

# 开源社区

## 借助开源项目，学习软件开发

### 《借助开源项目，学习软件开发》

本书使用 [GitBook](#) 来 build 电子书。

联机电子书版本在这里: [Learn Coding With Open Source](#)

开放文档: 《借助开源项目，学习软件开发》

诚邀您的参与!

### 著作权申明

License: CC BY-NC-SA 4.0

 [Creative Commons License](#)

- 本作品选择采用：署名-非商业性使用-相同方式共享 的CC协议。
  - 您可以：复制、发行、展览、表演、放映、广播或通过信息网络传播本作品。以及创作演绎作品。
  - 惟须遵守下列条件：
    - 署名 — 您必须按照作者或者许可人指定的方式对作品进行署名。
      - 署名方式为：在转载或新作品开头的显著位置，注明原作者的姓名、来源及其采用的知识共享协议，与本作品在Github上的原发地址建立链接
    - 非商业性使用 — 您不得将本作品用于商业目的。
    - 相同方式共享 — 如果您改变、转换本作品或者以本作品为基础进行创作，您只能采用与本协议相同的许可协议发布基于本作品的演绎作品。

## 介绍

[百度百科] 开源社区又称开放源代码社区，一般由拥有共同兴趣爱好的人所组成，根据相应的开源软件许可证协议公布软件源代码的网络平台，同时也为网络成员提供一个自由学习交流的空间。由于开放源码软件主要被散布在全世界的编程者所开发，开源社区就成了他们沟通交流的必要途径，因此开源社区在推动开源软件发展的过程中起着巨大的作用。

在如今的软件设计，架构及开发中，开源扮演着越来越重要的角色。

作为开发工程师，服务器、数据库、各种编程语言和框架的源代码，我们可以随时利用这些开源技术实现自己的业务要求。其中，方便且免费地使用这些技术，当然离不开上述各种技术的开源，当今世界，这是一个开源的时代，所以，我们应该主动拥抱开源，工作之余了解与学习开源社区那些事。

## 开源社区分类

其实，开源社区已经存在许久啦，但是对于我和一些人来说，可能比较新奇，所以本文介绍了一些开源社区的概念、如何参与社区、为什么要参与社区？

[CNCF 基金会](#)

[kubernetes](#)、[prometheus](#)、[Dragonfly](#) 等。

## Apache 软件基金会 Apache 项目列表

如 [Kafka](#)、[ZooKeeper](#)、[Hive](#) 等。

## TensorFlow

..... 等等，开源社区很多，找到适合自己的才是最好的。

## 如何挑选开源社区

这里简单地列出了几个条件：

1. 社区/项目方向是否符合你的期望/兴趣。比如希望更多关注在后端领域，那很多前端方向的开源项目就可以放弃。
2. 社区/项目的技术栈是否与你匹配，匹配的技术栈可以使你以更快的速度熟悉代码。技术栈包括编程语言、采用的框架或者库等。
3. 社区接纳新人的友好程度。这对于初入开源，而且并不以开源为生的新人来说，是非常重要的。对于新人来说，开源的贡献活动并不能带给他们任何短期可见的回报，因此需要友好的社区来给予他们正面的反馈，使得贡献活动进入一个良性循环。友好程度，有很多地方可以反映，比如项目是否有清晰易懂的贡献指南，是否具有完善的 CI/CD 支持，是否有完善的晋升路线（新贡献者 - Reviewer - Committer - Maintainer 等），是否可以简单地直接接触到项目的维护人员，有完善的答疑渠道等方面。不过要注意，友好程度跟项目的受欢迎程度是两回事，受欢迎（Star 多）的项目未必有一个友好的社区。一个额外的小建议，刚开始参与开源时，不要选择公司开源的开源项目进行贡献。公司开源的项目通常会不如社区驱动的开源项目友好。
4. 是否是使用过或者熟悉的开源项目。一个好的贡献者也是一个好的用户。如果之前就有使用过的开源项目，会免去熟悉的过程。

最后，在我看来，选择的优先级是：[1] > [3] > [4] > [2]。因为方向无论如何是最重要的，在确定方向后，社区的友好程度应该是更加值得考虑的，这也是我在接触开源以后才发现的。最后，是技术栈的熟悉程度。个人认为，编程语言等等的选择，相对不是那么重要的。如果是 Java 后端的话，可能 Apache or Eclipse 基金会下那一堆开源项目或者 Spring 等等，会比较合适。

整理于知乎话题 [在校如何如何在开源社区中成长？](#)

## 如何参与开源社区

这里以 CNCF 为例，举例说明如何参与 CNCF Kubernetes 开源社区。

### 参与社区

如何参加 CNCF 等开源社区呢？如何向 Kubernetes 等开源仓库提交贡献呢？这里提供一点参考！

### 初识 CNCF

CNCF 是一个开源软件基金会，致力于使云原生计算具有普遍性和可持续性。云原生计算使用开源软件技术栈将应用程序部署为微服务，将每个部分打包到自己的容器中，并动态编排这些容器以优化资源利用率。云原生技术使软件开发人员能够更快地构建出色的产品。

具体详情可以参考以下网址

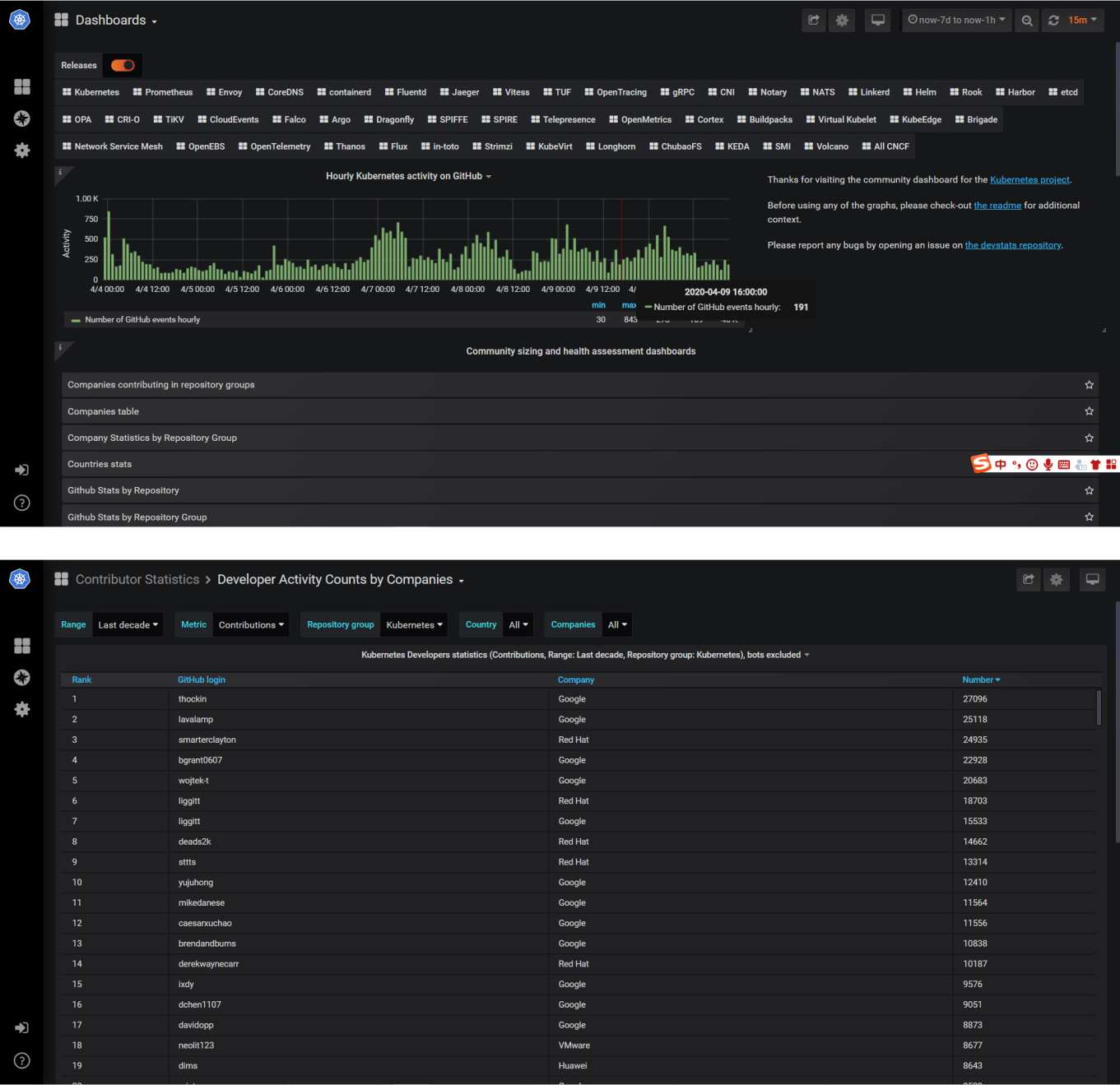
- [英文官网](#)

- [中文社区](#)

CNCF DashBoard 展示

其中 [cncf/devstats](#) 罗列世界上活跃于 CNCF 组织的各个公司以及非营利组织与个人。

Grafana 展示效果：

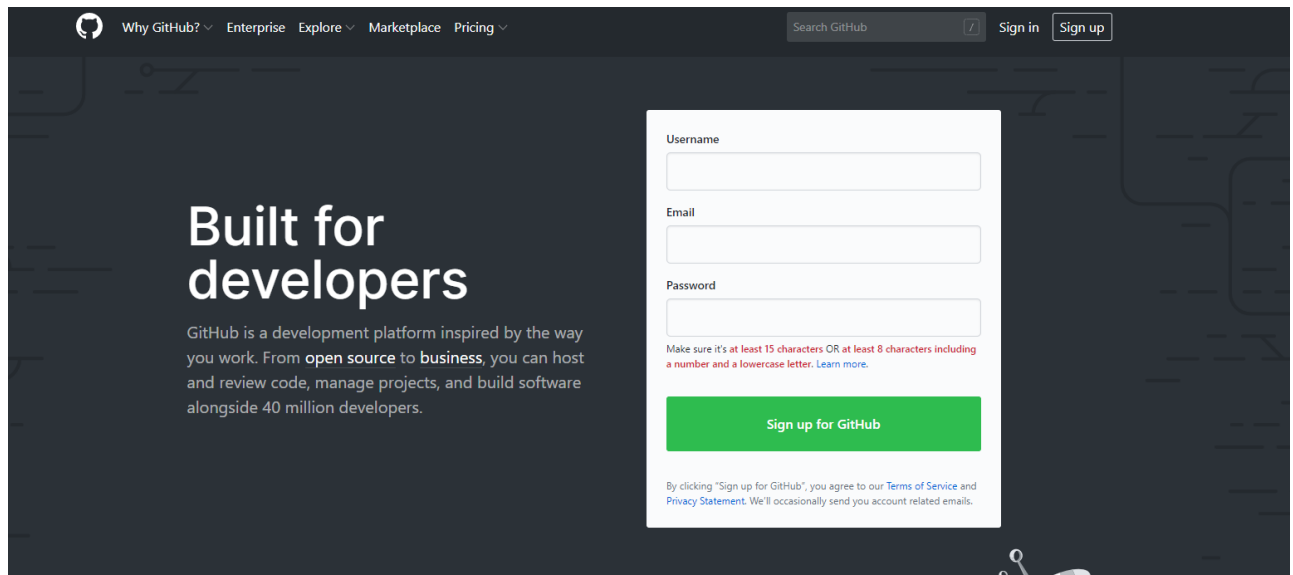




贡献准备

注册 GitHub & 签署 CNCF-CLA

- 注册 GitHub 账户 [注册 Github](#)



提倡 @gmail.com @163.com 等邮箱注册，尽量避免 QQ 邮箱。避免以后关联 Slack 出现问题

- 签署 CNCF-CLA 协议

登录 [linuxfoundation](#) 进行注册以及签署协议。

The image shows the Linux Foundation ID login page. At the top, there's the Linux Foundation logo. Below it, a yellow banner states: "Linux Foundation ID is migrating to an updated platform. For more details, please read our FAQ." The main login area has four social login buttons: "Log in with Facebook", "Log in with Google", "Log in with GitHub", and "Log in with OpenStack". Below these is a horizontal line with the word "Or". Then there are two radio buttons: "I already have a Linux Foundation ID" (selected) and "I need to create a Linux Foundation ID". Below these are input fields for "Username" and "Password". A blue "Log in" button is at the bottom of the login section. Below the button is a link: "Request new password". At the bottom of the page, there's a section titled "About Linux Foundation ID" with a paragraph explaining what LFID is. Below that are links for "Terms of Use" and "Privacy Policy", and a copyright notice: "Copyright © 2019 The Linux Foundation". The Linux Foundation logo is at the very bottom.

**签署 CLA 协议的邮箱应该与 Git 客户端本地配置邮箱保持一致，否则验证不过。访问 CLA 签署网站需要科学上网。所注册邮箱不推荐使用 QQ 邮箱。**

具体流程可以参考：[官方指导](#)

参与贡献

知识必备

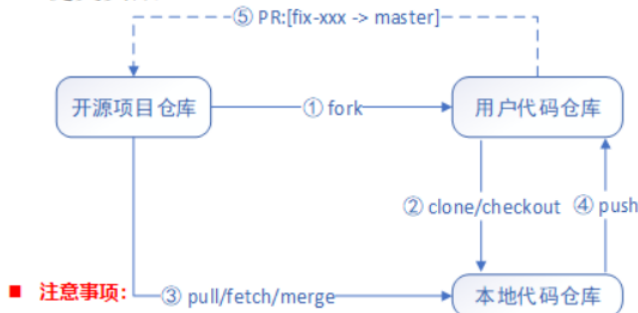
[SIGs and Working Groups](#)，它们是 Kubernetes 社区中关注特定模块的组织，Kubernetes 作为一个拥有几十万行源代码的项目，某些小组专门负责不同的某块，如网络、调度、安全等。

目前主要有 API Machinery、Apps、Architecture、Auth、Autoscaling、CLI、Cloud Provider、Cluster Lifecycle、Contributor Experience、Docs、Instrumentation、Multicluster、Network、Node、Release、Scalability、Scheduling、Service Catalog、Storage、Storage、Testing、UI、Usability、Windows。

熟悉 PR 流程

## 提交 PR 的一般步骤

### ■ 提交贡献



### ■ 注意事项:

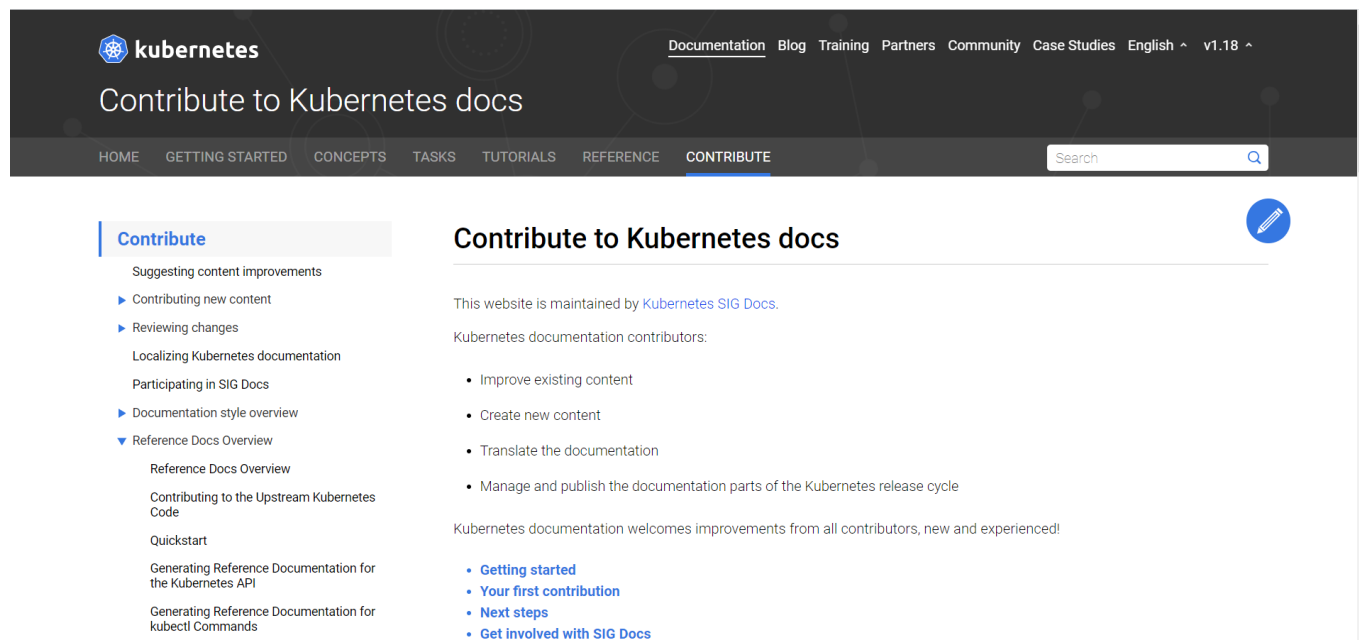
1. 不要在项目已有分支上直接修改代码，每个 PR 请建立独立分支
2. 每次修改代码前，最好与上游做一次同步操作，避免不必要的冲突
3. 合并多次提交步骤：  
# git rebase -i HEAD~n (n 为合并提交的次数)  
# git push -f (第一次提交时不用进行，强制推送)

[示例视频](#)

1. fork 项目到自己的 GitHub 账户
2. 将项目克隆到本地
3. 配置 git name & email
4. 添加上游 (remote) 项目地址
5. 与上游同步代码
6. 切到新分支
7. 修改代码
8. 提交修改，推送到新分支
9. 创建 PR (Pull Request)

数据源: <http://stackalytics.com>

详情参考官网(website): <https://kubernetes.io/docs/contribute/>



**Contribute to Kubernetes docs**

This website is maintained by [Kubernetes SIG Docs](#).

Kubernetes documentation contributors:

- Improve existing content
- Create new content
- Translate the documentation
- Manage and publish the documentation parts of the Kubernetes release cycle

Kubernetes documentation welcomes improvements from all contributors, new and experienced!

- [Getting started](#)
- [Your first contribution](#)
- [Next steps](#)
- [Get involved with SIG Docs](#)

## 以 CNCF Kubernetes 为例

向 Kubernetes 提交 issue/pr, 详见 <https://github.com/kubernetes/community/#contribute>

## Linux 开发环境

熟悉基本 Linux 操作、Git 操作

推荐 [Git 操作教程](#)

**熟悉 Git 操作必不可少**

## 开始第一个贡献

如何向 Kubernetes 贡献你的第一个贡献呢?

The screenshot shows the GitHub interface for the 'kubernetes / kubernetes' repository. The 'Issues' tab is selected, showing 2,045 open issues. A search filter 'is:issue is:open' is applied. A dropdown menu 'Filter by label' is open, displaying two options: 'good first issue' (Denotes an issue ready for a new contributor) and 'help wanted' (Denotes an issue that needs help from the community). The list of issues below includes titles like '--all-namespaces should parse anywhere -n parses', 'Clarify scope of --etcd-servers-overrides option', and 'ESIPP [Slow] should handle updates to ExternalTrafficPolicy field is failing'.

如上图所示，在 Issues 中输入 `help` 或者 `good first issue` 进行过滤，新手一般都是从这里开始参与社区贡献之门。

The screenshot shows the same GitHub interface, but the filter has been updated to 'is:open is:issue label:"help wanted"'. The 'Clear current search query, filters, and sorts' button is visible. The list of issues is now filtered to show only those with the 'help wanted' label. The first issue is 'Human readable duration formats exact year inconsistently', followed by 'Clean up scheduler dependencies on k8s.io/kubernetes packages', 'Need better debug logging for patch operations', 'Describe how resource namespace and name are determined for get/list/watch requests, in authorization and API Priority and Fairness', and 'go-bindata/... matched no packages'.



<b>IntStr GetValueFromIntOrPercent assumes all strings are percentages</b> <span>good first issue</span> <span>help wanted</span> <span>kind/bug</span> <span>sig/api-machinery</span> #89082 opened 29 days ago by JoelSpeed			
<b>No pod restart after updating initContainer image</b> <span>area/kubelet</span> <span>help wanted</span> <span>kind/bug</span> <span>sig/docs</span> #88886 opened on 6 Mar by willise			
<b>Make CSI test suite timeouts configurable</b> <span>help wanted</span> <span>kind/feature</span> <span>sig/storage</span> #88848 opened on 5 Mar by jsafrane			
<b>apiserver: Don't log stack traces on context cancellation?</b> <span>help wanted</span> <span>kind/bug</span> <span>sig/api-machinery</span> #88180 opened on 15 Feb by lavalamp			
<b>Flexvolume e2e tests are skipped in sig-storage test dashboards</b> <span>help wanted</span> <span>kind/bug</span> <span>priority/important-soon</span> <span>sig/storage</span> #88132 opened on 14 Feb by verult			
<b>Scheduler: Add tests for metrics recorder in framework/v1alpha1</b> <span>help wanted</span> <span>kind/cleanup</span> <span>kind/feature</span> <span>sig/scheduling</span> #88038 opened on 12 Feb by bismitaguha			
<b>Provide richer output for "kubelet --version"</b> <span>help wanted</span> <span>kind/feature</span> <span>priority/backlog</span> <span>sig/node</span> #86779 opened on 2 Jan by gprossliner			
<b>/test/e2e/framework: resolve pending TODOs</b> <span>area/e2e-test-framework</span> <span>help wanted</span> <span>kind/cleanup</span> <span>priority/backlog</span> <span>sig/testing</span> #86763 opened on 2 Jan by tanjunchen			
<b>/test: resolve pending TODOs</b> <span>area/e2e-test-framework</span> <span>help wanted</span> <span>kind/cleanup</span> <span>priority/backlog</span> <span>sig/testing</span> #86756 opened on 1 Jan by tanjunchen			
<b>Improve documents and guidance of each script in hack directory</b> <span>help wanted</span> <span>kind/feature</span> <span>sig/testing</span> #86597 opened on 25 Dec 2019 by hase1128			

## 进阶之路

参见 [贡献指南](#)

关注 kubernetes 其他仓库，如 [enhancements community website](#)

enhancements 有关最近 kk 以及未来的发展方向、community 有关 kk 的各种规章制度、website 表示 kk 官网

加入 kk 社区的 [Slack](#) 频道

## 进阶方向

KEP 的全称是 Kubernetes Enhancement Proposal，因为 Kubernetes 目前已经是比较成熟的项目了，所有的变更都会影响下游的使用者，对于功能和 API 的修改都需要先在 kubernetes/enhancements 仓库对应 SIG 的目录下提交提案才能实施，所有的提案都必须经过讨论、通过社区 SIG Leader 的批准。

研究 kubernetes 源码，关注 [KEPS](#)。参与相关 [线上会议](#)。

## kubernetes 汉化

kubernetes 汉化 <https://github.com/k8smeetup>

[K8SMeetup 翻译流程与翻译校稿规范](#)

## Istio 汉化

Istio 汉化 <https://github.com/servicemesh>

Istio 官方文档翻译指导手册

Github 常用语

见 Github 常用语

Github 常见 Topic

GitHub Trend 页面总结了每天/每周/每月周期的热门 Repositories 和 Developers，你可以看到在某个周期处于热门状态的开发项目和开发者。而 GitHub Topic 展示了最新和最流行的讨论主题，在这里你不仅能够看到开发项目，还能看到更多非开发技术的讨论主题，比如 Job、Chrome 浏览器等。

github.com/trending

GoogleJavaPythonk8sgo大数据学习社区常用工具servicemesher微信公众平台百度TranslateStackalytics | Offic...

ch or jump to...Pull requestsIssuesMarketplaceExplore

ExploreTopicsTrendingCollectionsEventsGitHub SponsorsGet email updates

Trending

See what the GitHub community is most excited about today.

RepositoriesDevelopersSpoken Language: AnyLanguage: AnyDate range: Today

microsoft / winget-cli

Windows Package Manager CLI (aka winget)

C++3,94068Built by932 stars today

★ Star

TheAlgorithms / C-Plus-Plus

All Algorithms implemented in C++

C++5,8162,092Built by255 stars today

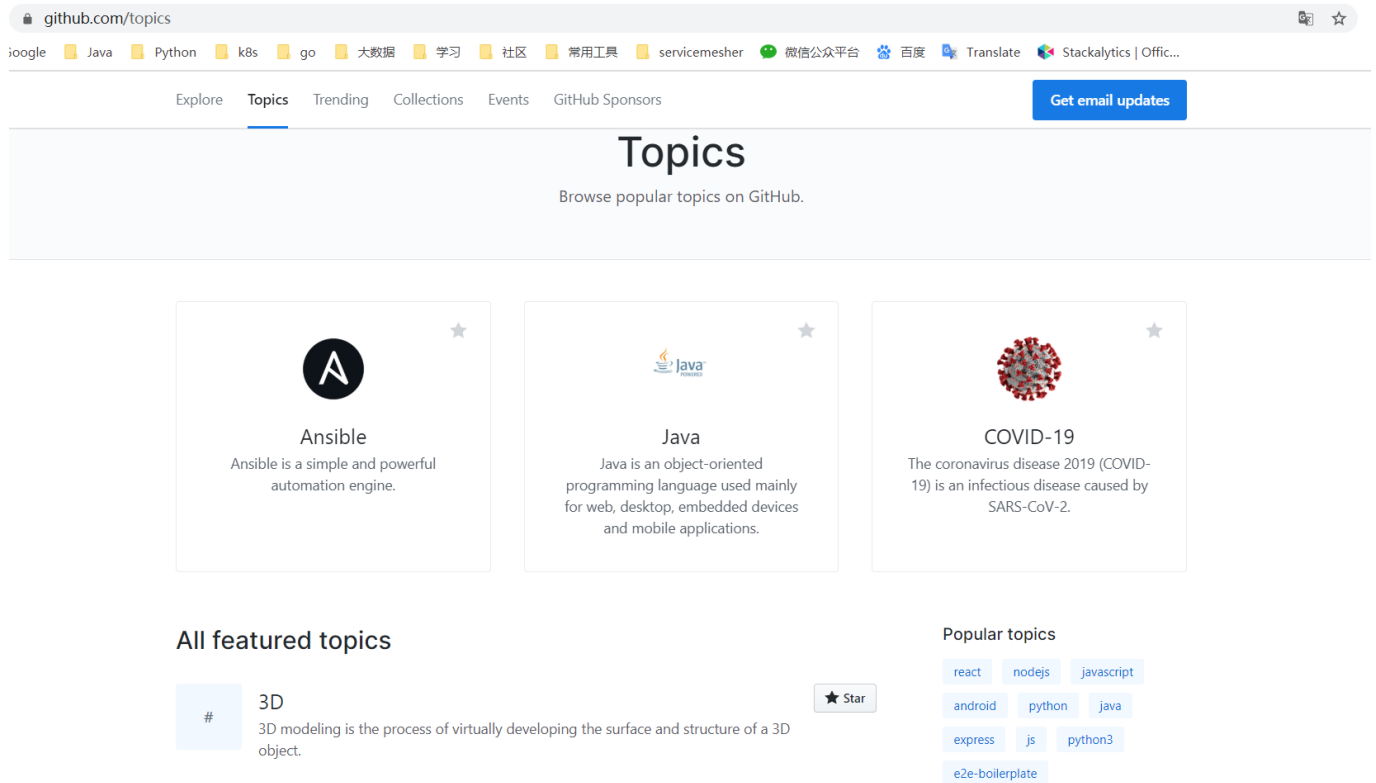
★ Star

idris-lang / Idris2

A purely functional programming language with first class types

Idris25421Built by75 stars today

★ Star



## 期待与收获

Kubernetes 目前已经是云时代的操作系统，作为容器编排领域的事实标准和云原生领域的 Boss，学习与了解 Kubernetes 的使用和原理成为云原生时代开发工程师的必备技能。

开源社区如此之多，选择自己感兴趣的或者对自己从事的工作有帮助的是最好不过的啦，开源与工作相结合是比较推崇的，既可以提升平时工作的效率(社区中比较好用的工具应用到工作中)，也可以提升对开源社区的兴趣(比如你对某项开源项目中的某个 topic 或 issue 感兴趣，则可以尝试用自己所掌握的知识去实现它)。从不同的社区中我们能够学习到不同的东西，比如：社区的管理与运作方式、提升某项技能、增强编码能力与编码规范。

在开源社区能够收获什么呢？

1. 提升编码水平与编码规范。
2. 与世界上最优秀的工程师们交流，能够开拓视野，如 k8s 新增一项功能，必须严格遵循 alpha -> beta -> release -> GA 等阶段、e2e 测试、编写完整的文档、代码注释。
3. 可以快速理解项目的实现原理以及工作流程，提升工作效率，提升个人水平。
4. 大厂加分项。
5. ....

## Github 得力助手

在 Github 上有很多的项目，我们应该如何去抉择呢？有很多种方式提高我们的搜索效率。

如：搜索某个用户：

location:china language:java followers:>=1000

## GitHub 搜索技巧 - 找开发者

搜索条件	备注
location:	location:china, 匹配用户填写的地址在 china
language:	language:javascript, 匹配开发语言为 javascript 的开发者
followers:	followers:>=1000, 匹配拥有超过 1000 名关注者的开发者
in:fullname	jack in:fullname, 匹配用户实名为 jack 的开发者

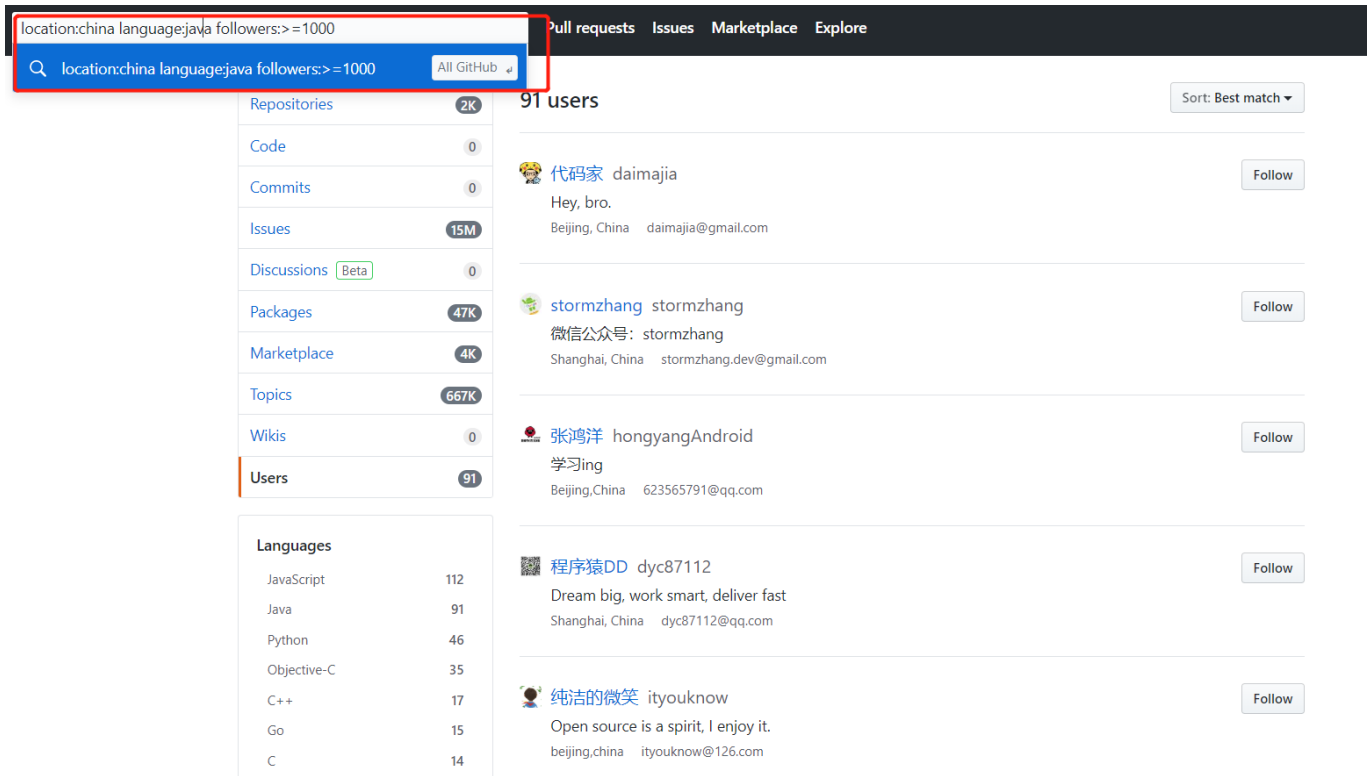
The screenshot shows the GitHub search interface. The search bar contains the query 'location:china language:java followers:>=1000'. Below the search bar, the 'Users' tab is selected, showing a list of 91 users. The left sidebar shows the search filters: Repositories (2K), Code (0), Commits (0), Issues (15M), Discussions (Beta, 0), Packages (47K), Marketplace (4K), Topics (667K), Wikis (0), and Users (91). The 'Languages' section shows counts for JavaScript (112), Java (91), Python (46), Objective-C (35), C++ (17), Go (15), and C (14). The user list includes profiles for 代码家 (daimajia), stormzhang, 张鸿洋 (hongyangAndroid), and 程序猿DD (dyc87112).

### 搜索某个项目

awesome:高并发 language:java starts:>=500 forks:200

## GitHub 搜索技巧 - 找项目

搜索条件	备注
Awesome + 关键字	神奇的关键字 Awesome, 帮助找到优秀的工具列表
stars:	stars:>=500, 匹配收藏数量超过 500 的项目
language:	language:javascript, 匹配以 javascript 作为开发语言的项目
forks:	forks:>=500, 匹配分支数量超过 500 的项目



location:china language:java followers:>=1000

91 users

Sort: Best match

Repositories	Count
Code	0
Commits	0
Issues	15M
Discussions	0
Packages	47K
Marketplace	4K
Topics	667K
Wikis	0
Users	91

Languages	Count
JavaScript	112
Java	91
Python	46
Objective-C	35
C++	17
Go	15
C	14

代码家 daimajia

Hey, bro.

Beijing, China daimajia@gmail.com

stormzhang stormzhang

微信公众号: stormzhang

Shanghai, China stormzhang.dev@gmail.com

张鸿洋 hongyangAndroid

学习ing

Beijing, China 623565791@qq.com

程序猿DD dyc87112

Dream big, work smart, deliver fast

Shanghai, China dyc87112@qq.com

纯洁的微笑 ityouknow

Open source is a spirit, I enjoy it.

beijing, china ityouknow@126.com

## 源码之旅 - 题外话

学习与阅读源码是一门苦差事，参与开源社区也需要耐心与持之以恒的决心。

1. 后端开发者，尤其是 Java Web 从业者，不知是否听说过访问量 3 百万 + 的 [芋道源码](#)。
2. 前端开发者不知是否知道[司徒正美](#) JavaScript 专家，开源框架 avalon 创始人。

## 总结

我参与 Kubernetes 社区，从中学习到了很多，比如不仅学习到了 Go 语言的开发规范、庞大的开源社区的治理方式，还从社区中的很多优秀工程师身上学到了软件设计的一些技巧，与优秀的开发者交流提高了我的见解，这些都是在日常工作中很难得到的。可能与我们平时的工作内容相关，毕竟大部分集中于业务开发，框架的使用。优秀的开源社区和项目中真的是值得我们去学习与借鉴，等着我们去挖掘和学习。加入开源社区吧、后浪们!!!