

# 开源课程期中作业

## 分析一个流行的归档开源项目

### 基本信息

小组成员：秦长久 51215903107, 吴贤佑 51215903108, 刘添然 51215903077

组内分工：

秦长久：数据挖掘与可视化

吴贤佑：信息汇总报告撰写

刘添然：项目基本信息收集

项目序号: 25

项目 id: 83285138

项目名: alibaba/vlayout

### 一、项目的基本背景和发展历程介绍

#### ● 技术类型

VirtualLayout 是一个针对 RecyclerView 的 LayoutManager 扩展, 主要提供一整套布局方案和布局间的组件复用的问题。通过定制化的 LayoutManager, 接管整个 RecyclerView 的布局逻辑; LayoutManager 管理了一系列 LayoutHelper, LayoutHelper 负责具体布局逻辑实现的地方; 每一个 LayoutHelper 负责页面某一个范围内的组件布局; 不同的 LayoutHelper 可以做不同的布局逻辑, 因此可以在一个 RecyclerView 页面里提供异构的布局结构, 这就能比系统自带的 LinearLayoutManager、GridLayoutManager 等提供更加丰富的能力。同时支持扩展 LayoutHelper 来提供更多的布局能力。

#### ● 版本发布历史

共 34 次更新, 其中 longerian 贡献前 24 次 (2017.03.11-2018.05.22), MikeAfc 贡献后 10 次 (2018.06.21-2019.12.16)。

**2017.03.11** 首次上线 **v1.0.2**, 来自 longerian:

1. 兼容 RecyclerView 25.1.0+;
2. 修复 StickyLayoutHelper 的在滚动期间的位置;
3. 修复 notifyDataSetChanged()被'DelegateAdapter.Adapte'调用时的崩溃;
4. 默认情况下使 FloatLayoutHelper 中的视图可单击, 从而可以启用拖动;
5. 添加 OnePlusNLayoutHelper 扩展;
6. 修复了 StaggeredGridLayoutHelper 子视图的位置错误。

**2018.5.22** 最后一个由 longerian 发布的版本 **v1.2.15**:

1. 为 PerformanceMonitor 方法添加@Keep。

**2018.6.21** 首次由 MikeAfc 发布的版本 **v1.2.16**:

1. 添加界面控件可以滚动功能。

**2019.12.16** 最后一次更新 **v1.2.36**, 来自 MikeAfc:

- 1.修复 mspan 在 StaggeredGridLayoutHelper 可能为 null 的异常;
- 2.修复 sticky listener 和空指针异常。

- **主要贡献者的构成 (国家、区域和组织等)**

前期: longerian (洪晓龙), 位于杭州的阿里工程师。

后期: MikeAfc, 皮卡丘爱好者, 可能是泰国艺人迈克·安吉洛的粉丝小号。

- **CI/CD 的使用**

并未使用

- **其他有价值的信息**

## **主要功能**

- 默认通用布局实现, 解耦所有的 View 和布局之间的关系: Linear, Grid, 吸顶, 浮动, 固定位置等。
  - LinearLayoutHelper: 线性布局
  - GridLayoutHelper: Grid 布局, 支持横向的 colspan
  - FixLayoutHelper: 固定布局, 始终在屏幕固定位置显示
  - ScrollFixLayoutHelper: 固定布局, 但之后当页面滑动到该图片区域才显示, 可以用来做返回顶部或其他书签等

- FloatLayoutHelper: 浮动布局, 可以固定显示在屏幕上, 但用户可以拖拽其位置
- ColumnLayoutHelper: 栏格布局, 都布局在一排, 可以配置不同列之间的宽度比值
- SingleLayoutHelper: 通栏布局, 只会显示一个组件 View
- OnePlusNLayoutHelper: 一拖 N 布局, 可以配置 1-5 个子元素
- StickyLayoutHelper: stikcy 布局, 可以配置吸顶或者吸底
- StaggeredGridLayoutHelper: 瀑布流布局, 可配置间隔高度/宽度
- 上述默认实现里可以大致分为两类: 一是非 fix 类型布局, 像线性、Grid、栏格等, 它们的特点是布局在整个页面流里, 随页面滚动而滚动; 另一类就是 fix 类型的布局, 它们的子节点往往不随页面滚动而滚动。
- 所有除布局外的组件复用, VirtualLayout 将用来管理大的模块布局组合, 扩展了 RecyclerView, 使得同一 RecyclerView 内的组件可以复用, 减少 View 的创建和销毁过程。

## 相关软件

Tangram 具备 vlayout 里所有的功能, 且隐藏了细节, 通过数据配置即可搭建页面, 能避免绝大多数 Issue 里提到的问题, 而且重大更新维护主要基于 Tangram, 包括局部刷新、响应式接口等。

Tangram 是一套动态化构建 Native 页面的框架, 它包含 Tangram Android、Tangram iOS, 管理后台等一些列基础设施。Tangram 不仅仅是一个 Native (iOS & Android) 的界面开发框架, 而是从日常工作中沉淀出的一套界面解决方案, 涵盖了 Native SDK, GUI 操作台, 后端逻辑容器, 组件库机制的一整套方案。

Tangram 从手机天猫 - 首页方案抽象而来, 是面向组件的界面方案, 是我们不断权衡性能、稳定性、开发效率、灵活性和动态性多方面表现的结果。除了手机天猫首页外, 还支撑了天猫 App 中的天猫直播, 我的天猫, 猜你喜欢等多个业务, 并且在阿里星球等多个阿里系 App 中有所应用。

就如 Tangram 主页所说, 我们重点关注方案的多平台一致性, 高性能和业务支撑能力。

## 二、项目的历史轨迹分析

### 1. 每月新增 Star 和 Fork 的个数



表 1 每月新增 Star 数量

时间	数量	时间	数量	时间	数量	时间	数量
2017-02	51	2018-06	134	2019-10	85	2021-02	20
2017-03	2939	2018-07	148	2019-11	101	2021-03	39
2017-04	542	2018-08	151	2019-12	95	2021-04	51
2017-05	1126	2018-09	120	2020-01	50	2021-05	32
2017-06	732	2018-10	114	2020-02	42	2021-06	29
2017-07	360	2018-11	128	2020-03	73	2021-07	16
2017-08	324	2018-12	112	2020-04	83	2021-08	12
2017-09	222	2019-01	105	2020-05	91	2021-09	11
2017-10	220	2019-02	121	2020-06	63	2021-10	8
2017-11	522	2019-03	190	2020-07	66	2021-11	12
2017-12	262	2019-04	158	2020-08	53	2021-12	19
2018-01	239	2019-05	146	2020-09	51	2022-01	14
2018-02	181	2019-06	105	2020-10	48	2022-02	12
2018-03	256	2019-07	115	2020-11	58	2022-03	23
2018-04	151	2019-08	106	2020-12	47	2022-04	10
2018-05	164	2019-09	106	2021-01	28		

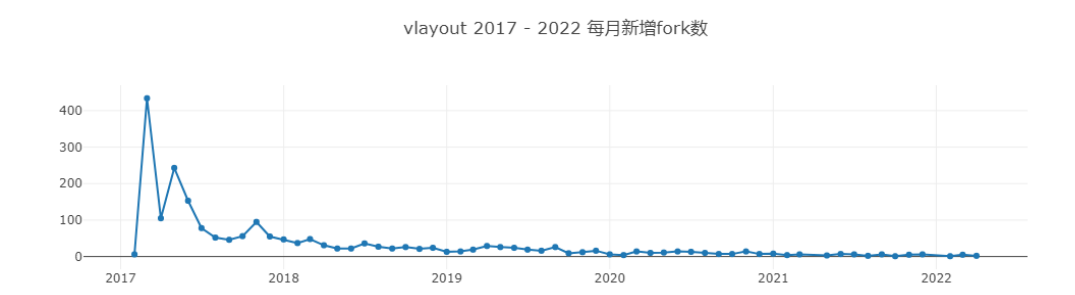


表 2 每月新增 Fork 数量

时间	数量	时间	数量	时间	数量	时间	数量
2017-02	6	2018-06	22	2019-10	9	2021-02	4
2017-03	434	2018-07	36	2019-11	12	2021-03	6
2017-04	105	2018-08	27	2019-12	16	2021-05	3
2017-05	243	2018-09	22	2020-01	6	2021-06	7
2017-06	153	2018-10	26	2020-02	4	2021-07	6
2017-07	78	2018-11	21	2020-03	14	2021-08	2
2017-08	52	2018-12	24	2020-04	10	2021-09	6
2017-09	46	2019-01	13	2020-05	11	2021-10	1
2017-10	56	2019-02	14	2020-06	14	2021-11	5
2017-11	95	2019-03	19	2020-07	13	2021-12	6
2017-12	55	2019-04	29	2020-08	10	2022-02	1
2018-01	47	2019-05	26	2020-09	7	2022-03	5
2018-02	37	2019-06	24	2020-10	7	2022-04	2
2018-03	48	2019-07	19	2020-11	14		
2018-04	31	2019-08	16	2020-12	7		
2018-05	22	2019-09	26	2021-01	8		

## 2. 每月打开 Issue 和 关闭 Issue 的个数

vlayout 2017 - 2022 每月issue打开的个数

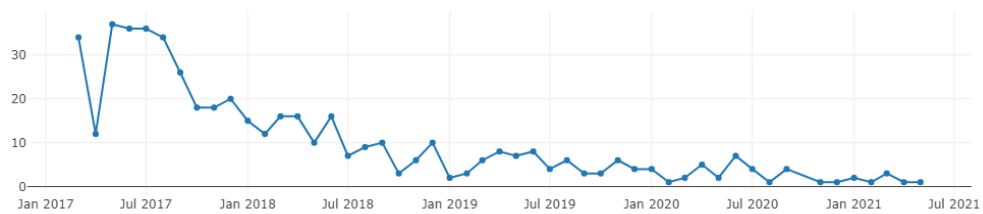


表 3 每月打开 Issue 数量

时间	数量	时间	数量	时间	数量	时间	数量
2017-03	34	2018-04	16	2019-05	7	2020-06	7
2017-04	12	2018-05	10	2019-06	8	2020-07	4
2017-05	37	2018-06	16	2019-07	4	2020-08	1
2017-06	36	2018-07	7	2019-08	6	2020-09	4
2017-07	36	2018-08	9	2019-09	3	2020-11	1
2017-08	34	2018-09	10	2019-10	3	2020-12	1
2017-09	26	2018-10	3	2019-11	6	2021-01	2
2017-10	18	2018-11	6	2019-12	4	2021-02	1
2017-11	18	2018-12	10	2020-01	4	2021-03	3
2017-12	20	2019-01	2	2020-02	1	2021-04	1
2018-01	15	2019-02	3	2020-03	2	2021-05	1

2018-02	12	2019-03	6	2020-04	5		
2018-03	16	2019-04	8	2020-05	2		

vlayout 2017 - 2022 每月issue关闭的个数

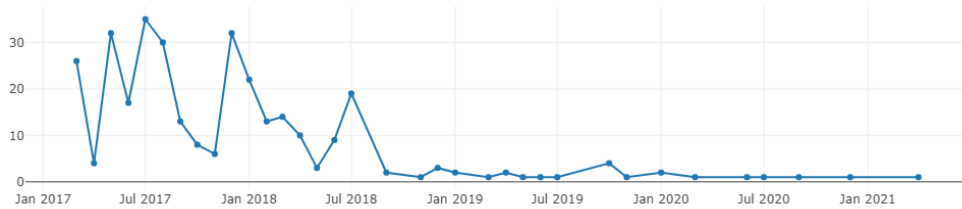


表 4 每月关闭 Issue 数量

时间	数量	时间	数量	时间	数量	时间	数量
2017-03	26	2017-12	32	2018-11	1	2019-11	1
2017-04	4	2018-01	22	2018-12	3	2020-01	2
2017-05	32	2018-02	13	2019-01	2	2020-03	1
2017-06	17	2018-03	14	2019-03	1	2020-06	1
2017-07	35	2018-04	10	2019-04	2	2020-07	1
2017-08	30	2018-05	3	2019-05	1	2020-09	1
2017-09	13	2018-06	9	2019-06	1	2020-12	1
2017-10	8	2018-07	19	2019-07	1	2021-04	1
2017-11	6	2018-09	2	2019-10	4		

### 3. 每月打开 PR 和合入 PR 的个数

vlayout 2017 - 2022 每月打开的PR的个数

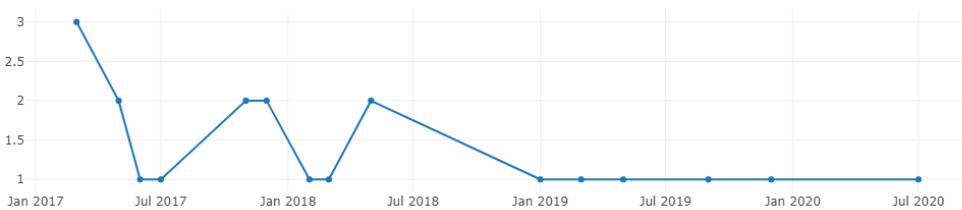


表 5 每月打开 PR 的数量

时间	数量	时间	数量	时间	数量	时间	数量
2017-03	3	2017-11	2	2018-05	2	2019-09	1
2017-05	2	2017-12	2	2019-01	1	2019-12	1
2017-06	1	2018-02	1	2019-03	1	2020-07	1
2017-07	1	2018-03	1	2019-05	1		

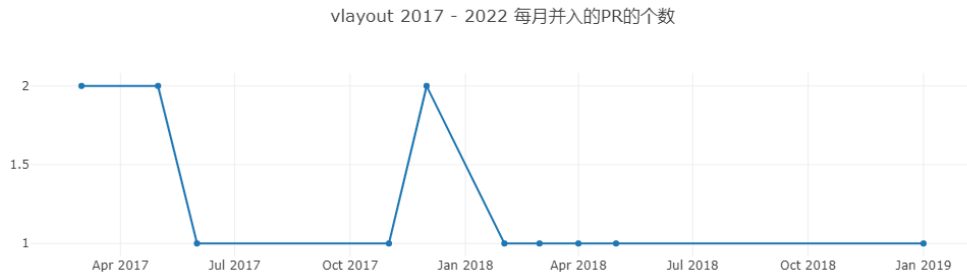


表 6 每月合并 PR 的数量

时间	数量	时间	数量	时间	数量	时间	数量
2017-03	2	2017-11	1	2018-03	1	2019-01	1
2017-05	2	2017-12	2	2018-04	1		
2017-06	1	2018-02	1	2018-05	1		

#### 4. 每月在仓库中活跃的不同开发者总数



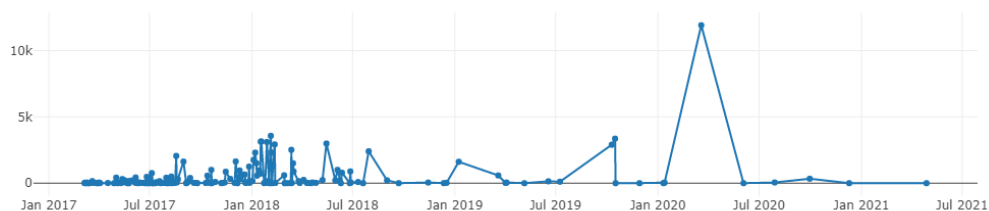
表 7 每月在仓库中活跃的不同开发者的数量

时间	数量	时间	数量	时间	数量	时间	数量
2017-03	34	2018-04	18	2019-05	20	2020-06	7
2017-04	12	2018-05	20	2019-06	13	2020-07	8
2017-05	36	2018-06	17	2019-07	10	2020-08	6
2017-06	40	2018-07	10	2019-08	16	2020-09	5
2017-07	40	2018-08	13	2019-09	8	2020-11	2
2017-08	36	2018-09	15	2019-10	4	2020-12	5
2017-09	35	2018-10	13	2019-11	9	2021-01	4
2017-10	26	2018-11	18	2019-12	6	2021-02	1
2017-11	27	2018-12	16	2020-01	5	2021-03	7
2017-12	29	2019-01	12	2020-02	1	2021-04	3
2018-01	28	2019-02	6	2020-03	4	2021-05	3
2018-02	22	2019-03	10	2020-04	10	2021-06	2
2018-03	21	2019-04	10	2020-05	10	2021-07	1

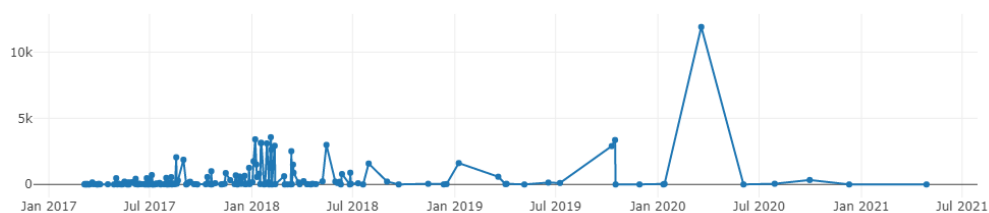
## 5. Issue 从打开到关闭的平均时长和中位数 (单位: 天)

项目创建至归档期间 Issue 从打开到关闭的时长中位数: 2.6 天。

vlayout 2017 - 2022 Issue 从打开到关闭的平均时长(单位: h)



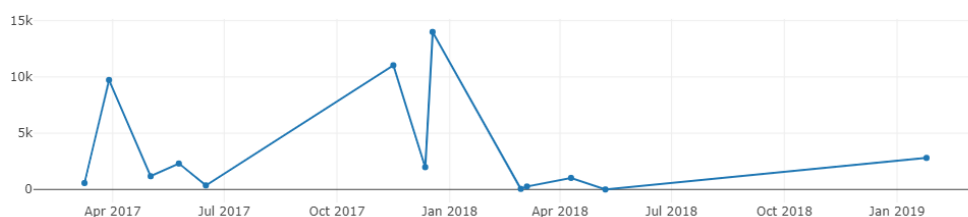
vlayout 2017 - 2022 Issue 从打开到关闭的时长中位数(单位: h)



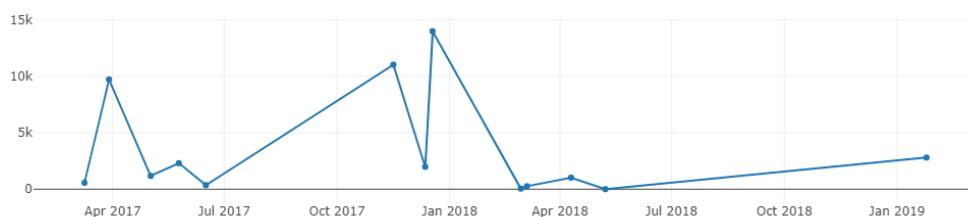
## 6. PR 从打开到合入的平均时长和中位数 (单位: 天)

项目创建至归档期间 PR 从打开到合入的时长中位数: 0.82 天。

vlayout 2017 - 2022 PR从打开到合入的平均时长(单位: minute)



vlayout 2017 - 2022 PR从打开到关闭时长的中位数(单位: minute)



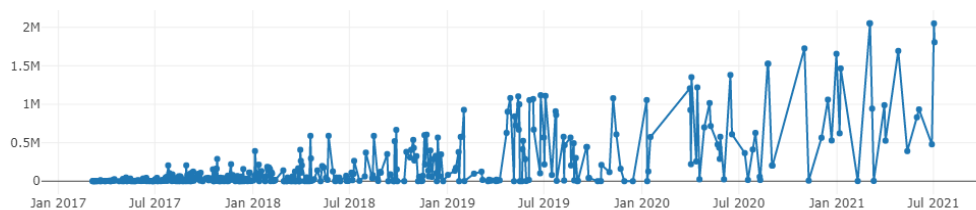


## 7. Issue 和 PR 从打开到第一次有人回复（非本人回复）的平均时长和中位数（单位：天）

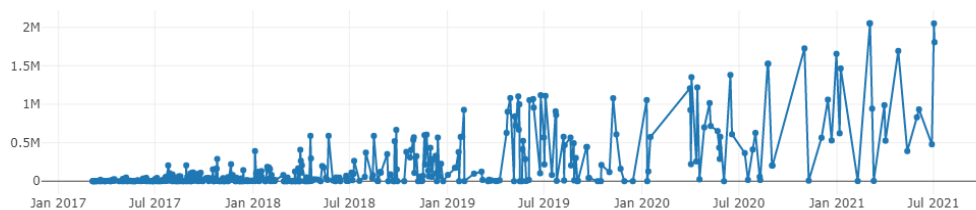
项目创建至归档期间 Issue 从打开到第一次有人回复的时长中位数：5.08 天。

项目创建至归档期间 PR 从打开到第一次有人回复的时长中位数：0.0001 天。

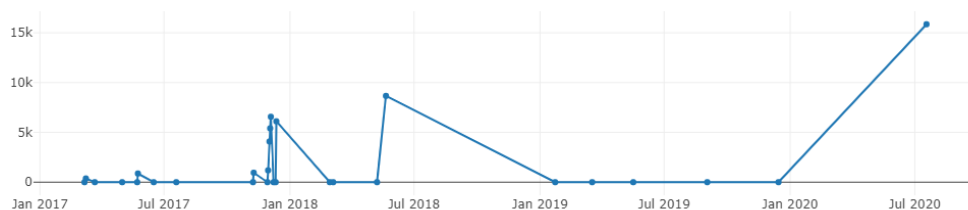
vlayout 2017 - 2022 Issue 从打开到第一次有人回复的平均时长(单位: second)



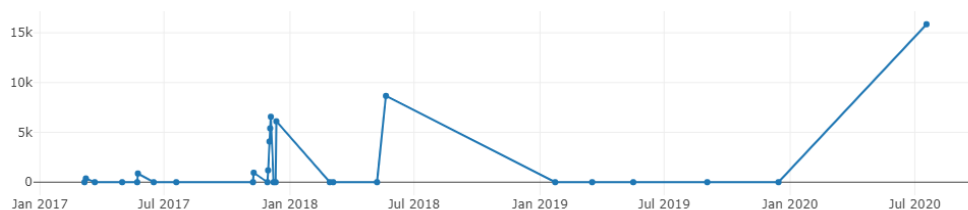
vlayout 2017 - 2022 Issue 从打开到第一次有人回复的时长中位数(单位: minute)



vlayout 2017 - 2022 PR 从打开到第一次有人回复的平均时长(单位: minute)



vlayout 2017 - 2022 PR 从打开到第一次有人回复时长中位数(单位: minute)



## 8. 根据你观察到的仓库的历史数据，尝试找到几个你认为关键或值得注意的时间节点

2017.3.11：第一个版本 v1.0.2 发布。

2017.3-2017.9：2017 年 3 月项目第一次发布后即刻引起了广泛关注。3-6 月期间项目热度高，star 和 fork 的数量都位于峰值，持续吸引用户及开发者。3-9 月期间开发者活跃度位于峰值。

2017.11：项目关注度出现回暖。

2018.5.22：longerian 最后一次发布版本。

2018.6.21：MIkeAfc 首次发布版本。

2019.12.16：MIkeAfc 最后一次发布版本。

2020.5.14：最后一次 commit。

2021.7.14：项目归档。

## 三、洞察项目被归档的可能原因

- 阅读分析项目的相关信息：

(此块具体内容分布在全文中)

- 主页、主要贡献者发表的相关技术博客

主页：<https://github.com/alibaba/vlayout>

主要贡献者相关技术博客：<http://pingguohe.net/>

<https://www.jianshu.com/u/a4c40b057676>

- Issue 和 PR 中的相关讨论

longerian 于 2017 年 10 月 31 日发布了一则工作小组招募的 Issue。

- README 文件，贡献者指南，Code of Conduct 及其他可能有的相关文档

README 文件：

<https://github.com/alibaba/vlayout/blob/master/README-ch.md>

贡献者指南：

<https://github.com/alibaba/vlayout/blob/master/CONTRIBUTING.md>

布局属性文档：

<https://github.com/alibaba/vlayout/blob/master/docs/ATTRIBUTES-ch.md>

FAQ:

<https://github.com/alibaba/vlayout/blob/master/docs/VLayoutFAQ.md>

- **结合一和二中得到的信息和分析结果，尝试总结项目可能的归档原因**

#### **新工具的出现：**

虽然 vlayout 布局灵活，然而 API 相对原始，手工维护数据及 LayoutHelper 比较麻烦，项目管理者强烈建议大家使用 Tangram-Android 来间接使用 vlayout，Tangram 具备 vlayout 里所有的功能，且隐藏了细节，通过数据配置即可搭建页面，能避免绝大多数 Issue 里提到的问题，而且重大更新维护主要基于 Tangram，包括局部刷新、响应式接口等。longerian 2018 年的工作重心似乎转向了 Tangram，由此减少了对 vlayout 的更新和维护。更进一步，我们猜测该项目最终被归档可能是因为出现了某些更加优越的新工具。

#### **开发者离职：**

项目创建者 longerian 于 2018 年 5 月 22 日发布了版本 v1.2.15 后就不再对项目有所贡献，直至 2021 年 7 月 14 日，他对 README 文件更新了项目不再维护的说明，应该同时进行了项目归档。在创建者 longerian 维护的博客“苹果核”的“关于”中可以了解到，他于 2018 年秋天离开了移动开发，这与他停止对 vlayout 项目进行维护的时间点相近。由此我们推测，开发者从原岗位离职是他停止维护的主要原因。

MikeAfc 在网络上足迹不多，但根据 longerian 仓库中几个关联性较强的项目中 MikeAfc 均作为第二贡献者的情况推测，我们认为 MikeAfc 可能是 longerian 在阿里的同事。进一步推测，MikeAfc 停止对项目进行维护可能是因为公司里的工作任务调整，或者是同 longerian 一样离开了原岗位。

#### **业务需求下降：**

搜索发现，在项目近乎停止更新之后，网络上出现了一些对于 vlayout 的使用总结，我们认为这说明项目热度虽然已经冷却，但是使用者对于该项目仍是认可的态度，或许是因为该项目已经比较完整，能够适用大部分的使用情况。对此我们推测，对于一个较为完善的项目，在没有足够的新需求的情况下，开发者及其背后的组织便不再有足够的动力去持续维护。

- **结合你搜集到的信息，尝试分析项目归档后可能产生的影响（对开发者和用户）**

**对于开发者：**

一旦项目被归档，将无法添加或删除协作者或团队，其 Issue、PR、代码、标签、里程碑、项目、wiki、发布、提交、标签、分支、reaction、代码扫描警报、评论和权限都将变为只读。但仍可以对项目进行 star 或 fork，允许项目在其他地方继续开发。

通过项目归档，项目管理者不再接受任何更新，可以把精力投入到新的地方。但旧项目并非没有了价值，归档保存了开发过程中的资料，若日后有需要仍可以将项目解归档继续开发。

**对于用户：**

归档后的项目意味着该项目将被停止维护，不再接受新的需求，不再修补 bug，不再解答使用中的问题，这会导致用户的体验感不会再被优化。用户仍可以继续使用该项目提供的功能，但是随着时间流逝，停滞不前的工具终会被淘汰，因此用户需要寻找其他替代品。又或许旧项目被归档是因为产生了某些更加优秀的替代品，归档的项目可以将用户引导至更加完善的产品，助力新产品的发展。

- **表述你对开源项目如何可持续发展的理解**

**合作开发者数量：**

开源项目想要可持续发展，需要有一定的合作开发者数量。本项目只有两位主要开发者，当主要开发者离职或者因为某些原因停止对项目进行贡献时，容易出现后继无人的现象。可能是因为项目的规模，或是因为公司工作任务安排的原因，项目初始成员的数量有限，那么如何从社区吸引开发者为项目提供贡献则显得重要。有开发者数量作保障就能更好地避免一些个人原因导致的问题，源源不断地有人愿意为项目做出贡献。

**影响力的维持：**

开源项目的影响力是吸引开发者或者开发者背后的组织愿意持续为项目付出的保障。有影响力的项目容易吸引到众多用户和开发者参与其中，用户的反馈有助于开发者有针对性地对项目进行优化，开发者则可以从自己的贡献中获得成就感，形成良好的氛围。影响力越大越容易吸引人们自发地推广该项目，例如撰写个人博客或者在各平台发文介绍，形成正反馈，进一步拓展项目影响力。若无法保障影响力，则容易出现昙花一现的场面。

**文档的构建：**

开源项目如果希望能够长久的发展，就需要构建足够详细的文档。随着时间推进，不可避免地出现开发者的交替变更，优秀的文档可以帮助后继的开发者优先地接手项目的管理工作。更加丰富的文档也有助于帮助用户减少使用上的困难，改进使用体验，进而增加其自发推广的潜在可能。