

开源课程期末作业

归档开源项目名称	10391073-openedx/edx-platform
小组成员	刘颖凡 51215903009
	魏如蓝 51215903002
	王原昭 51215903074

一、项目的基本背景和发展历程介绍

1.1 项目基本情况

- 项目openedx/edx-platform是github中一个高度流行的开源项目，自2011年12月开始创建维护
- openedx是一个集合多种机器学习资源、分布式、负载均衡等最新技术的一个开源平台。平台旨在展示最新的学习科学和教学设计，由开发人员、技术提供商、研究团队和用户社区推动发展。
- openedx可以通过ubuntu、docker、django、react来构建，并可以运行在amazon、google、IBM、Azure等云服务平台上。它的开放式学习管理系统（LMS）可以为学习者、课程团队和教师在整个课程期间提供数据支持。包含可以发布课程、发布讨论板、管理团队和群组、编辑成绩以及 与学习者交流的教师视图；可以通过他们的学习者仪表板访问他们的个人资料、检查他们的注册状态并浏览课程的学习者视图。LMS 可用于桌面、iOS 和 Android 版本，提供对课程内容和支持基础设施（时间表、讨论板、协作工具、学生管理、证书生成、消息传递等）的访问，同样支持在线课程、在线校园和在线学位课程，并且可根据学习者特质定制独特的学习体验。
- 此外，Open edX Insights 提供以学习者为中心的智能分析，帮助教师了解学习者如何参与课程材料。内置报告可帮助教师和课程团队管理任何规模的群组的在线内容。学习团队可以利用现有数据构建自定义自适应内容并将数据存储在仓库或学习记录存储 (LRS) 中。
- openedx的开发语言构成如下：
 - Python 54.1%
 - JavaScript 28.2%
 - HTML 14.2%
 - SCSS 3.0%
 - CSS 0.4%
 - Shell 0.1%

1.2 版本发布历史

- openedx共发布了14个版本：
 - 第一个版本发布于2014-10-28 (Aspen)
 - 最后一个版本发布于 2022-04-12 (Nutmeg)

1.3 主要贡献者

项目参与者众多，主要参与者（commit次数大于100）有如下9位：

- **cpennington**
 - 国家或地区：Arlington, MA
 - 所属公司或组织：Reify Health
 - 该作者主要参与openedx的早期工作，主要贡献在2012年至2015年之间，并持续贡献至2021年，共有2219次commit，是对openedx贡献排名度第一的作者。该作者目前是Reify Health公司的staff engineer，虽然2021年之后活跃度明显降低，但至今仍在github活动。
- **VikParuchuri**
 - 国家或地区：San Francisco, CA
 - 所属公司或组织：dataquestio
 - 该作者是openedx早期提交最多的作者（900次commit），全部集中于2013年。目前仍活跃在github，最近一次github提交在2022年6月27日，主要工作用Python完成。
- **talbs**
 - 国家或地区：Boston, MA
 - 所属公司或组织：FortAwesome
 - 该作者是openedx中期的主要贡献者，主要贡献在2013年至2015年之间，至今仍活跃在github，过去的一年中完成了556次contributions，主要工作用JavaScript和Python完成。
- **nedbat**
 - 国家或地区：Boston
 - 所属公司或组织：2U/edX on Open edX.
 - nedbat是一名python开发者和社区组织者，该作者从2013年开始参与对openedx项目的贡献，一直持续到现在，仍然在为openedx工作，主要负责2U/edX的部分，至今仍活跃在github，过去的一年中完成了3484次contributions，主要工作用Python完成。

1.4 CI/CD的使用

- **CI (Continuous integration, 持续集成)**
 - 在持续集成环境中，开发人员将会频繁的提交代码到主干。这些新提交在最终合并到主线之前，都需要通过编译和自动化测试流进行验证。这样做是基于之前持续集成过程中很重视自动化测试验证结果，以保障所有的提交在合并主线之后的质量问题，对可能出现的一些问题进行预警。
- **CD (持续集成/持续交付)**
 - 持续交付（Continuous delivery）指的是，频繁地将软件的新版本，交付给质量团队或者用户，以供评审。如果评审通过，代码就进入生产阶段。持续交付可以看作持续集成的下一步。它强调的是，不管怎么更新，软件是随时随地可以交付的。
 - 持续部署（continuous deployment）是持续交付的下一步，指的是代码通过评审以后，自动部署到生产环境。持续部署的目标是，代码在任何时刻都是可部署的，可以进入生产阶段。
- **GitHub Actions**
 - 本项目使用Github Actions。Github Actions是由Github创建的 CI/CD服务。它的目的是使所有软件开发工作流程的自动化变得容易。直接从GitHub构建，测试和部署代码。CI（持续集成）由很多操作组成，比如代码合并、运行测试、登录远程服务器，发布到第三方服务等。GitHub 把这些操作就称为 actions。
 - GitHub Actions 的配置文件叫做 workflow 文件，可以在openedx代码仓库中的 `.github/workflows` 目录中看到多个以.yml结尾的配置文件，选取最近更新的static-assets-check.yml，其内容如下：

```

name: static assets check for lms and cms

on:
  pull_request:
  push:
    branches:
      - master

jobs:
  static_assets_check:
    name: static-assets-check
    runs-on: ${{ matrix.os }}
    strategy:
      matrix:
        os: [ ubuntu-20.04 ]
        python-version: [ 3.8 ]
        node-version: [ 16 ]
        npm-version: [ 8.5.x ]

    steps:
      - name: Checkout repo
        uses: actions/checkout@v2

      - name: Setup Python ${{ matrix.python-version }}
        uses: actions/setup-python@v2
        with:
          python-version: ${{ matrix.python-version }}

      - name: Setup Node
        uses: actions/setup-node@v2
        with:
          node-version: ${{ matrix.node-version }}

      - name: Setup npm
        run: npm i -g npm@${{ matrix.npm-version }}

      - name: Get pip cache dir
        id: pip-cache-dir
        run: |
          echo "::set-output name=dir::$(pip cache dir)"

      - name: Cache pip dependencies
        id: cache-dependencies
        uses: actions/cache@v2
        with:
          path: ${{ steps.pip-cache-dir.outputs.dir }}
          key: ${{ runner.os }}-pip-${{
hashFiles('requirements/edx/development.txt') }}
          restore-keys: ${{ runner.os }}-pip-

      - name: Install Required Python Dependencies
        run: |
          pip install -r requirements/pip.txt
          pip install -r requirements/edx/base.txt

      - name: Initiate Mongo DB Service
        run: sudo systemctl start mongod

```

```
- name: Add node_modules bin to $Path
  run: echo $GITHUB_WORKSPACE/node_modules/.bin >> $GITHUB_PATH

- name: Run Static Assets Check
  env:
    LMS_CFG: lms/envs/bok_choy.yml
    CMS_CFG: cms/envs/bok_choy.yml

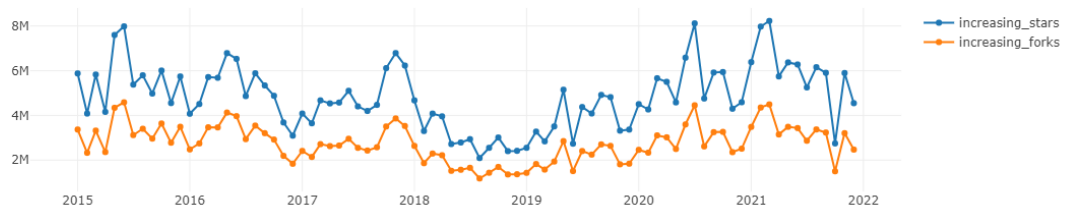
  run: |
    paver update_assets lms
    paver update_assets cms
```

- 通过配置文件内容可知，workflow的触发条件是push (master) , pull_request, 并定义了static_assets_check job, 运行在ubuntu系统上。

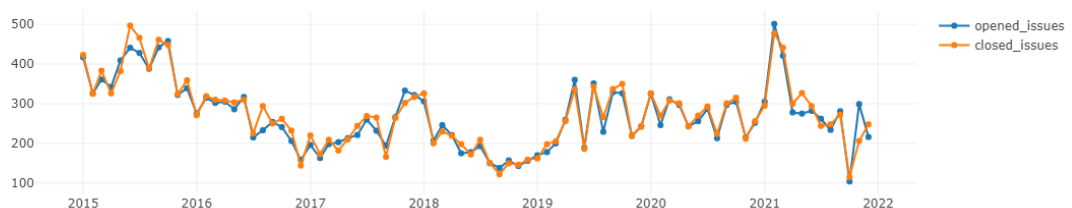
二、项目的历史轨迹分析

- 关于项目的具体数据分析详见 10391073-edx-platform.ipynb 文件
- 可视化结果展示:

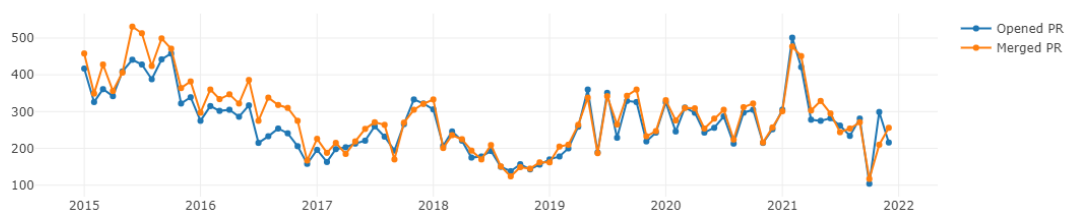
openedx/edx-platform: Increasing Stars/Forks by Month (2015 - 2021)



openedx/edx-platform: Opened/Closed Issues by Month (2015 - 2021)



openedx/edx-platform: Opened/Merged PRs by Month (2015 - 2021)



openedx/edx-platform: Active Developers by month (2015 - 2021)



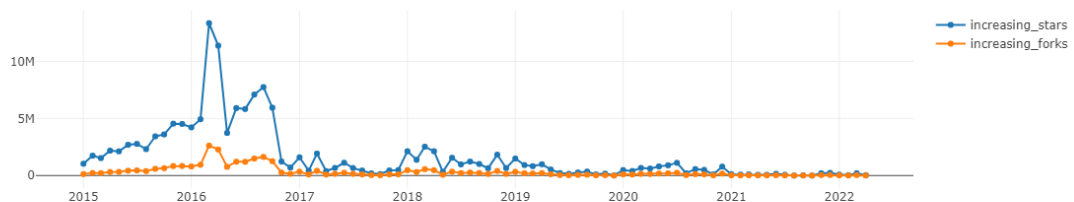
- 几个重要的时间节点：
 - 从上述可视化结果可以发现，openedx/edx-platform项目从发起之初到至今，始终保持一个较高的活跃水平。
 - 开源项目开发者的活跃度与项目本身的关注度和活跃度有很大的关系：本项目在2015年上半年、2020-2021年期间的活跃度日益增加，相应的活跃的开发者的数量也呈上升趋势。
 - 项目在2020.1-2021.12年期间受到较多的关注，期间发布了三个版本的更新（Open edX Koa Release & Open edX Lilac Release & Open edX Maple Release），项目与日俱增的 issues/PRs也获得了及时的反馈和修改。

三、针对活跃项目和归档项目的分析与思考

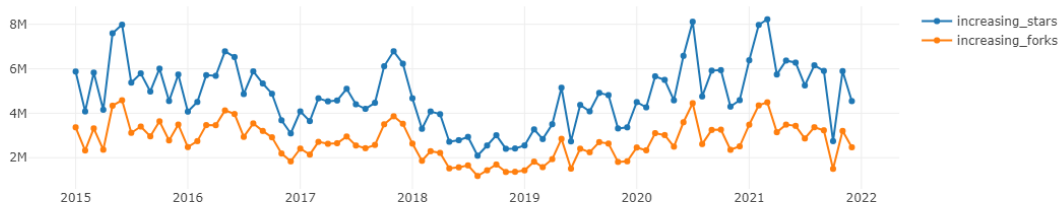
3.1 项目基础数据变化趋势的对比

- 每月新增 Star 和 Fork 的个数

angular/material: Increasing Stars/Forks by Month (2015 - 2022)



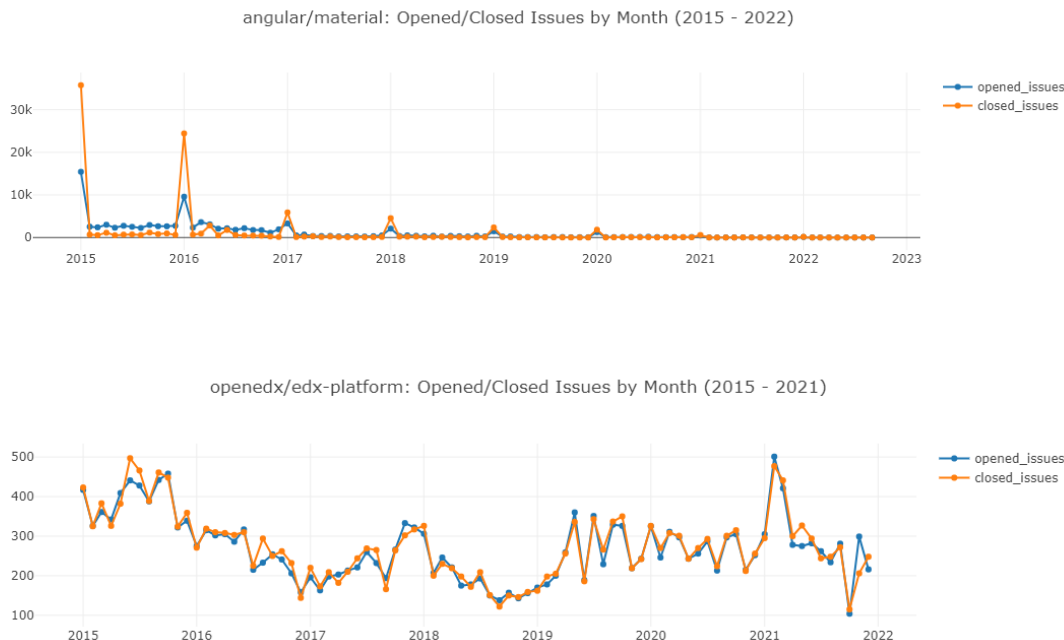
openedx/edx-platform: Increasing Stars/Forks by Month (2015 - 2021)



- 上图为归档项目 `angular/material`，下图为活跃项目 `openedx/edx-platform`。
- 从“每月新增 Star /Fork 数”变化趋势图可以看到，`angular/material` 的获得关注的程度在项目开始的一年时间内（即2015-2016年期间）呈上升趋势，2016-2017年期间活跃度日益下降，从2017-至今基本趋于个位数，**越来越低的关注度**成为该项目被归档的一个直接因素。

- 从“每月新增 Star /Fork 数”变化趋势图可以看到，`openedx/edx-platform` 从项目开始之初到至今，始终维持在一个较高关注度的状态，每月新增关注数均在百万级以上；除了项目刚发起的时间内，在2020-2022年期间，项目的关注度再次呈现一个上升的趋势，甚至多次创造了关注度的峰值，由此可以发现，计算教育领域在近些年里**获得了越来越多的关注**。

- 每月打开 Issue 和 关闭 Issue 的个数



- 上图为归档项目 `angular/material`，下图为活跃项目 `openedx/edx-platform`。
- 从“每月打开 /关闭issues数量”变化趋势图可以看到，`angular/material` 项目在初始阶段有较多的issues，且关闭的issues数量普遍小于打开的issues数量；自2017年之后，该指标逐步趋于个位数，直到项目被归档；而 `openedx/edx-platform` 从项目开始之初到至今，每月的issues数量较为稳定，且关闭的issues数量与打开的issues数量基本持平，说明该项目的贡献者较为活跃，能尽快的处理新提出的issues。
- 从“每月打开 /关闭issues数量”上分析（详见下图），从2015.1-2016.12期间，`angular/material` 项目的每月打开issues数量基本是 `openedx/edx-platform` 项目的十倍左右，而 `angular/material` **每月关闭issues数量要远小于该项目的打开issues数量**，且仅是 `openedx/edx-platform` 项目的三倍左右。
- 从以上分析可以得出：
 - `angular/material` 项目在该指标上虽整体上呈下降趋势，但在项目之初受到的关注度仍是较高的，但**没有得到及时的反馈与处理**，此差异可能是导致项目最终被归档的一个重要因素；
 - `openedx/edx-platform` 项目在该指标上整体上呈上升趋势，单月的具体数值虽只有百位级，但总体上是较为稳定的，说明该项目**有较稳定的开发者社区来维护整个项目的更新迭代**，这也是该项目生命力旺盛的一个重要原因。

angular/material

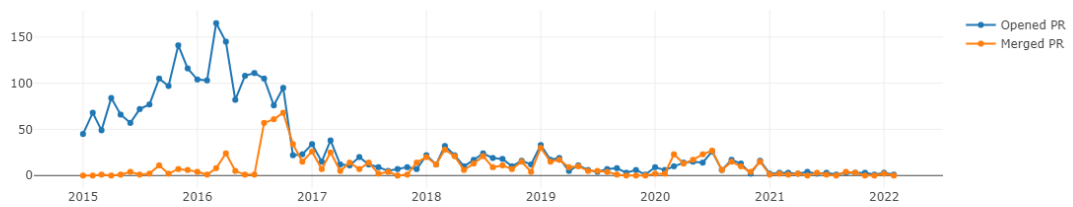
(index)	Time	Opened Issues	Closed Issues
0	'2015-1'	'15421'	'35741'
1	'2015-2'	'2538'	'686'
2	'2015-3'	'2397'	'553'
3	'2015-4'	'3005'	'1120'
4	'2015-5'	'2298'	'527'
5	'2015-6'	'2760'	'691'
6	'2015-7'	'2515'	'733'
7	'2015-8'	'2248'	'579'
8	'2015-9'	'2946'	'1158'
9	'2015-10'	'2653'	'809'
10	'2015-11'	'2637'	'959'
11	'2015-12'	'2749'	'611'
12	'2016-1'	'9583'	'24426'
13	'2016-2'	'2326'	'678'
14	'2016-3'	'3620'	'917'
15	'2016-4'	'3097'	'2825'
16	'2016-5'	'2076'	'494'
17	'2016-6'	'2175'	'1778'
18	'2016-7'	'1816'	'596'
19	'2016-8'	'2194'	'473'
20	'2016-9'	'1766'	'455'
21	'2016-10'	'1737'	'424'
22	'2016-11'	'1132'	'245'
23	'2016-12'	'1928'	'139'

openedx/edx-platform

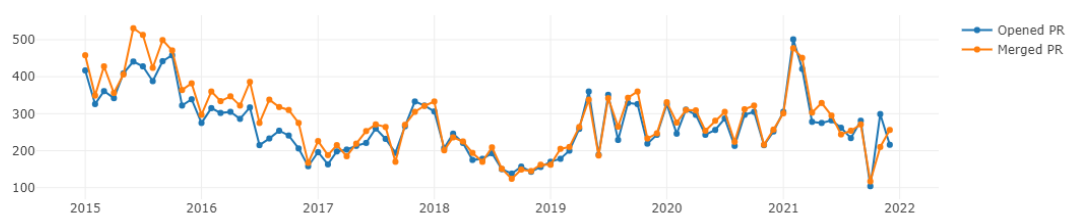
(index)	Time	Opened Issues	Closed Issues
0	'2015-1'	'417'	'423'
1	'2015-2'	'326'	'325'
2	'2015-3'	'361'	'383'
3	'2015-4'	'342'	'326'
4	'2015-5'	'409'	'382'
5	'2015-6'	'441'	'497'
6	'2015-7'	'428'	'466'
7	'2015-8'	'388'	'389'
8	'2015-9'	'442'	'461'
9	'2015-10'	'458'	'448'
10	'2015-11'	'322'	'324'
11	'2015-12'	'339'	'359'
12	'2016-1'	'275'	'271'
13	'2016-2'	'315'	'319'
14	'2016-3'	'302'	'310'
15	'2016-4'	'305'	'308'
16	'2016-5'	'286'	'303'
17	'2016-6'	'317'	'310'
18	'2016-7'	'215'	'225'
19	'2016-8'	'233'	'294'
20	'2016-9'	'254'	'250'
21	'2016-10'	'241'	'262'
22	'2016-11'	'206'	'232'
23	'2016-12'	'158'	'144'

• 每月打开 PR 和合入PR 的个数

angular/material: Opened/Merged PRs by Month (2015 - 2022)

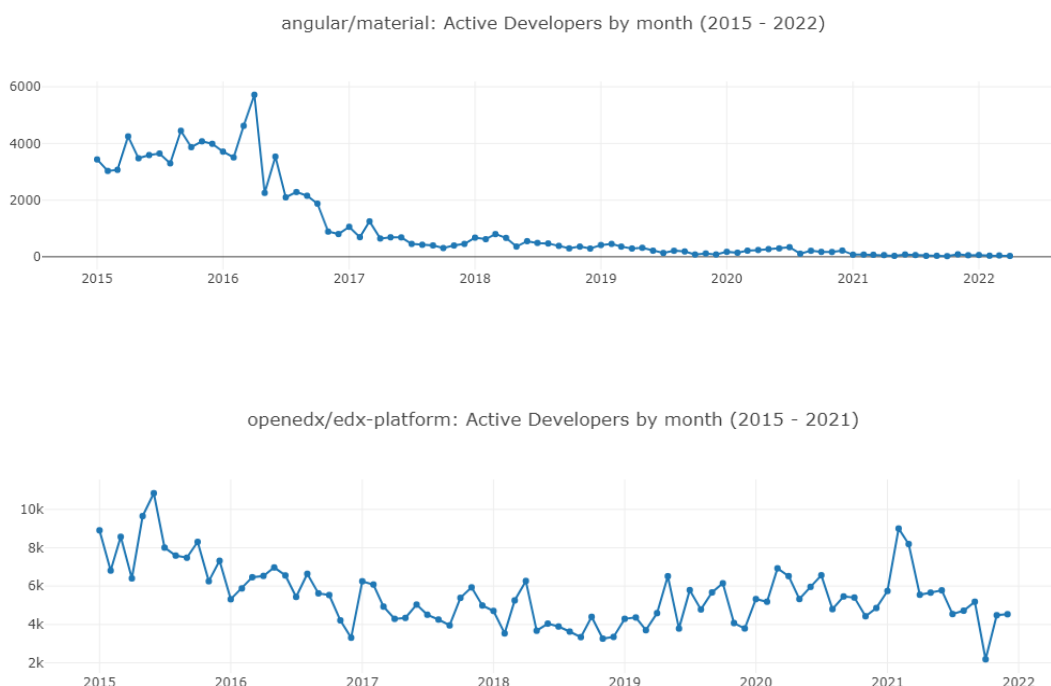


openedx/edx-platform: Opened/Merged PRs by Month (2015 - 2021)



- 上图为归档项目 `angular/material`，下图为活跃项目 `openedx/edx-platform`。
- 与上一指标的变化趋势类似，从“每月打开 / 合入PR 的个数”变化趋势图可以看到，`angular/material` 在项目初始阶段较为活跃，后期逐渐趋于个位数，并且在项目运行阶段 **打开的PRs数量要高于合入的PRs数量**，这再一次说明该项目的PRs没有得到开发者社区及时的反馈处理；而 `openedx/edx-platform` 在2015-2016年以及2020-2021年期间发生两次剧烈上升的趋势，中间阶段始终维持在一个比较稳定的状态，**打开的PRs数与合入的PRs呈持平状态**。
- 不同于上一指标的数量差异，从“每月打开 / 合入PR 的个数”上看，`angular/material` 对应的数量要小于 `openedx/edx-platform`，两个指标的差异再次说明，`openedx/edx-platform` **开发者社区的活跃程度以及反馈速度要远胜于 `angular/material`**。

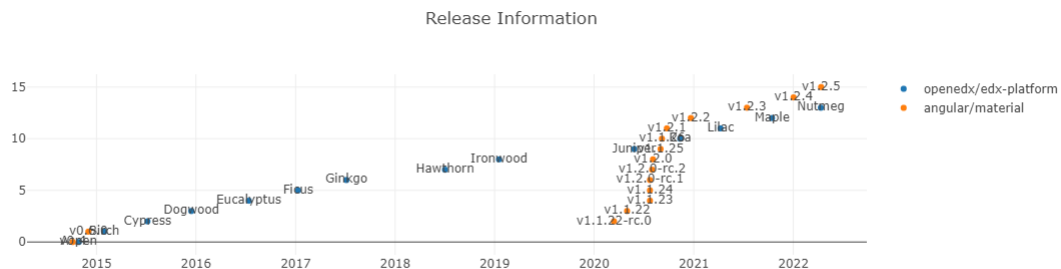
3.2 开发者数量变化趋势的对比



- 上图为归档项目 `angular/material`，下图为活跃项目 `openedx/edx-platform`。
- 从“每月活跃的开发者的数量”变化趋势图可以看到，`angular/material` 项目在2015-2017年期间活跃的开发者的数量较多，且**先上升后下降的趋势**，2017年之后数量逐渐下降；而 `openedx/edx-platform` 项目的活跃开发者的数量从2015年到至今**始终保持在千位级别**，处于稳定发展的状态。
- 两个项目在这一指标上的差异明确表现出，开发者的活跃程度对项目的生命力起着至关重要的作用。

3.3 项目版本发布频率分析

- `angular/material` 共发布版本16次，`openedx/edx-platform` 共发布版本14次，其发布具体时间如下图所示：



- 从上图可以发现，`openedx/edx-platform` 项目从2015-2022年期间始终以一个较稳定的频率（一年之内）发布项目的更新版本；而 `angular/material` 项目在发布前两个版本之后，中间有五年（2016-2020年）的空闲期并未发布任何更新版本，这段时间也是该项目被 Star/Fork、打开/关闭Issues、打开/合入PRs数、活跃开发者数量最少的一年。
- 从项目版本发布频率的角度可以看出，**维护者社区能否积极迭代开源项目**对于项目本身的活跃度和生命力起着至关重要的作用。

3.4 对于项目发展到活跃/归档影响因素的总结与思考

- 开源项目的维护是一件长期且艰巨的工作，对于开源维护者而言，**社区的支持和团队的合作**是开源项目的主要动力来源，然而获得社区支持和积极的贡献者是一个巨大的挑战。我们都希望技术人员能够找到自己的开源项目，并开始做出贡献，但现实是能够积极且坚持投入到其中的技术人员仅在少数；并且，如何在众多开源项目中吸引技术人员参与你所在的开源项目中，也是十分重要的。试想一下，开发人员每天创建好几个公共存储库，但是没有人知道这些存储库的存在；如果开源项目没有社区协作的参与，其实并未真正获得开源的优势。
- 因此，如何保持开源项目的活跃度，在以下方面可以获得启发：
 - **主动推广开源项目。**如何在数以万计的开源项目中脱颖而出，获得更多开发者的关注，主动的“推销宣传”是必不可少的。维护者社区通过多种渠道来宣传自己的项目，如举办会议讲座、线下Meeting活动、线上Blog平台发文宣传等，增加项目的曝光率。
- **建立稳定的、积极的贡献者社区**，获得社区支持。社区是开源的很大一部分组成成分，也是一个成功开源项目的基石。社区支持是双向的，为了维持社区的支持，及时的提供社区成员适当的奖励是一个很好的方法，如，在README中公开贡献者/维护者列表，以表示对其开发贡献的认可，同时也可以激励更多成员参与。
- **营造健康的、有秩序的社区环境。**在开源项目中**规范行为准则**，有助于社区成员高效及时的交流，同时也能够更快的让新手认识并参与到其中；若有人不遵守规则，也应该具有相应的处罚办法。此外，一个活跃的开源项目往往**具有一个开放讨论的空间**，通常来说，贡献者加入一个开源社区是为了和志同道合的技术人员交朋友，或者是他们遇到一个技术问题却找不到人进行咨询。开源项目的维护者经常使用一些工具来搭建一个开放讨论的平台，让贡献者能够沟通、交流。例如：GitHub Discussions、Discord等。
- **保持开源项目的“持续性”。**开源项目的活跃度往往是具有正反馈性的：人们往往会更愿意参与一个活跃的项目，随着贡献者的不断增多，项目会得到更多的代码支持以及更多的关注度。因此，作为开源项目的维护者，即使没有其他的贡献者，也要保持你自己的贡献是积极的。

参考材料

1. openedx官网: <https://openedx.org/>
2. openedx文档: <https://docs.edx.org/>
3. angular/material链接: <https://github.com/angular/material>
4. <https://baijiahao.baidu.com/s?id=1736776368554284633&wfr=spider&for=pc>