

Electron.js in 2019

赵成 @ 第13届D2前端技术论坛

自我介绍

- 赵成 (zcbenz)
- Electron 原作者 (original author) & 全职开发
- 在 GitHub 工作
- 最近的几个业余项目
 - GUI库 <https://github.com/yue/yue>
 - Slack客户端 <https://github.com/yue/wey>

Electron 开发团伙

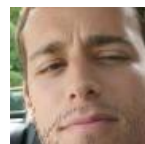
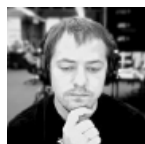
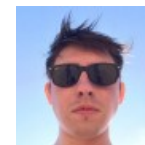
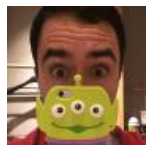
- ▶ 2013 ~ 2014



- ▶ 2015 ~ 2016



- ▶ 2017 ~ 现在



参与开发的主要公司和团队

GitHub

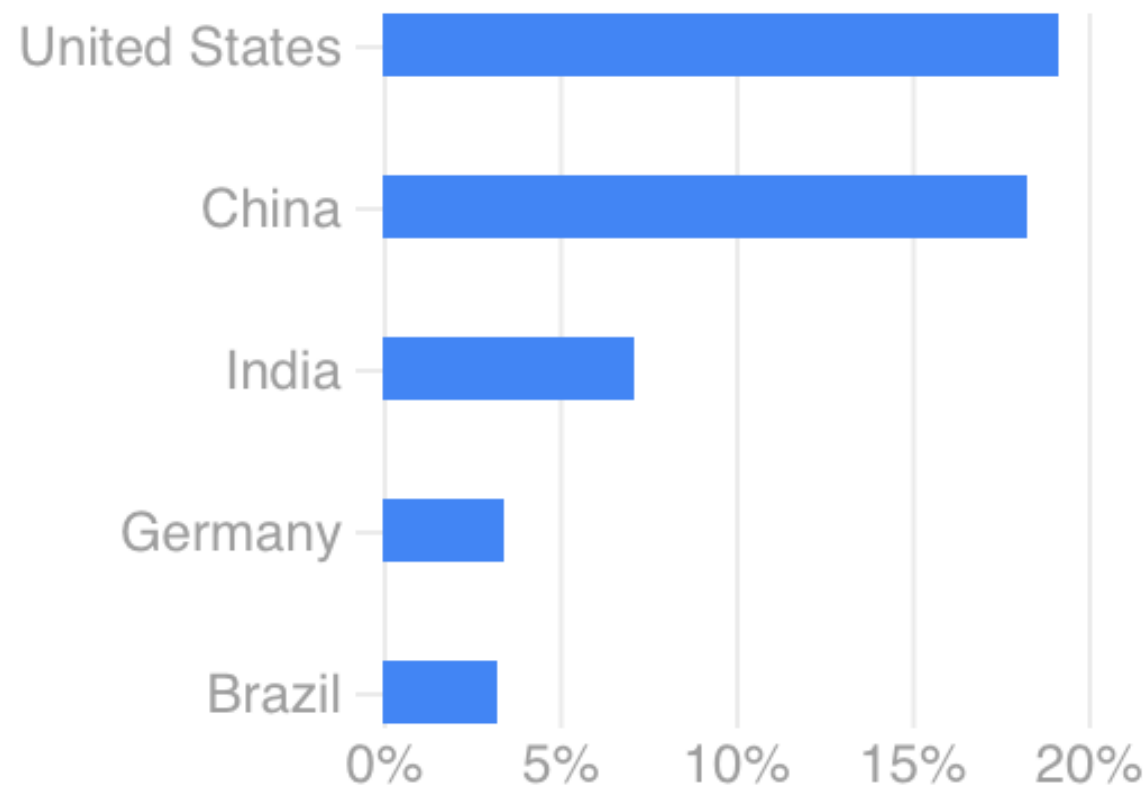


ATLASSIAN



Electron 在中国

- 文档的社区中文翻译完成度达到了 79%
- electronjs.org 中国访问比例达 18%，略低于美国



2018/2019的开发方向

更加安全

更加安全

- 微软的安全专家进行代码审核
 - ◆ 分离内部和外部IPC通道等
- 更安全的默认选项 (5.x)
 - ◆ nodeIntegration 默认关闭
 - ◆ contextIsolation 默认开启
 - ◆ sandbox 默认开启
- 安全漏洞悬赏 (bug bounty)
 - ◆ 发送邮件至 security@electronjs.org

及时更新 Chromium

为什么更新 Chromium 很慢

- Electron 有大量的 Chromium 补丁（目前有98个）
 - ◆ 取消安全检查
 - ◆ 增加接口
 - ◆ 和 Node.js 共同运行
- Electron 增加了多种运行模式（sandbox, contextIsolation, nodeIntegration, nativeWindowOpen...）
- 大量依赖于 Chromium 具体实现的功能（webFrame, remote, mixed-sandbox, webview, protocol...）

提升 Chromium 更新速度

- 降低更新难度
 - ◆ 编译系统切换到 GN（和 Chromium 保持一致）
 - ◆ 将补丁合并到 Chromium 和 Node.js 上游
- 增加人手
 - ◆ 来自 Slack, VS Code, Microsoft Team 的开发者协助 Chromium 更新

矛盾

- 成熟应用的开发者不愿意升级到新版本
 - ◆ 更新 Chromium 一定会引入 bug
 - 单元测试未覆盖到的功能出现问题
 - GPU 渲染机制变换导致应用渲染bug
- 部分开发者催着升级到新版本
 - ◆ Chromium 一直在引入新功能
 - ◆ 旧版本的 Chromium 得不到安全更新

应用反馈项目 (App Feedback Program)

App Feedback Program

- 应用开发者参与新版本 beta 测试
 - ◆ 在你的应用里测试 10000 个用户小时以上
 - ◆ 有人能每周参与讨论应用里遇到的 bug
- 反馈的 bug 作为优先事项修复
- 详情: <https://electronjs.org/blog/app-feedback-program>

Electron 大版本发布流程

1. 有了新 Chromium 更新
2. Electron 更新到最新的 Chromium，在所有单元测试通过后发布 beta 版
3. 参与 App Feedback Program 的应用开发者在自己的应用中测试 beta 版
4. 应用开发者回报 bug
5. 视频会议时讨论并向开发团队分配 bug
6. 修复 bug 后发布新 beta 版
7. 重复 3 - 6 直至发布稳定版

一些常见问题

有什么新功能？

Electron 的开源文化

- ▶ 用户对开发者来说，是邻居而不是客户
- ▶ 新功能主要由第三方公司的开发者主导
- ▶ 提供相应的资源，就可以影响 Electron 的开发

如何影响 Electron 的未来

- 提供开发者参与 Electron 的开发
- 加入 App Feedback Program
- 加入 Electron 的 Slack 讨论组参与讨论
 - ◆ 应用开发者限定
 - ◆ 向 groundwater@github.com 发邮件申请加入
(请简单说明一下应用的情况)

能不能所有 App 共享
一个 Electron runtime ?

共享 runtime 的问题

- App 开发者选择 Electron 大部分时候就是要削减开发成本，而不是降低用户的使用成本
 - ◆ 很多 App 针对特定的 Chrome 版本编写
 - ◆ 需要 App 的开发者测试多个 Electron 的版本
 - ◆ 新的 Electron 更新很可能会导致旧 App 出 bug

所以

- ▶ 用户基数大的 App 不太可能使用共享 runtime
- ▶ 类似的尝试效果都不好 (Chrome Apps, Adobe AIR...)
- ▶ 至少2019年不会有共享的 Electron runtime

谢谢