

开源课程期中作业

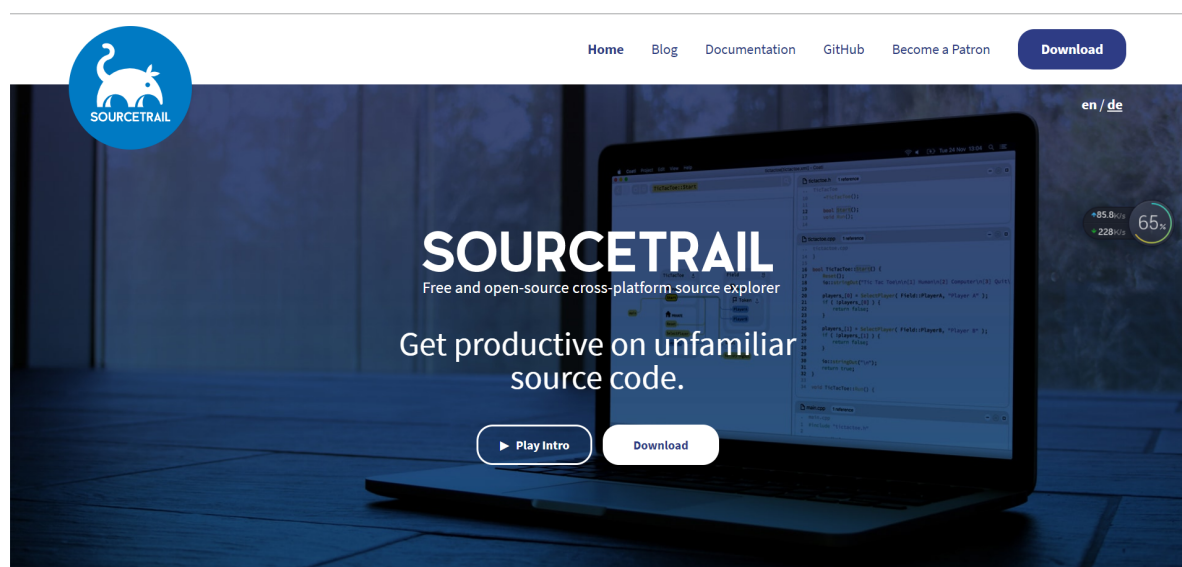
小组成员：

阎松 51215903029, 李雪51215903030, 卢哲钰51215903011

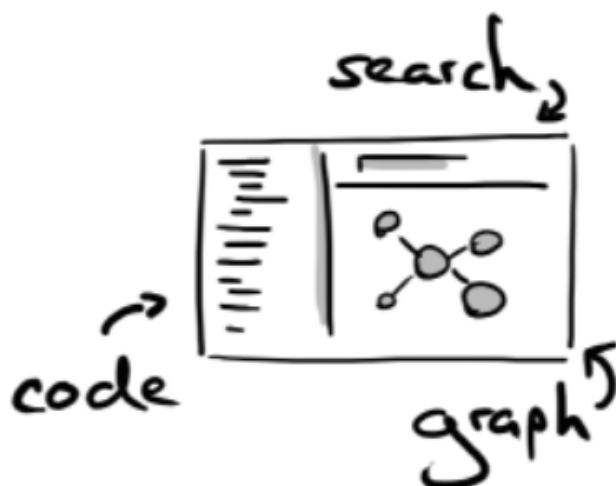
一、项目的基本背景和发展历程介绍

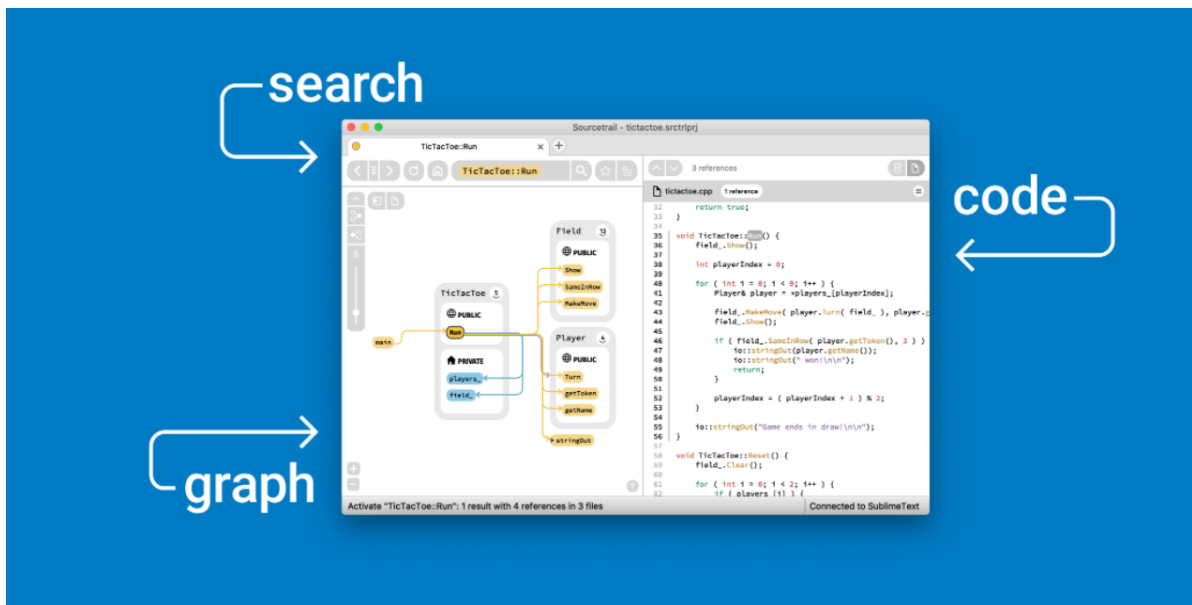
通过项目仓库的查阅，以及搜索引擎上项目的相关资料，给出项目一些基本的介绍：

技术类型：

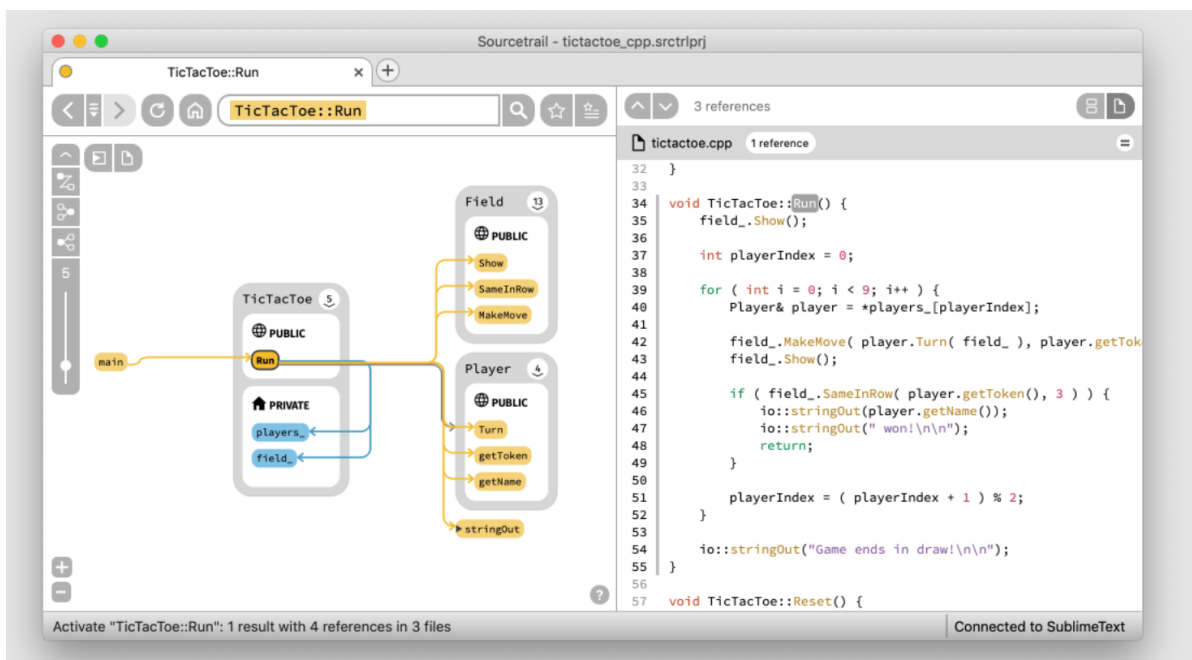


Sourcetrail 是一个免费的开源跨平台源代码浏览器，可帮助您在熟悉的源代码上高效工作。Sourcetrail 是一个交互式源代码浏览器，它通过索引代码和收集有关其结构的数据来简化现有源代码中的导航。然后 Sourcetrail 提供了一个简单的界面，该界面由三个交互式视图组成，每个视图在帮助您获取所需信息方面发挥着关键作用：





- 搜索：使用搜索字段快速查找和选择源代码中的索引符号。自动完成框将立即提供整个代码库中所有匹配结果的概览。
- 图表：图表显示源代码的结构。它专注于当前选定的符号，并直接显示所有传入和传出对其他符号的依赖关系。
- 代码：代码视图在代码片段列表中显示当前选定符号的所有源位置。单击不同的源位置可让您更改选择并深入挖掘。



Sourcetrail 的用户界面分为三个主要视图。它们的排列可以根据需要进行调整，也可以从主窗口中分离出来并分成不同的屏幕。

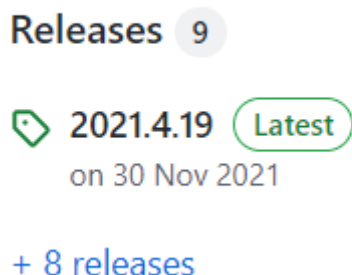
软件开发人员大部分时间都在找出现有的源代码，但常见的代码编辑工具对这项任务几乎没有帮助。调试器只允许详细检查一个狭窄的代码路径。“查找所有引用”可帮助您在文件之间导航，但不提供上下文以查看所有相关依赖项的全局。Sourcetrail 通过结合交互式依赖图、简洁的代码视图和高效的代码搜索来提供概述和详细信息，所有这些都内置在易于使用的跨平台开发工具中。它支持您探索遗留代码、理解实现和重构软件架构。Sourcetrail 是：

- 自由
- 离线工作

- 在 Windows、macOS 和 Linux 上运行
- 支持 C、C++、Java 和 Python
- 提供 SDK ([SourceTrailDB](#)) 来编写自定义语言扩展

版本发布历史:

egraether在其github网站上面共发布9次releases。



在其官网上面，2017.2月宣布他们的 Beta 版结束，并发布Sourcetrail 1.0。

Sourcetrail 2017.3

显示变量类型和函数签名的交互式工具提示；用于索引的命令行 API；性能改进；更好的 Java 覆盖率；自动更新检查。

Sourcetrail 2017.4

现在免费用于非商业用途！我们的新版本还具有屏幕搜索、改进的 C++ 模板索引、完整的 Java 支持和 Gradle 的项目设置。

Sourcetrail 2018.1

优化的 UI 风格，许多新的可用性功能和已知问题的修复。

Sourcetrail 2018.2

更清晰的图形布局，按文件路径和命名空间分组，改进了错误修复的可用性，代码视图中的性能改进。

Sourcetrail 2018.3

在索引时继续浏览，Sonargraph 和 Code::Blocks 项目的项目设置以及图形和代码视图的新可用性功能。

Sourcetrail 2018.4

大幅提高索引性能、减少内存消耗、同时打开多个符号的新标签栏以及模板/泛型的更多类型使用边缘。

Sourcetrail 2019.1

实验性 Python 支持，使用 SourceTrailDB 和 UI 更新编写自定义索引器，包括更简单的代码片段 UI。

Sourcetrail 2019.3

新的自定义跟踪对话框，用于使用节点和边缘过滤器在符号 A 和 B 之间创建图形。预编译头文件 (PCH) 支持 C 和 C++ 源代码组。

Sourcetrail 2020.1

为图形和代码视图引入了基于键盘的导航、项目设置中更简单的源组设置以及适用于 Linux 的 AppImage 下载。

Sourcetrail 2020.2

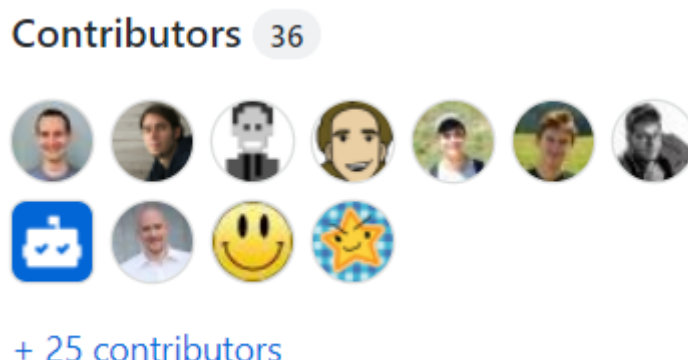
更新到 LLVM/Clang 10、Python 索引器更新、修订的 Linux Tarball 包以及一些修复和可用性改进。

2021 年 9 月 23 日

停止 Sourcetrail

决定停止维护 Sourcetrail。

主要贡献者的构成（国家、区域和组织等）：



Eberhard Gräther有1,395 commits，来自德国。是Coati Software的创始人和开发人员，致力于 Sourcetrail，这是一个 C、C++ 和 Java 的可视化源代码浏览器。

Malte Langkabel 有973次commits，来自德国的Salzburg（萨尔茨堡）一个专门的游戏程序员和 AI 爱好者。

Andreas Stallinger有117次commits。

wrongway88有21次commits。

其他的贡献者或多或少的进行了1-10次以内的commits。

CI/CD 的使用：

CI/CD 是一种持续开发软件的方法，可以不断的进行构建、测试和部署代码迭代更改。这种迭代有助于减少基于错误或失败的版本进行开发新代码的可能性。使用这种方法，从新代码开发到部署，可以减少人工干预甚至不用干预。

达到持续的方法主要是：持续集成，持续交付，持续部署。

CI（Continuous Integration）：持续集成，也就是当每一次更改的代码被推送到远程分支后，可以创建一组脚本来自动地构建和测试这些更改，确保这些更改可以通过一些基本的准则，减少引入错误的机会。

CD：

Continuous Delivery：持续交付，在持续集成的基础上更进一步，当每一次更改的代码落库后，不仅会构建和测试，也会进行部署，但是部署需要人工干预，手动的有目的进行部署。

Continuous Deployment：持续部署，持续集成之外的另一个步骤，类似于持续交付。不同之处在于，它不是手动部署应用程序，而是将其设置为自动部署。不需要人为干预。

在项目迭代过程，可能有一个专门负责 CI/CD 的人员，但当想做一些静态代码检查，依赖检查，图片大小检查等事情的时候，就自己需要了解 CI/CD，编写特定 Pipeline Job。

在实际项目中，CI 主要是提交或合并代码的时候触发，负责一些基本规则的检查，如果检查遇到失败，那么回滚或修改代码后再提交或合并，降低代码风险；CD 主要手动的触发，在CI的基础上，还负责功能检查，如果功能符合验收标准，那么就可以交付或部署。

通过一个工具，我们可以看到其使用情况。

在Github上增加了一个第三方的插件—Appveyor，来简单的实现了CI/CD 操作,通过注册账号, 然后各种配置以后,可以实现每次向Github提交,会自动编译，然后生成报告。

AppVeyor			Sign in
docs: Change link to blog post in Readme		0.0.0.946	
5 months ago by Malte Langkabel (committed by GitHub)	🔗 master ↗ 4b1b0e4f	5 months ago in 14 min 39 sec	
Pull request #1232 - Fix compilation issue			
Merge pull request #1 from str0g/archlinux-compilation-fix		0.0.0.945	
5 months ago by Lukasz Buško (committed by GitHub)	🔗 master ↗ b09636cf ← 🔗 master ↗ 56ab496c	5 months ago in 14 min 39 sec	
res: Changelog for 2021.4.19		0.0.0.944	
5 months ago by Eberhard Graether	🔗 master ↗ bdd3c4d9	5 months ago in 16 min 9 sec	
update changelog		0.0.0.943	
5 months ago by Malte Langkabel (committed by GitHub)	🔗 changelog ↗ a9f47e53	5 months ago in 16 min 39 sec	
changelog		0.0.0.942	
5 months ago by Eberhard Graether	🔗 changelog ↗ 721de618	5 months ago in 15 min 12 sec	
docs: Changed image sizes to not cover the full width		0.0.0.941	
5 months ago by Eberhard Graether	🔗 2021.4.19 ↗ 2863686b	5 months ago in 15 min 14 sec	
docs: Changed image sizes to not cover the full width		0.0.0.940	
5 months ago by Eberhard Graether	🔗 master ↗ 2863686b	5 months ago in 16 min 16 sec	
src: switch in-app links to documentation to .md file on GitHub(#1230)		0.0.0.939	
6 months ago by Malte Langkabel (committed by GitHub)	🔗 master ↗ e992f14f	6 months ago in 18 min 3 sec	
doc: remove website links from readme		0.0.0.938	
6 months ago by Malte Langkabel (committed by GitHub)	🔗 master ↗ 3833a93f	6 months ago in 15 min 59 sec	
Pull request #1230 - switch in-app links to documentation to .md format			
switch in-app links to documentation to .md format		0.0.0.937	
6 months ago by mlangkabel	🔗 master ↗ 8844aaff ← 🔗 documentation_link ↗ b2a016b5	6 months ago in 18 min 53 sec	

下图展示的是其最后一次使用情况：

Sourcetrail

Current build History

LOG

docs: Change link to blog post in Readme 0.0.0.946

5 months ago by Malte Langkabel (committed by GitHub) 🔗 master ↗ 4b1b0e4f 5 months ago in 14 min 39 sec

Console

Messages

Tests

Artifacts

```
1 Build started
2 git clone -q --branch=master https://github.com/CoatiSoftware/Sourcetrail.git C:\projects\sourcetrail
3 git checkout -qf 4b1b0e4fd19c4af235fef12b0564c05348f5fd3
4 Running Install scripts
5 set NINJA_URL="https://github.com/ninja-build/ninja/releases/download/v1.9.0/ninja-win.zip"
6 appveyor DownloadFile %NINJA_URL% -FileName ninja.zip
7 Downloading ninja.zip (254,497 bytes)...100%
8 7z x ninja.zip -oC:\projects\deps\ninja > nul
9 set PATH=C:\projects\deps\ninja;%PATH%
10 ninja --version
11 1.9.0
12 mkdir build
13 cd build
14 mkdir temp
15 cd ../
16 cd build
17 mkdir win64
18 cd win64
19 call "C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\VC\Auxiliary\Build\vcvarsall.bat" amd64
20 *****
21 ** Visual Studio 2017 Developer Command Prompt v15.9.19
22 ** Copyright (c) 2017 Microsoft Corporation
23 *****
24 [vcvarsall.bat] Environment initialized for: 'x64'
```

其他有价值的信息：

为什么这个产品被开发并且一直维护了三年。

作为软件开发人员，我们拥有实现我们所有想法的技能 and 能力。但它往往比我们最初预期的要复杂。因此，专业软件开发主要是团队合作。作为一个团队，我们可以解决更困难的问题。然而，这也有代价：随着代码库的不断增长，跟踪所有更改变得更加困难。因此，越来越多的时间花在理解现有代码上，而不是添加新功能上。刚接触软件项目的开发人员通常要花费数天甚至数周的时间来浏览代码库，然后才能最终获得生产力。

Sourcetrail 是一个轻量级的开发工具，专注于通过软件可视化进行快速和有洞察力的导航。以前的软件可视化通常只适用于小型项目。静态图很快就会变得复杂、难以阅读或几乎无法在屏幕上显示。另一方面，Sourcetrail 使用计算机系统的交互性并动态生成仅显示当前相关子区域的图表，包括所有依赖项。因为仅仅可视化是不够的，Sourcetrail 总是向您显示除图表之外的相关代码行。这个代码和图表包通过强大的搜索功能来完善，可以快速让您进入导航的入口点。通过这种组合，Sourcetrail 缩短了您熟悉不熟悉代码的时间。这为您真正想做的事情留出了更多时间。

二、项目的历史轨迹分析

见Sourcetrail.ipynb。

三、洞察项目被归档的可能原因

该项目由 Sourcetrail 的原始作者和维护者于 2021 年底存档。

在 2019 年秋季将 Sourcetrail 转换为开源项目后，他们的这个项目最初得到了很多关注，并且非常有动力继续开展该项目。在他们的前几个季度发布时保持了良好的时间表，但很快进展变得缓慢，最终不再对电子邮件、新问题和拉取请求（PR）做出响应。

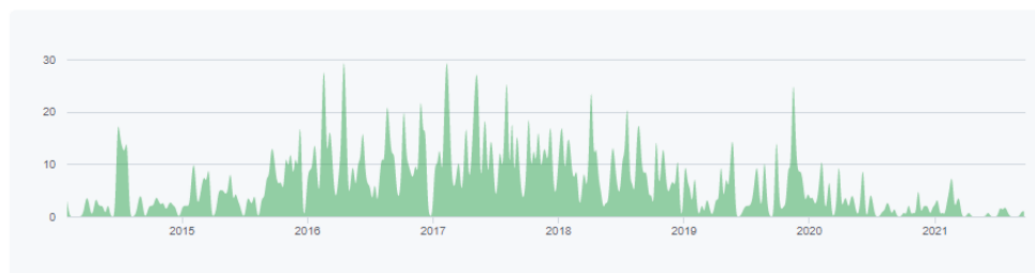
有几个原因：

- 1.关闭他们的初创公司后，他们将每天的工作时间投入到新工作中。
- 2.然后在 2020 年春季，公司的老板从萨尔茨堡搬到了维也纳，这使得合作变得更加困难，因为他们的成员不再同处一地。
- 3.慢慢地，重新开始该项目的工作也成为一项挑战。毕竟，它一直是一个雄心勃勃的软件，具有许多技术依赖项，需要持续关注以保持最新版本的最新状态。
- 4.特别是考虑到 Sourcetrail 在多个平台上运行，支持多种语言和构建系统，并与不同的 3rd 方工具集成。总而言之，这使得维护和恢复变得相当困难。

Feb 2, 2014 – Sep 23, 2021

Contributions: Commits ▼

Contributions to master, excluding merge commits and bot accounts



Sourcetrail 的兴衰随着时间的推移而变化。

最终，他们的创始者也失去了在语言分析和软件可视化领域工作的兴趣，并开始将重点放在新的领域。虽然他们所取得的成就感到自豪，但在了解了创建开发工具的现实挑战，并且知道它需要大量的工作，这些工作往往令人沮丧，而且根本不好玩。

由于所有这些原因，他们决定停止该项目。

但他们在2021年秋季发布 Sourcetrail 2021.4 的最终版本。之后，将网站离线并将存储库存档在 GitHub 上，这实际上冻结了git 存储库、问题列表和 wiki。但是，它仍然允许任何人查看存储库、分叉并下载任何托管版本。

究其原因，我认为是开源项目最需要的是大家的共同合作，一个人很难完成一个巨大的工作，因此，合作共赢是一个团队应该做的事情。