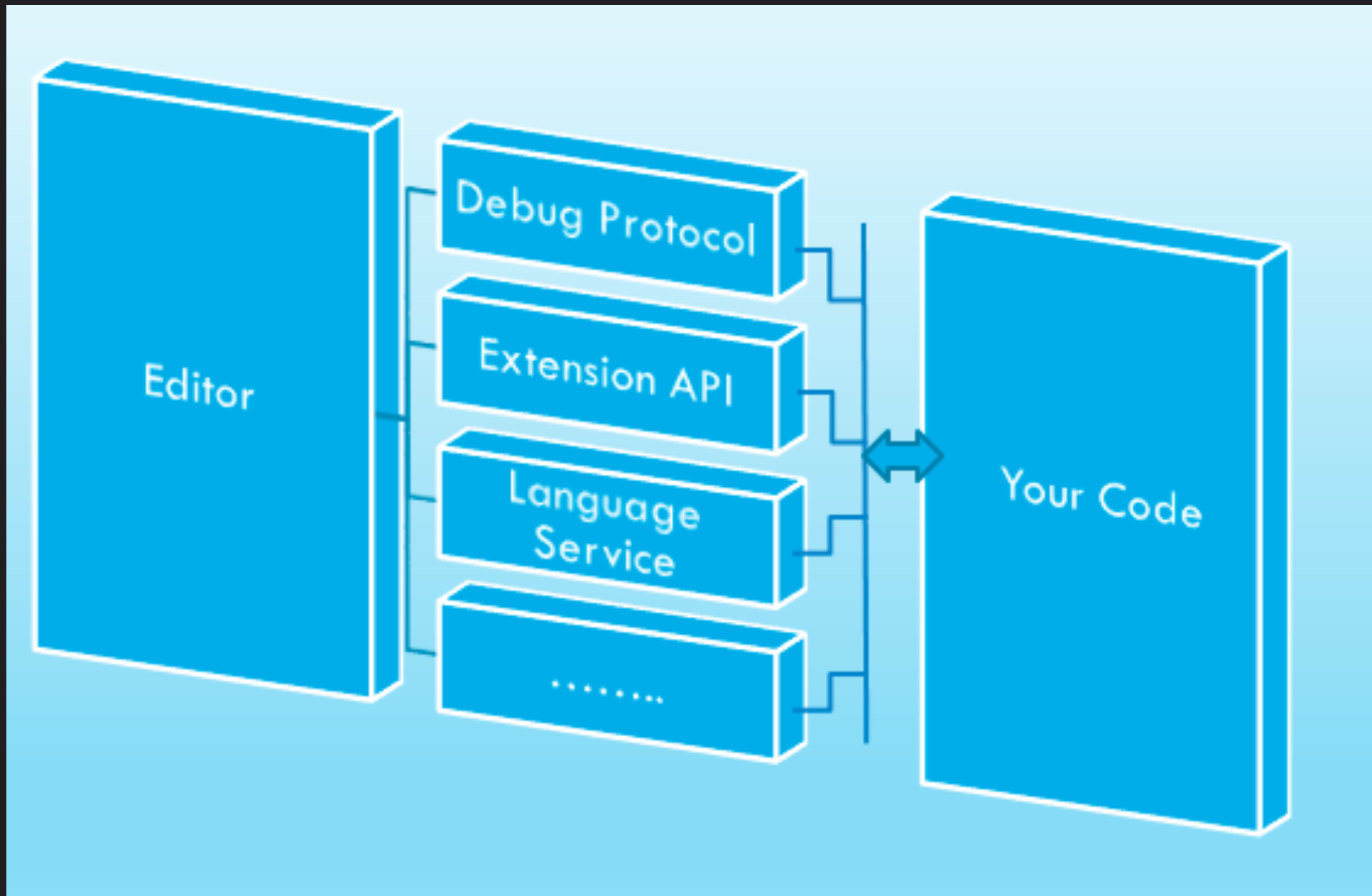
Runtime: Node.js、.NetCore、Unity等

Language: JS、C++、C#、HTML、TS、CSS、SASS 等

# VSCode 架构



# VSCode 和我们的代码如何交互?



# VSCode 的源码分析

# VSCode 的源码结构

```
├─ bootstrap-amd.js    # 子进程实际入口
├─ bootstrap-fork.js   #
├─ bootstrap-window.js #
├─ bootstrap.js        # 子进程环境初始化
├─ buildfile.js        # 构建config
├─ cli.js              # CLI入口
├─ main.js             # 主进程入口
├─ paths.js            # AppDataPath与DefaultUserDataPath
├─ typings
├─   └─ xxx.d.ts       # ts类型声明
├─ vs
├─   └─ base           # 定义基础的工具方法和基础的 DOM UI 控件
├─     └─ browser      # 基础UI组件, DOM操作、交互事件、DnD等
├─       └─ common     # diff描述, markdown解析器, worker协议, 各种工具函数
├─         └─ node      # Node工具函数
├─           └─ parts   # IPC协议 (Electron、Node), quickopen、tree组件
├─             └─ test  # base单测用例
├─               └─ worker # Worker factory 和 main Worker (运行IDE Core: Monaco)
├─ code            # VSCode Electron 应用的入口, 包括 Electron 的主进程脚本入口
├─   └─ electron-browser # 需要 Electron 渲染器处理API的源代码 (可以使用 common, browser,
├─     └─ electron-main  # 需要Electron主进程API的源代码 (可以使用 common, node)
├─       └─ node         # 需要Electron主进程API的源代码 (可以使用 common, node)
├─         └─ test
├─           └─ code.main.ts
├─ editor          # Monaco Editor 代码编辑器: 其中包含单独打包发布的 Monaco Editor 和只能
├─   └─ browser    # 代码编辑器核心
├─     └─ common   # 代码编辑器核心
├─       └─ contrib # vscode 与独立 IDE共享的代码
├─         └─ standalone # 独立 IDE 独有的代码
├─           └─ test
├─             └─ editor.all.ts
├─               └─ editor.api.ts
├─                 └─ editor.main.ts
├─                   └─ editor.worker.ts
├─ platform        # 依赖注入的实现和 VSCode 使用的基础服务 Services
├─ workbench       # VSCode 桌面应用程序工作台的实现
├─ buildunit.json
├─ css.build.js    # 用于插件构建的CSS loader
├─ css.js          # CSS loader
├─ loader.js       # AMD loader (用于异步加载AMD模块, 类似于require.js)
├─ nls.build.js    # 用于插件构建的 NLS loader
├─ nls.js          # NLS (National Language Support) 多语言loader
```

# VSCode 的源码结构

Base

Electron

Monaco Editor

Chromium

Nodejs

TypeScript

Core

Base

Platform

Editor

Workbench

Code

App

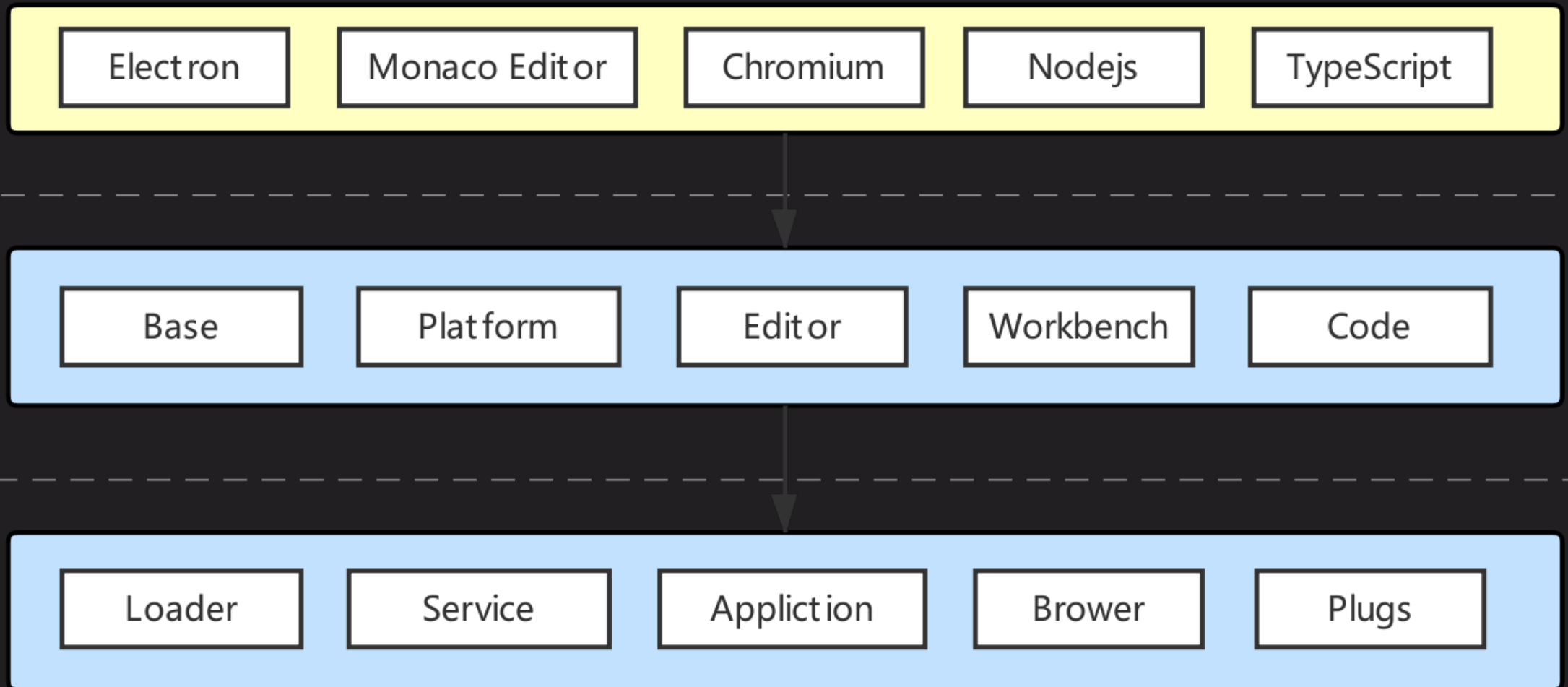
Loader

Service

Appliction

Brower

Plugs



# VSCode 启动过程分析

**入口：**初始化 service，启动主实例

**实例化服务：**CodeApplication 打开窗口，创建共享进程，并且打开窗口

**主界面布局：** 平板、电脑、手机等多设备均可编程

**工作台：** workbench 负责创建主界面，渲染主工作台

感谢收听