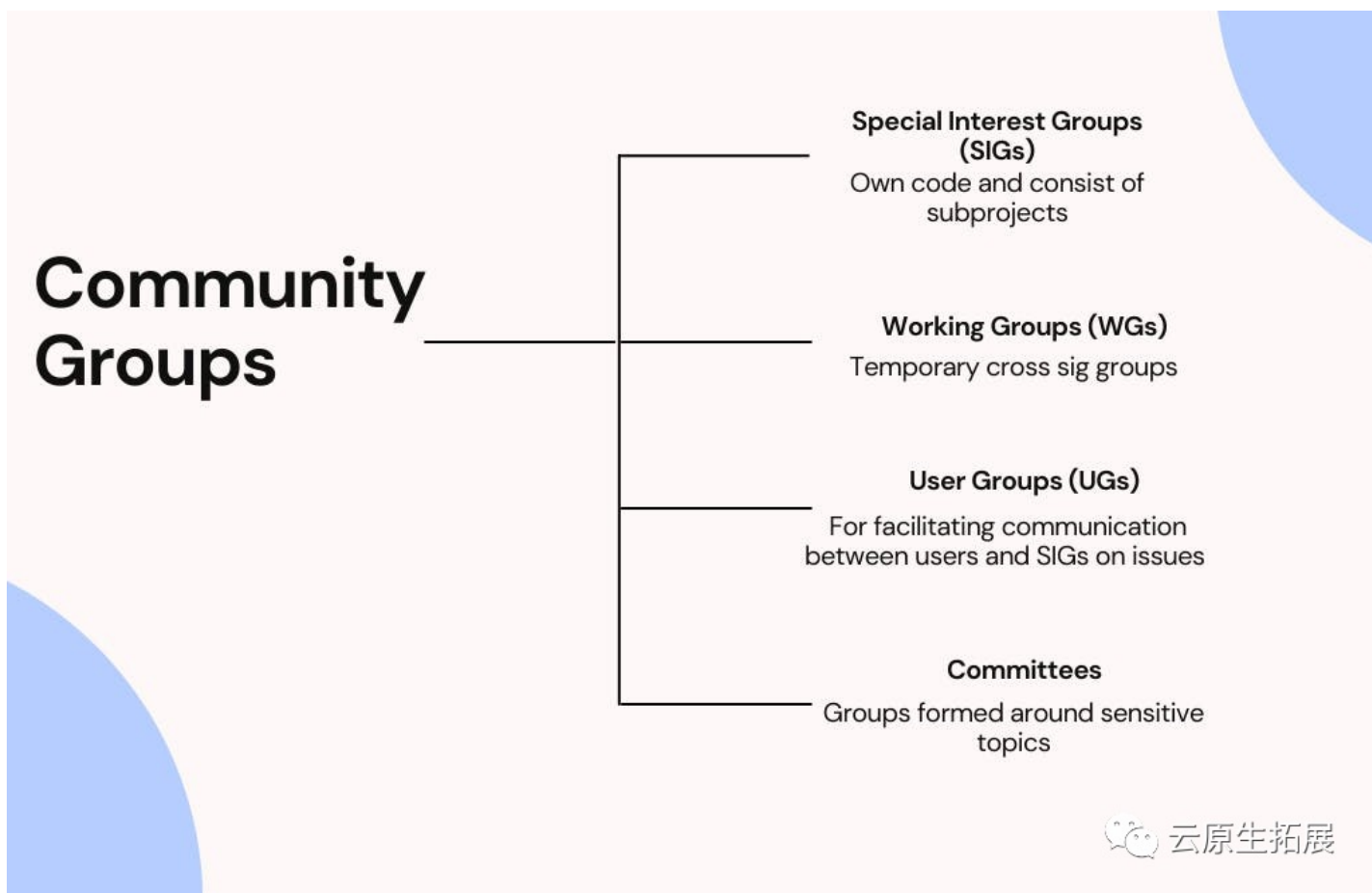


Kubernetes 系列（五十五）如何为 Kubernetes 做出贡献

你正在关注 K8s 的仓库，目前对你来说没有任何意义。这里有很多不同的文件夹和文件。你访问问题页面，看到所有不同的标签，你不知道这一切意味着什么。

让我告诉你，感到不知所措是正常的，因为要理解这一切，你需要退一步，一步一步地看它，因为这是你应该做的。K8s是一个大项目，管理它需要一个合适的结构。如果你把它看作一个整体，那么这个结构就没有意义了，因此你必须以组件的方式来理解它。顺便说一句，这是一个初学者的指南，但即使你是专家，也一定要坚持下去。你可能会学到一些新东西。

社区团体



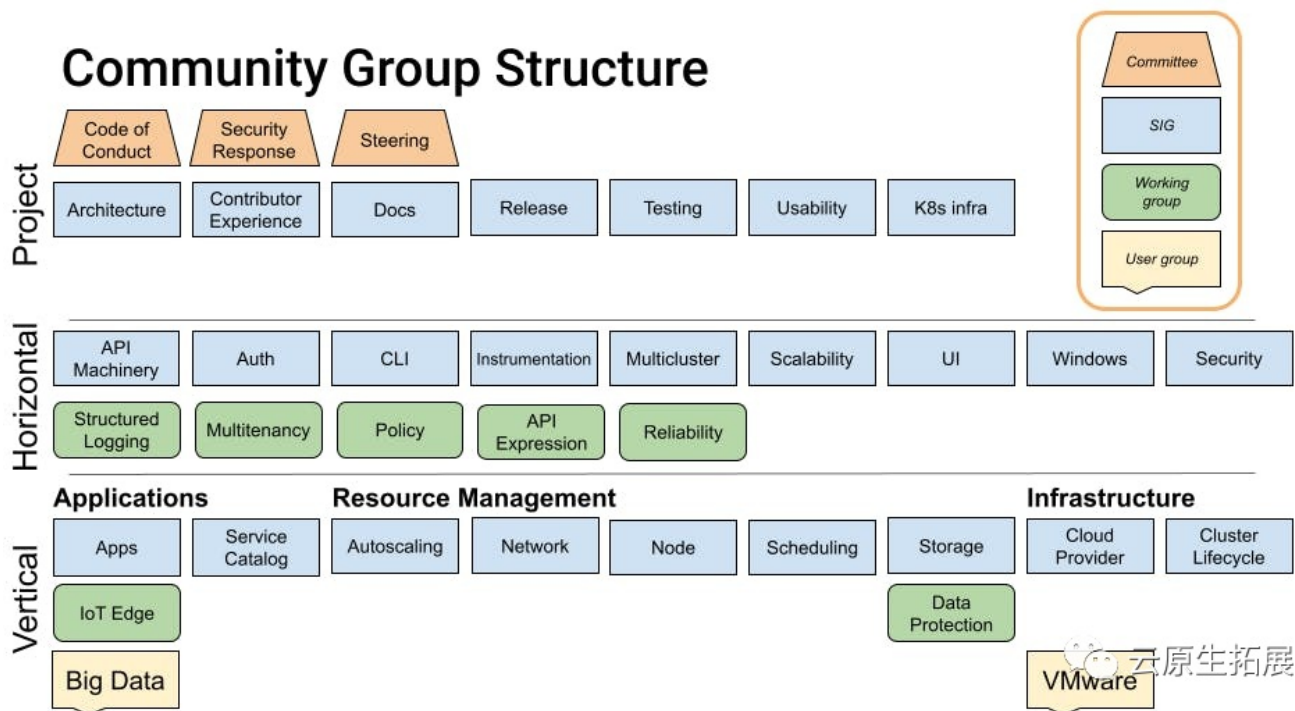
整个K8s治理结构分为4部分:

1. SIGs: SIG 的任务是在CLI、网络等特定主题上推进k8s。整个代码库被划分为这些SIG，每个SIG都有一个管理主体。例 `kubectl` 属于 `sig cli`。SIG 进一步分为更多类型

- 纵向: Network, Storage, Node, Scheduling (特别关注)
- 横向: 可伸缩性、体系结构(项目级关注点)
- 项目: 测试文档 项目管理 贡献值体验

所以，如果你想为 Kubernetes 上工作，你将会在SIG下工作，选择适合你的位置。

1. 工作组: 设立工作组以讨论涉及不同 SIG 的某些主题
2. 用户组: 设立讨论可能没有明确交付物但对大量用户具有相关性的主题的机制。UG 可以与 SIG 合作，以解决可用性问题或改进文档。
3. 委员会: 围绕需要谨慎的主题设立。与 SIG 相比，这些不在公开场合工作，并且涉及社区成员的开戒。



这是在 [kubernetes/community](https://kubernetes.io/community/) 存储库中可用的SIG图。强烈建议您访问这个存储库，以了解更多关于贡献k8的信息，并浏览可用的sig。

SIGs

选择 SIG

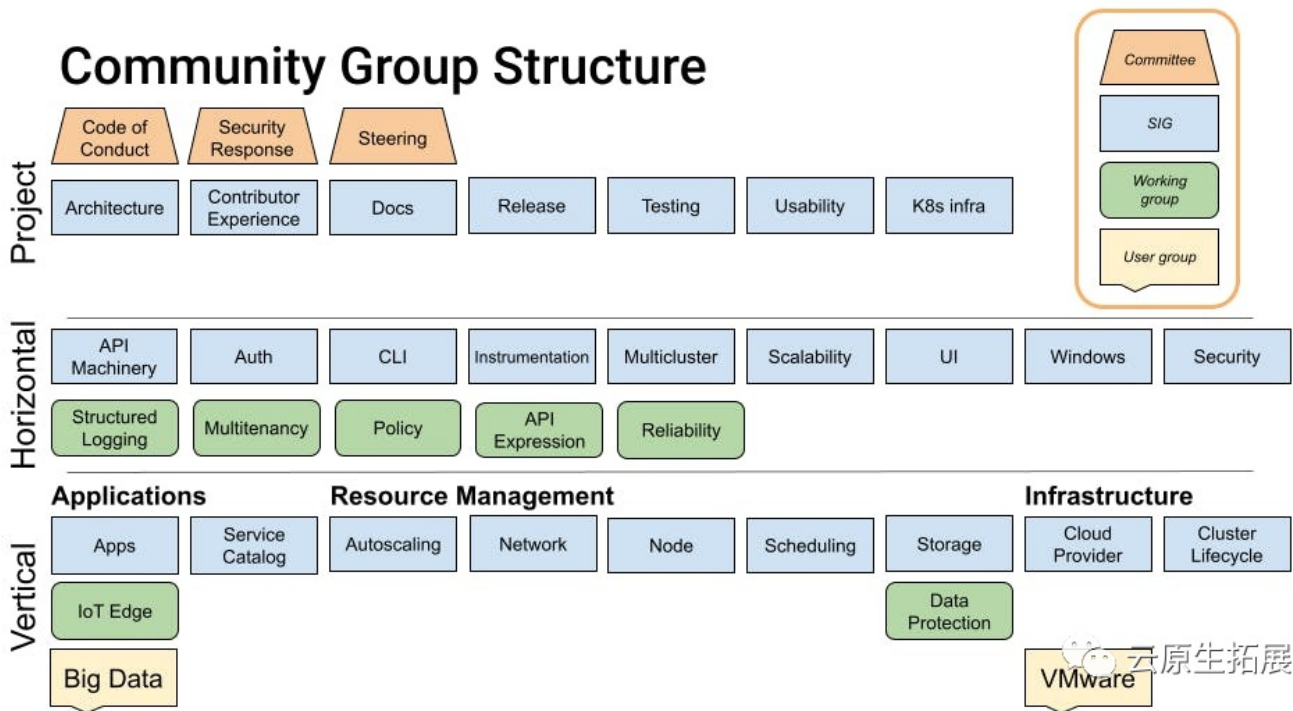
您必须了解不同的社区团体，并了解您必须选择一个团体才能开始。你是怎么做到的?以下是一些建议:

- 选择一个您感兴趣的SIG。在Kubernetes下有很多不同的sig，你一定会找到一个你想要的。如果您不能决定一个SIG，那么尝试找到您想要从事的项目所属的SIG。那将会是你的SIG。如果你仍然困惑，加入 [#sig-contribex](#)，并提及你的兴趣和你想从事的工作。他们在那里帮助贡献者。
- 请查看sig中的文档。浏览可用的文档 (<https://github.com/kubernetes/community>)，看看哪些SIG文档可以帮助您清楚地理解SIG是关于什么的。这包括明确的期望，以及在每个阶段期望你做什么工作。特定于SIG的代码在内部是如何结构的。不同标签的正确定义。明确的会议时间表。我还可以继续并下去，但你们已经明白了。

当你选择一个团体，了解团体成员。在slack频道上介绍自己，试着参加会议，如果不可能的话，参考会议记录，加入邮件列表，总之，试着玩得开心。

旁注:可用内容的数量可能令人难以承受。你如何克服这种感觉?潜伏在周围，看看正在进行的对话。SIG成员可能正在处理任务。试着在那边帮帮忙。告诉他们你是一个初学者，你对正在进行的工作很感兴趣，想和他们一起工作。参加会议，试着了解正在发生的事情。甚至可以观察一下谁在做什么。所有这些都可以帮助您大致了解SIG。

Community Group Structure



参加 SIG

检出 docs 文件，我不能强调这有多重要。在为任何系统做贡献之前，请先阅读文档。这些文档可以帮助回答您可能遇到的大多数问题，并提供大量信息，这些信息将真正帮助初学者快速了解正在发生的事情以及事情是如何结构的。

在开始一个特性之前，试着理解它是做什么的。访问用户友好的文档 (<https://kubernetes.io/docs/home/>)，并尝试了解该功能是如何工作的，以及特定用户如何与它交互。你也可以跟着教程在本地尝试一下。这真的有助于你适应上述功能。

一旦你理解了一个特性，试着去阅读贡献者文档。开发README文件有很多您作为贡献者需要的信息。它有关于了解github工作流程，设置您的开发环境，管理测试，编码约定等信息。如果你是初学者，那么这可能是你开始的好地方。

自我介绍并提出问题。说你想要处理某个SIG的问题，并在遇到困难时提出问题。如果你在一个特定的问题上卡住了，那就提一下你是如何解决这个问题的，以及你认为可能的解决方案是什么(你不必提供确切的解决方案，只要提一些你认为可能有效的方法就可以了)。同样，把这作为提出好问题的一般建议。

旁注:如果您想开始使用自己选择的语言，那么您可以考虑为不同的kubernetes-client存储库做出贡献。例:python客户端

(<https://github.com/kubernetes-client/python>)。更多信息在这里:客户端库 ([<https://kubernetes.io/docs/reference/using-api/client-libraries/>])(<https://kubernetes.io/docs/reference/using-api/client-libraries/>)

处理 issues

您在工作时发现了一个bug。试着自己修复一下。尝试添加更好的注释来解释代码块。这或多或少是一个一般性的建议。一些K8s的具体建议:

标签

检查所有问题的标签。如果问题有“good first issue”或“help wanted”标签，这意味着它是为初学者准备的。在处理这个问题时，你可以期待得到帮助和指导。

也有SIG特定的标签，如 `sig/cli`，`sig/testing` 等，表明一个问题属于特定的SIG。还有一些特定的标签，如种类 `kind/bug`，`kind/feature`，以帮助您了解您正在查看的内容。当你寻找问题时，试着搜索这样的标签。

参与路线图规划

每个SIG都有自己的路线图来决定当前发布周期中谁将负责哪个特性。计划是根据有多少人愿意工作来制定的。一个人不需要成为专家才能开始。你只需要致力于你承诺投入的工作。

| 跟随某人

发布团队有明确的角色。每个角色在整个发布周期中都有一位负责人，负责人会对他们所在团队的跟随者进行导师引导。有明确的手册可供您了解指定的任务。这是进入 Kubernetes 贡献的好方法，您还可以成为下一个发布周期的领导。在这里 (<https://github.com/kubernetes/sig-release/blob/master/release-team/shadows.md>) 获取更多信息

| 非代码贡献

您不必必须对代码库做出贡献。文档总是可以改进的。如果您认为某些文档缺少关键信息或文档可以更好地排版，请继续创建问题并进行必要的修改。您还可以为 K8s 贡献者站点 kubernetes.dev 写博客文章。事件组织和管理是您可以帮助的另一个地方。通过主持活动或在社区中帮助管理来帮助在您所在地区宣传 Kubernetes。在这里 (<https://www.kubernetes.dev/docs/guide/non-code-contributions/>) 获取更多信息

| 关于 pr 的技术细节

你会跟 k8s-ci-机器人打交道。它的存在是为了执行检查。你需要在开始工作时签署CLA。然后是审查过程。您需要有/lgtm和/approve标签才能合并pr。在这个文件中，您将能够找到您的SIG的所有者别名，然后您可以在OWNERS_ALIASES文件中找到与您的SIG相关的人，然后请求他们进行审查。如果你想从某某那里得到评论，你可以在pr中输入/cc @xyz。如果你的pr没有得到太多关注，可以尝试使用#pr-reviews slack 频道。

如果您的pr没有被合并，您将能够向下滚动您的pr以找到一个ui，其中详细说明了可能失败的测试或可能丢失的标签。

欢迎关注我的公众号“**云原生拓展**”，原创技术文章第一时间推送。