

《开源软件与社区治理》

第五讲 开源协作与工程全流程

主讲人：赵生宇



内容

1

开源协作

Open Source Collaboration

2

Git

Git

3

GitHub

GitHub

4

深入理解

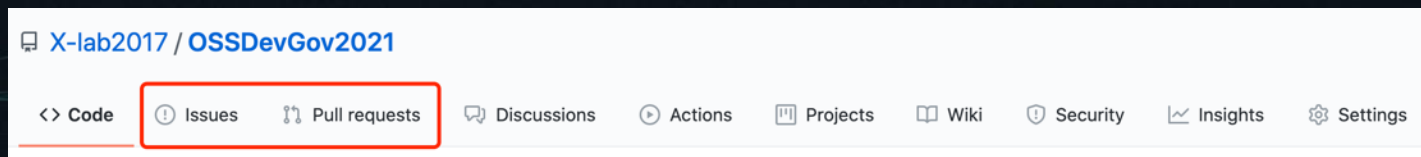
Understanding

01

开源协作

Open Source Collaboration

- 开源
 - 知识成果的开放、共享、创新
- Linux 早期协作
 - 1991 - 2002 邮件列表 + Patch
 - 2002 - 2005 BitKeeper
 - 2005 Git
- 现代开源协作
 - Git
 - 中心化源代码托管平台
 - 1999 Source Forge (SVN, CVS)
 - 2005 Google Code (2014 停止新项目创建)
 - 2008 GitHub (Issue + Pull Request)



02

Git

基本操作

Git 是一个分布式版本控制系统

Fork repo to self account

```
git clone git@github.com:frank-zsy/opensource.git
```

```
git checkout -b feat/add-frank
```

```
git add .
```

```
git commit -sm 'feat: add frank name into file'
```

```
git push origin feat/add-frank
```

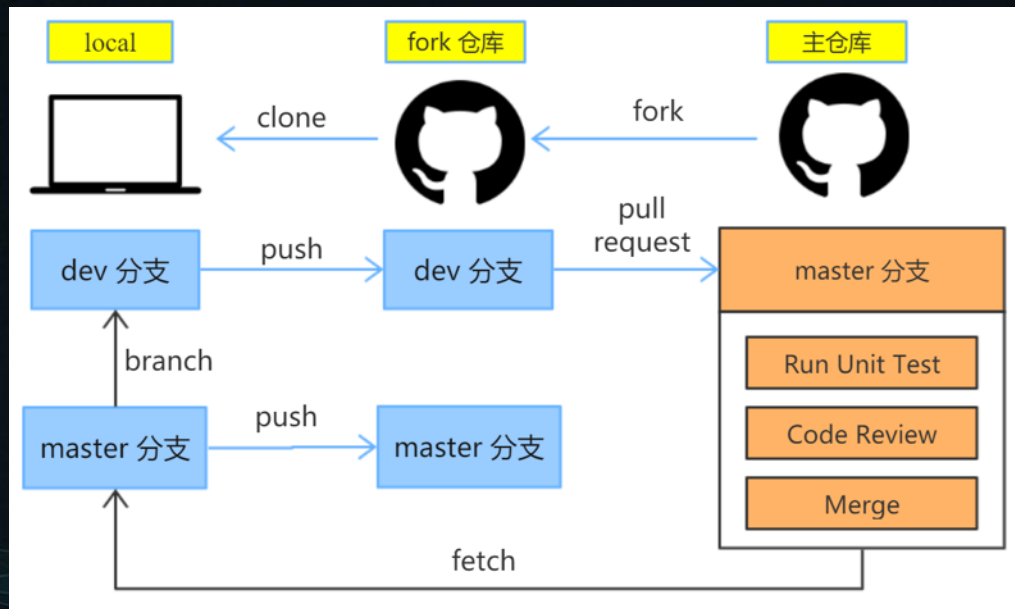
```
git checkout master
```

```
git fetch upstream master
```

```
git rebase upstream/master
```

```
git push origin master
```

Create pull request on GitHub



同步上游



02

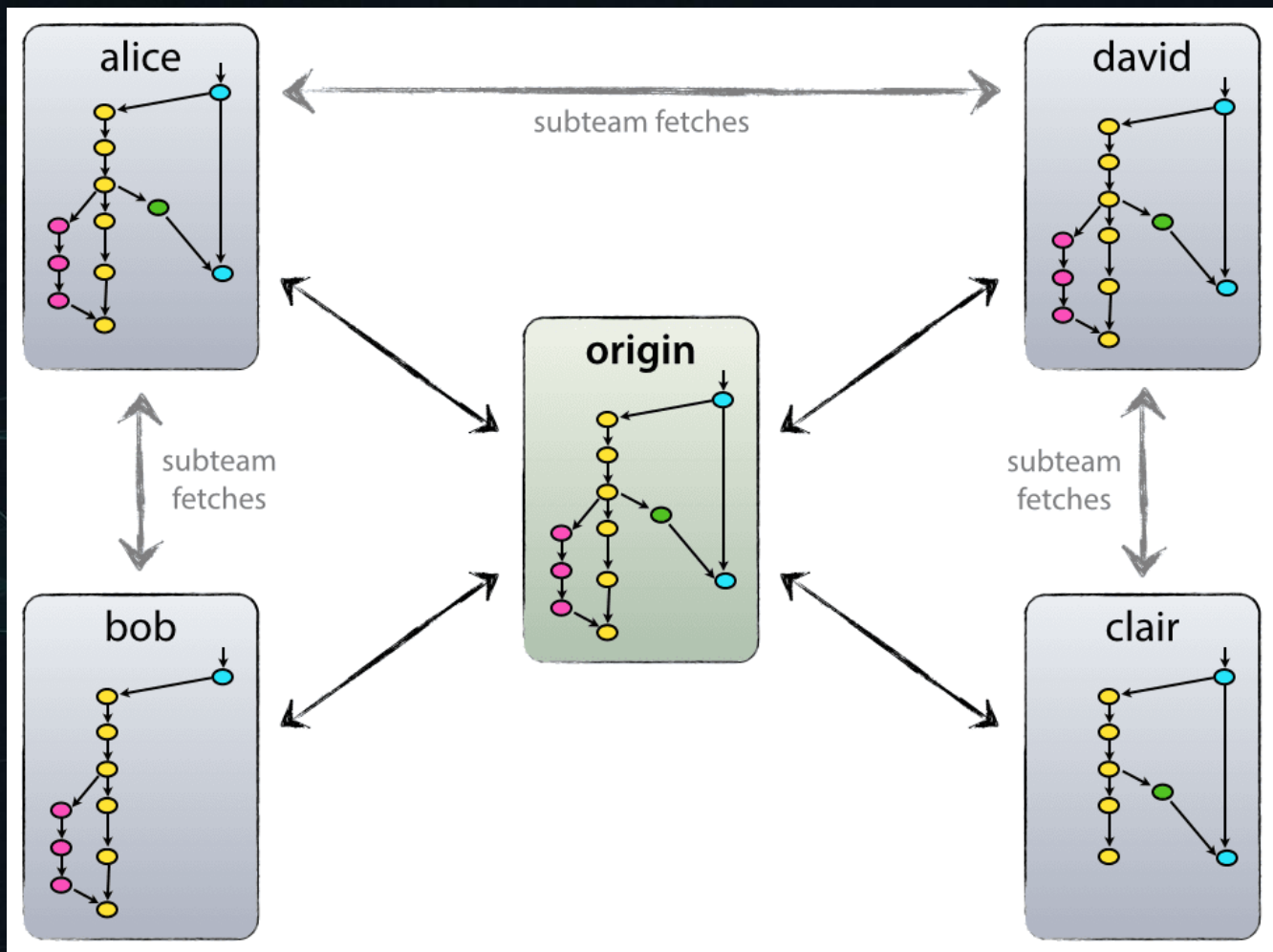
Git

分布式

Git 是一个分布式版本控制系统

即便 GitHub 让仓库看起来是中心化的
但本质上依然是分布式的仓库

所有开发者之间可以互相协作
不一定需要依赖原仓库



02 / Git

分布式

对于分支(branch)的使用

每次有新 feature 开发时都启用一个新分支

若均使用 master 分支开发:

- feature1 开发到一半需要紧急开发 feature2



```
frankzhao@frank-mbp fengari % git log --oneline --graph --decorate --all
* 0c9631c (HEAD -> master, origin/master, origin/HEAD) Merge remote-tracking branch 'fengari/refs/pull/165/head'
|
| * 91323c6 src/{llimits,luaconf}.js: remove luai_apicheck
| * 4f7a043 NEWS: update with unreleased changes
|
|
| * 624d9c4 src/ltable.js: fix finding table length
| * 8cbc96e src/lstrlib.js: remove String.repeat usage in the string library
| * c3bc19f src/loslib.js: implement os.setlocale
| * 9b4b687 {package.json,yarn.lock}: Upgrade dependencies
| * 714ee91 .travis.yml: Add newer node releases to matrix
| * 629049f README.md: Fix typos
| * e8c23d5 LICENSE: Update year
| * 750896c (tag: v0.1.4) Release v0.1.4
| * 0935244 package.json: upgrade dependencies
| * 9ebe39a Remove slot-reserving luaH_set
| * 4ec203c (tag: v0.1.3) Release v0.1.3
| * ccad087 NEWS: Add latest items
| * e83e0be src/loadlib.js: Remove now un-required annotation
| * 4c2408b {package.json,yarn.lock}: Upgrade dependencies
| * e7707b9 src/{luaconf,lobj}.js: Remove locale specific decimal point
| * c3c1703 test/load.test.js: Fix mixed indentation whitespace
| * 83fbf23 src/fengari.js: Whitespace
| * 7a73809 src/lualib.js: Add missing luaopen_base export
| * 682f0cb Merge branch 'fix-lgtm-issues'
|
|
| * e52f46f src/lparser.js: Remove extraneous Proto creation
| * e0a571c src/lapi.js: fix incorrect comparison due to operator precedence
| * 53f8d82 src/ltablib.js: Fix out of range shift
| * 516c945 Merge remote-tracking branch 'sjvs/master' (#131)
|
|
| * 704c498 Add LGTM.com code quality badges
|
|
| * 1d51e3a src/lundump.js: Fix mixed indentation
| * 5b4fecb src/lundump.js: Rename+reorganise functions to be more like C source
```

Git 官方文档: <http://git-scm.com/doc>
可视化 Git 学习: <https://learngitbranching.js.org>
游戏化学习 Git: <https://ohmygit.org/>



03

GitHub

Issue

Issue 的构成

- 标题 Title
- 内容 Body
- 评论 Comments
- 标签 Labels
- 指派者 Assignees
- 所属项目看板 Project
- 里程碑 Milestone

Issue 的典型功能

- 提问、交流
- 提 Bug
- 提需求

The screenshot shows a GitHub issue page with the following details:

- Title:** container has runAsNonRoot and image will run as root warning should include container name #94898
- Status:** Open (green button)
- Metadata:** grosser opened this issue 14 hours ago · 4 comments · May be fixed by #94911
- Comments:**
 - grosser** commented 14 hours ago: when deploying a container that has runAsNonRoot but runs as root this error appears, but for pods with lots of containers it can be confusing which container failed, especially if there are 3-rd party webhooks that inject containers please add the container name to the message, making it container "foo" has runAsNonRoot and image will run as root
 - grosser** added the **kind/feature** label 14 hours ago
 - k8s-ci-robot** added the **needs-sig** label 14 hours ago
 - grosser** commented 14 hours ago: /sig api-machinery
 - k8s-ci-robot** added **sig/api-machinery** and removed **needs-sig** labels 14 hours ago
 - wawa0210** commented 7 hours ago: /assign
I can try this fix
 - k8s-ci-robot** assigned wawa0210 7 hours ago
- Assignees:** wawa0210
- Labels:** kind/feature, sig/api-machinery
- Projects:** None yet
- Milestone:** No milestone
- Linked pull requests:** Enhance log information of verifyRunAsNo...
- Notifications:** Subscribe button, You're not receiving notifications from this thread.
- Participants:** 3 participants (wawa0210, grosser, k8s-ci-robot)

异步沟通

优点:

- 跨时区合作
- 个人开发者时间成本低

挑战:

- 总体沟通时间成本高
- 需要尽量减少沟通的回合数
- 需要尽量在一次沟通中表达尽量多的内容

信息沉淀

优点:

- 所有信息都沉淀下来，可检索
- 沉淀信息可为后来人提供有效的查询路径



03

GitHub

Issue

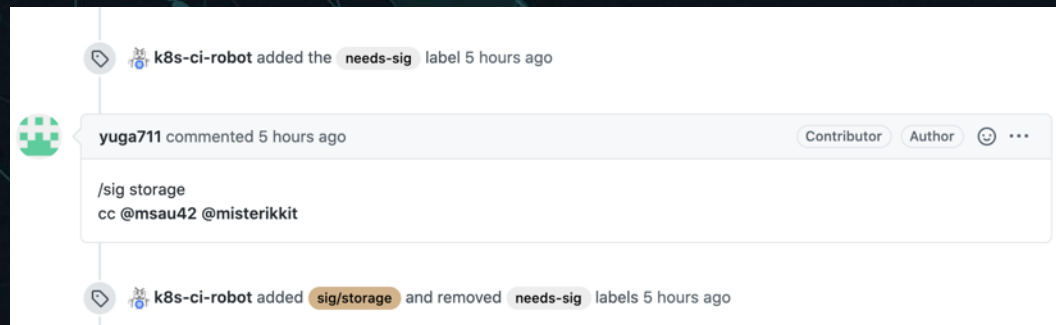
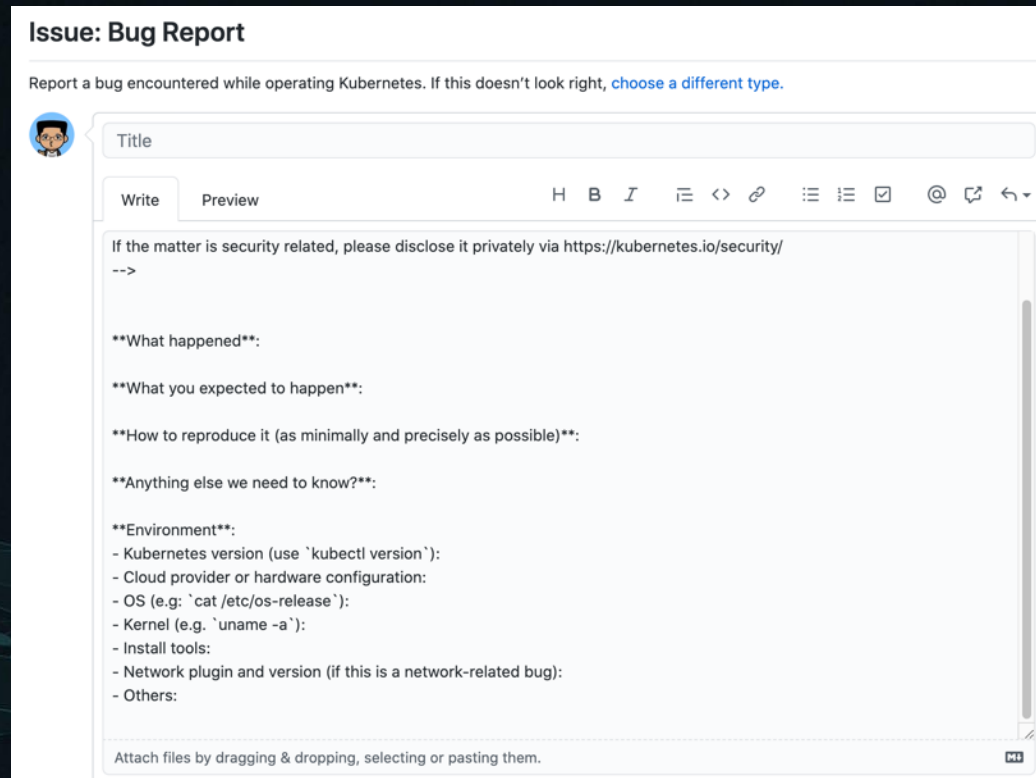
如何降低总体的沟通成本

开发者

- 提 Issue 之前先搜索一下有无相关历史 Issue
- 提交的内容的尽量详实充分
 - 例如如果是 Bug, 说明运行环境、参数配置等
 - 最好可以提供日志或最小可复现仓库

维护者

- 维护良好的社区沟通环境 CODE_OF_CONDUCT.md
- 提供有效的沟通指引 CONTRIBUTING.md
- 使用模板指引开发者提供更全面的信息
- 有效的流程设计使 Issue 可以被更快的响应
 - 制度与流程、自动化



04

深入理解

Issue

管理的主要目的：

在现有条件下，通过合理的组织和配置各种资源，提高生产力水平。

异步沟通 -> 减少同步沟通而减少核心维护者的时间成本

信息沉淀 -> 减少开发者搜寻解决方案的成本和维护者反复回答的成本

流程清晰 -> 减少决策的心智成本同时降低响应时间从而使开发者有更好的体验

流程自动化 -> 减少组织的制度成本而使整体项目可以更加顺滑的运转

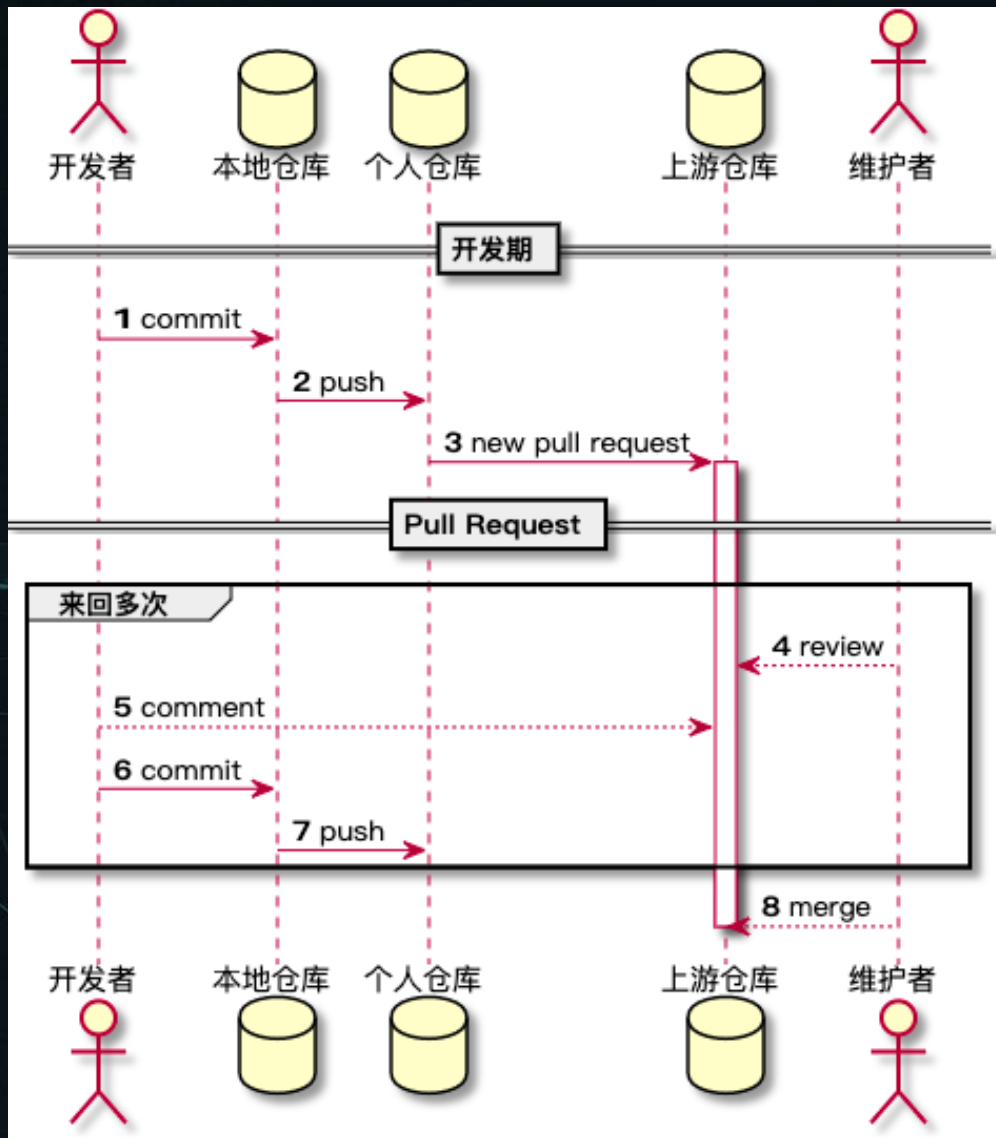
03

GitHub

Pull Request

PR 协作的基本流程

- 1 开发者在本地仓库开发
 - 2 本地仓库提交到 GitHub 个人仓库
 - 3 从个人仓库向上游仓库提起 PR
 - 4 维护者 Review PR
 - 5 开发者与维护者讨论细节
 - 6,7 需要修改则开发者添加新的修改并提交
 - 8 最终维护者合入 PR
-
- 自动化测试?



03

GitHub

Pull Request

PR 的提交与 Review

开发者

- 提交 PR 时，除了提交文件 commit 外
- PR 的标题与内容

维护者

- 通过 PR 的标题与内容了解 PR 内容
- 确定 PR 中的文件修改是正确、合理的
- 不正确、不合理之处与开发者进行讨论
- 将 PR 合入

refactor: modify lua vm #352

Merged menbotics merged 1 commit into hypertrons:master from frank-zsy:modify-lua

Conversation 2 Commits 1 Checks 1 Files changed 6

frank-zsy commented 24 days ago

Signed-off-by: Frankzhaopku syzhao1988@126.com

Brief change log

- Refactor Lua VM to support async function in js

Checklist

- Are tests written for added c: **Merged** analysis-report-... merged 1 commit into X-lab2017:master from sunshineming:readme
- Yes
- No because:
- Did you introduce any new de: **Merged** analysis-report-... merged 1 commit into X-lab2017:master from sunshineming:readme
- No
- Yes
 - Dependency:
 - Reason:

Conversation 6 Commits 1 Checks 0 Files changed 1

sunshineming commented 18 days ago

add invite link of Slack into README.md

16 16

17 + ## Communication

18 +

19 + Click here to join the [slack communication group](https://join.slack.com/t/x-github-analysis/shared_invite/zt-gz8ykv3l-qP53IpajsZIGsx6dFo_8Lg) if you want to know more about the project and join it.

frank-zsy 18 days ago Member

Sorry, but I think Slack should be capitalized.

frank-zsy 18 days ago Member

And if you use Click here to join, maybe the url should be added on here but not on Slack communication group

sunshineming 18 days ago Author Member

OK

Reply...

04

深入理解

Pull Request

Pull Request 到底是什么？开源到底是什么？

开源是一种大规模分布式异步协作生产软件的生产方式，是一种方法论

PR 是开源中由 GitHub 创造，运行在 git 协作流程之上的 Web 协作模式，用于替代 Linux 开发中使用邮件 patch 交付或多 remote 交互的流程。

问题1：为什么开源目前只用于软件生产，为什么可以协作生产代码，可以生产其他内容吗？

问题2：为什么 PR 流程是有效的，从 svn 到 git 到 PR，到底是什么决定了一个流程的有效性？

问题3：基于问题2，我们是否可以解决问题1？

04

深入理解

Pull Request

问题1：为什么开源目前只用于软件生产，为什么可以协作生产代码，可以生产其他内容吗？

- 文本化为大规模的协作和频繁修改迭代带来了可能
- 代码是软件的文本化表现形式
- 文本的修改是结果是一目了然的，加一字符删一个字符清清楚楚，而二进制的就……

```
docs/index.html
```

↑	@@ -15,7 +15,7 @@
15	15 <div id="app"></div>
16	16 <script>
17	17 window.\$docsify = {
18	- name: 'Hypertrons',
18	+ name: 'X-GitHub-Analysis',
19	19 repo: 'https://github.com/X-lab2017/github-analysis-report-2020',
20	20 coverpage: ['/', '/zh-cn/'],
21	21 loadSidebar: true,



04


深入理解

Pull Request


问题2: 为什么 PR 流程是有效的, 从 svn 到 git 到 PR, 到底是什么决定了一个流程的有效性?

- 为什么会有 git? (B站 git + linus) 真·分布式
- 为什么会有 PR? 更简单的可视化协作流程
- 它们在做什么?

协作规模越大则协作成本越高, 需要有流程机制的设计来控制成本




我认为 Subversion 是史上最无厘头的项目



如果有用户喜欢用 CVS, 那你真不该在这儿

你应该在别处的某个精神病院里呆着的



有个好东西叫做 google.com, 老好用了

这东西咋用呢? 你可能见过的

它里边有个地方可以敲字儿进去

你就敲个"Git", 然后按"运气不错"

你就进到 Git 的主页了

因为没人能第一轮就写出完美的代码 -

除了我之外。但是世界上只有一个我, 对不?

04

深入理解

Pull Request

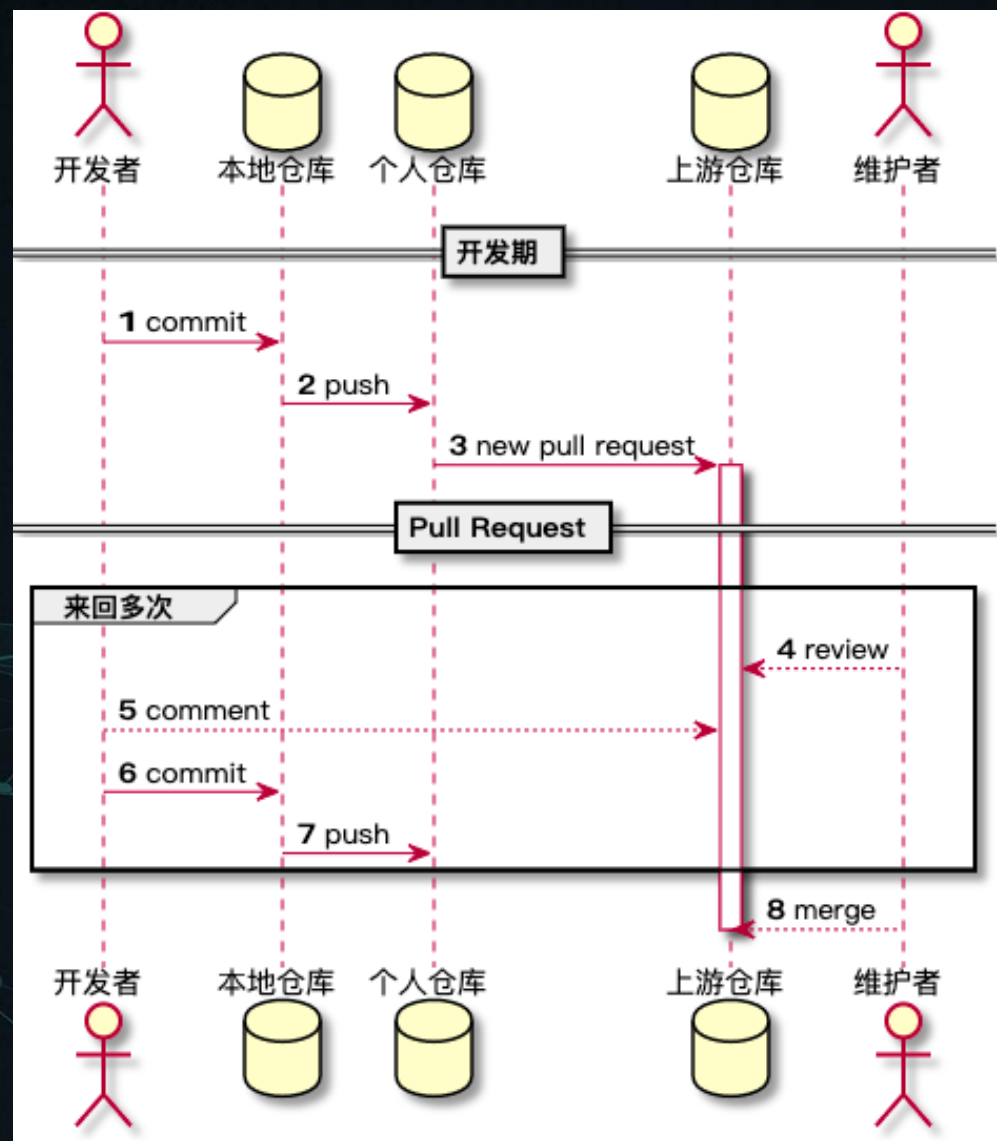
问题3：基于问题2，我们是否可以解决问题1？

需要控制哪些成本

- 代码生产成本
- Review 的成本
 - 理解和发现修改内容的成本
 - 对修改内容的正确性做出评判的成本
 - 与开发者沟通的成本

如果可以大幅降低上述成本，是否可以支持其他类型内容生产的协作呢？

答案是肯定的



04

深入理解

Pull Request

master hypertronstest / +

History Find file Web IDE

Update test.uml
Frank Zhao authored just now f4d453cd

Name	Last commit	Last update
.gitlab	Update hypertrons.json	9 months ago
README.md	Update README.md	2 minutes ago
test.uml	Update test.uml	just now

README.md

HypertronsTest

```
sequenceDiagram
    participant Core
    participant ClientManager
    participant ConfigManger
    participant RepoManager
    participant SchedulerRegistry
    participant EventRegistry

    Core->>ClientManager: load plugins
    ClientManager->>ClientManager: plugins onStart
    ClientManager->>ClientManager: login as GitHub App
    ClientManager->>ClientManager: init tokens
    ClientManager-->>ConfigManger: ClientRetriveInstalledReposDoneEvent
    ClientManager->>ConfigManger: generate config
    ClientManager-->>ConfigManger: ClientRetriveInstalledReposDoneEvent
    ConfigManger->>RepoManager: fetch all repo data
    RepoManager->>SchedulerRegistry: register update repo data
    SchedulerRegistry-->>EventRegistry: Star,Watch,Fork,Issue,Pull
```

https://frank-cdn.opensource-service.com/image/gif/uml_rt_render.gif

04

深入理解

Pull Request

```
<svg
  xmlns="http://www.w3.org/2000/svg" width="600" height="350" preserveAs
  viewBox="0 0 600 350" onload="onSvgLoad()">
<style type="text/css">
<![CDATA[
text {
  text-anchor: middle;
  dominant-baseline: middle;
}
.desc_text {
  text-anchor: start;
  font-size: 16px;
  color: white;
}
.title {
  font-size: 20px;
  font-weight: bold;
}
.subtitle {
  font-size: 12px;
}
.interfacetitle {
  font-size: 10px;
}
rect {
  stroke: black;
  stroke-width: 2;
  fill: transparent;
  rx: 5;
}
```



THANK YOU
