

# VSCode 开源时序数据分析与可视化报告

李佳懿、任雪宁

(选取赛道为：开源社区数据的时序分析与可视化)

(项目讲解完整视频：<https://b23.tv/5ddZrwT>)

## 研究目的：

### 一、洞察项目发展趋势

通过对 VSCode 在 GitHub 上开源数据的时序分析，精确描绘其从起始至今的发展轨迹。涵盖诸如每月或每年的 issue 创建与关闭数量、参与者数量等关键指标，深入理解项目在不同阶段的增长速度、活跃程度以及发展态势，同时精准把握这些趋势，为项目未来规划提供有力的数据支撑，助力合理分配资源，确保项目朝着预期方向稳健发展。

### 二、评估项目健康状况

借助对开源数据的多维度分析，全面评估 VSCode 项目的健康程度。例如从社区活跃度、开发者参与度以及新贡献者的加入频率等，判断社区是否充满活力与创造力；从巴士系数角度，分析项目对核心开发者的依赖程度以及团队知识的分散情况。通过对这些方面的综合评估，及时发现项目发展过程中潜在的问题，如社区活跃度下降、对核心开发者过度依赖等，以便采取针对性措施加以解决，维持项目的健康发展。

### 三、挖掘影响项目的关键因素

深入挖掘影响 VSCode 项目发展的各种因素，包括内部因素与外部因素。内部因素如项目自身的技术架构、功能特性、开发流程等；外部因素如行业技术发展趋势、竞争对手动态、市场需求变化等。明确这些关键因素及其作用机制，有助于项目团队制定更为有效的发展策略，在竞争激烈的开源领域保持领先地位。

### 四、促进开源社区协作与交流

通过数据大屏和数据洞察报告的可视化展示，为 VSCode 开源社区的参与者提供直观、全面的数据洞察。数据大屏以直观的图表、图形展示项目的关键指标和发展趋势，方便社区成员快速了解项目动态；数据洞察报告则深入剖析数据背后的原因和影响，为社区成员提供深入的见解。这不仅有助于增强社区成员对项目的了解和认同感，还能促进成员之间的协作与交流。例如，项目管理者可以依据数据调整管理策略，更好地协调社区资源。同时，可视化成果也便于向外部展示项目的优势和潜力，吸引更多潜在开发者和用户加入，进一步壮大开源社区。

## 数据来源：

通过开源工具集 OpenDigger 获取 2015 年 11 月-2023 年 3 月 VSCode 在 Github 上的官方仓库

microsoft /vscode 的相关数据。

## 研究内容：

开发完成一个功能完善的可视化大屏，能够按需展示 VSCode 项目的时序数据，支持多维度数据探索和交互操作，并给出一份详细的 VSCode 项目时序数据分析报告。

## 研究方法：

### 一、数据预处理

**选取数据集：**根据分析目的，从 OpenDigger 上选取需要的 VSCode 的数据集。

**数据清洗与整理：**识别并处理数据中的异常值，确保数据的准确性和一致性。

**数据格式化：**将数据按照时间粒度分段处理，以月或年分段，以便进行时序分析。

### 二、数据分析

**选取数据集中最具分析价值的数据：**提取 VSCode 项目的关键时序指标，如项目每月的活跃度、受关注度和巴士系数等关键指标。

**对数据进行趋势分析：**分析 VSCode 项目的各类数据随时间变化的趋势，关键指标每月或每年的增减情况。

**周期性分析：**探索数据中的周期性模式，如月度、季度或年度周期，识别项目中的季节性波动和周期性变化。

**对比分析：**选取相关的两组不同的数据进行对比分析，揭示不同数据之间的关联性和相互影响。

**计算基本统计量：**计算关键时序指标的平均值和标准差等基本统计量，评估数据的稳定性和离散程度。

### 三、数据大屏实现

**设计数据图表：**选择折线图、柱状图、饼图、雷达图以及滚动屏来展示数据分析结果，并使用 echarts.min.js 库来绘制各类图表。

**设计大屏的布局和样式：**采用了 Flexbox 布局技术来构建灵活的框架。具体而言，.mainbox 容器通过设置 display: flex; 属性，实现了其内部不同图表和数据区域的灵活排列与对齐。这一布局策略确保了各个区域能够整齐且有序地展示在大屏之上；选择一致的配色方案，以科技风的蓝绿色系作为大屏的主色调，确保图表和数据区域的颜色和谐；设计适当的图标，如切换年份的按钮增加了白色阴影，看上去更具真实感，数据量大的折线图增加半透明的滑块图标，饼状图和雷达图设置不同颜色的图例，提高了信息的可读性等。

**实现大屏的交互功能：**利用 HTML、CSS 和 JavaScript 以及 Echart 库和 jQuery 库实现大屏的交互功能。

如使用 JavaScript 监听按钮切换图表数据、展示提示信息、缩放图表与字体、滑动查看数据窗口；用 CSS 实现交互时鼠标的悬停效果以及滚动条模拟效果；用 HTML 实现实时时间显示，利用 jquery.limarquee.js 插件实现数据的列表项向上滚动显示等。同时进行了点击按钮时改变按钮颜色、可以自由拖动滚动屏等细节设计。

#### 四、测试与评估

**对可视化大屏进行全面的测试：**确保了所有图表和数据区域正确展示，数据准确无误；所有交互功能，如按钮点击、数据过滤等功能正常；测试了大屏在不同浏览器（360 浏览器、Google、Microsoft Edge、华为浏览器）上的显示效果和功能表现。

**根据测试结果对大屏进行优化和改进：**针对测试中发现的功能错误和性能问题，进行代码修复和优化，确保大屏的稳定性和准确性。

### 项目创新点：

#### 一、数据可视化大屏

**直观性：**借助多种图表，将原本复杂、抽象的数据直观且富有吸引力地展示出来。使之能迅速捕捉数据中的关键信息，洞察数据背后隐藏的趋势、规律。

**交互性：**融入了先进的交互技术，用户只需移动鼠标就可以展现细节数据等操作，进一步挖掘数据的深层价值。同时，我们添加滑块、按钮等各种交互元素，以使用户进行数据查看。

**高效性：**在有限的物理空间内同时容纳并展示海量的数据信息。只需简单操作即可轻松切换视角，全面审视数据在不同维度、不同时间段的表现，为精准决策提供了强有力的支持。

**多维度：**为了全面展现数据的多样性，从时间维度深入剖析各类数据的独特特征和演变趋势。旨在协助用户从多个视角洞悉当前状况，从而获取更全面的认知。通过时间序列分析，用户不仅能够观察到数据随时间的波动情况，还能揭示其背后的周期性以及潜在的增长或衰退模式。

#### 二、数据联动

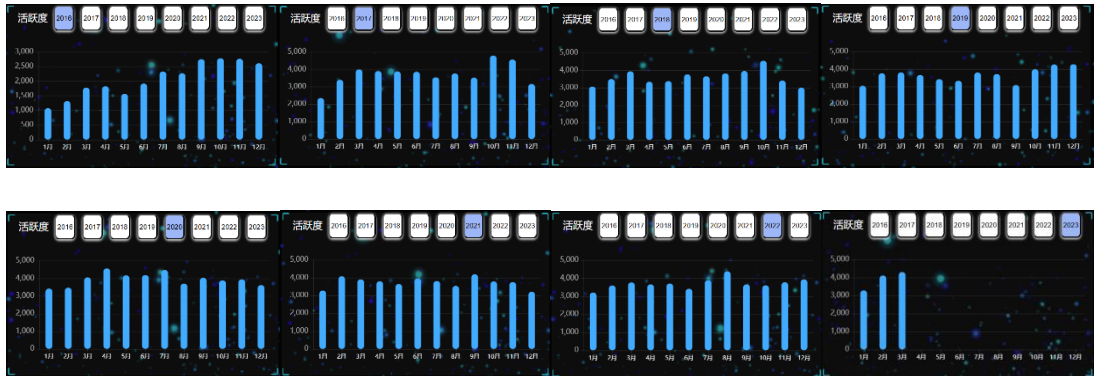
我们将在同一时序下的代表不同信息的数据进行一个总体联动，即通过各方面数据展现出的特点，进行综合分析，以更加全面的角度展现现状。

### 最终成果：



数据洞察:

活跃度柱状图:



整体趋势:

**波动上升趋势:** 从2016年1月到2023年3月, 活跃度整体呈现波动上升态势, 后期呈现平稳。虽然期间有不少起伏, 但早期(2016年)的活跃度数值普遍低于后期(2022-2023年), 例如2016年1月活跃度为1072.51, 而2023年3月已达到4303.48, 说明VSCode在长期发展过程中, 社区参与度在逐渐提升。

**周期性波动:** 每年内存在一定周期性波动, 年初和年末活跃度变化较为明显, 这种波动或许与节假日、开发计划周期等因素相关。

季节特征:

**部分年份二季度表现:** 通过观察数据, 一些年份的四季度(10-12月)活跃度表现亮眼, 同时一些年份的10月份达到当年活跃度顶峰, 例如: 2016年、2017年、2019年, 这可能是因为年末技术迭代更新等促使开发与交流频繁。

年份差异：

**快速增长年：**2017 年 VSCode 活跃度增长显著，对比 2016 年各月，几乎所有月份活跃度都有较大幅度提升，这可能得益于 VSCode 功能拓展、推广或者技术社区热点带动，让 VSCode 吸引到更多开发者参与互动。

**平缓过渡年：**部分年份活跃度增长平缓，例如 2021-2022 年，各月活跃度数值起伏相对较小，VSCode 或许进入稳定维护、小步迭代阶段，没有重大更新引发热度的急剧变化。

异常点：

**个别月份跃升：**2017 年 10 月活跃度 4787.01 显著高于相邻月份，很可能是 VSCode 在该月发布了重要版本、举办大型线上线下活动，或是被知名技术媒体报道，引发社区短期内高度关注。

**突然下滑：**个别月份有活跃度突然下滑的情况，如 2019 年 9 月活跃度 3085.86，相较 8 月 3714.19 落差明显，需观察同期是否有市场竞争、VSCode 自身技术故障或者社区运营问题。

未来预期：

根据柱状图分析，未来可能会继续保持相对稳定状态。

受欢迎程度折线图：



**变化趋势：**从折线图可以直观看到关注度和收藏量整体上都呈现出先上升后下降的趋势。在 2016-2019 年期间，关注度和收藏量都在增加，表明这期间 VSCode 项目的受关注程度和被喜爱程度不断提升。2019-2022 年出现下降，可能是由于项目发展进入新阶段等多种因素导致。

**平均值：**关注度平均值约为 31486.57，收藏量平均值约为 21713.29。这表明在 2016-2022 年期间，平均每年能吸引到三万多用户的关注，整体处于较高的受关注水平，说明该项目在开发工具领域具有一定的重要性和影响力，而收藏量相对关注度低一些，反映出虽然项目受到较多关注，但实际进行收藏操作的用户占比相对关注度而言略低，可能存在部分用户只是临时关注项目，而未将其加入收藏夹以便长期关注。

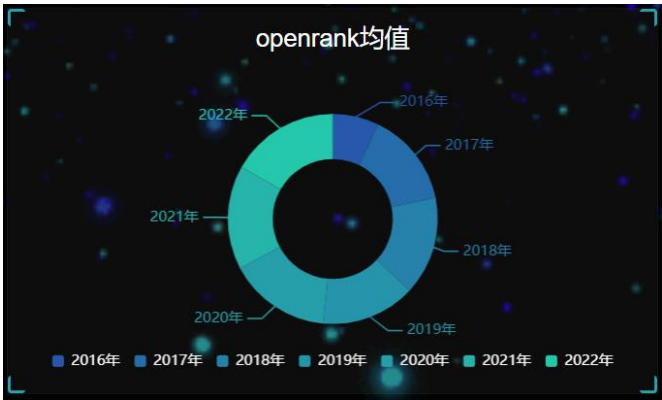
**标准差：**标准差反映了数据的离散程度。关注度标准差约为 6403.03，相对较大，说明关注度数据在均值周围的波动较为明显，即不同年份之间的关注度差异较大，项目受关注程度不够稳定，易受到多种因素的影响而出现较大幅度的波动。收藏量标准差约为 4324.16，小于关注度标准差，表明收藏量数据相对较为

集中，波动较小，说明项目的收藏量变化相对平稳，每年新增的收藏用户数量相对稳定。

**相关性：**关注度与收藏量的相关系数约为 0.94，呈现出高度正相关关系。这意味着当关注度升高时，收藏量也倾向于升高；反之，关注度降低时，收藏量也倾向于降低。说明用户对项目的关注行为和收藏行为之间存在较强的关联，关注的用户更有可能进行收藏。

**未来趋势：**可能会继续维持在当前水平或者有小幅度的波动。

**openrank 均值饼状图：**



项目的协作影响力，即一个影响力较大的项目，会和更多的项目有协作关系；对于影响力较大的项目，与其协作关联度较高的项目影响力也会较大。

**趋势分析：**

**整体上升趋势：**从 2016 年的 698.58 开始，到 2022 年增长至 1634.33，这表明在这段时间内，VSCode 项目在 GitHub 平台上的 openrank 指标呈现出持续上升的态势。这可能意味着该项目在代码开源生态中的相对地位在不断提升，更多的开发者参与到项目中来，包括提交代码、创建分支、发起讨论等活动，使得项目的整体热度和影响力逐步扩大。

**阶段性分析：**

**2016-2017 年快速增长阶段：**在这一阶段，openrank 数值大幅提升，可能是因为 VSCode 推出了协作功能，吸引了大量开发者加入协作，使得项目的 openrank 快速上升。

**2017-2019 年波动调整阶段：**2017-2018 年增长幅度变小，2018-2019 年出现下降。这可能是由于市场竞争加剧，其他类似的代码编辑器推出了有竞争力的功能，分散了 VSCode 的部分用户和开发者关注，导致其 openrank 增长乏力甚至出现下滑。

**2019-2022 年稳步回升增长阶段：**从 2019 年到 2022 年，openrank 持续上升，表明 VSCode 项目重新调整策略，通过不断优化性能、增加新特性以及加强社区建设等措施，重新吸引了开发者的关注和参与，使得项目在 GitHub 平台上的影响力持续提升。

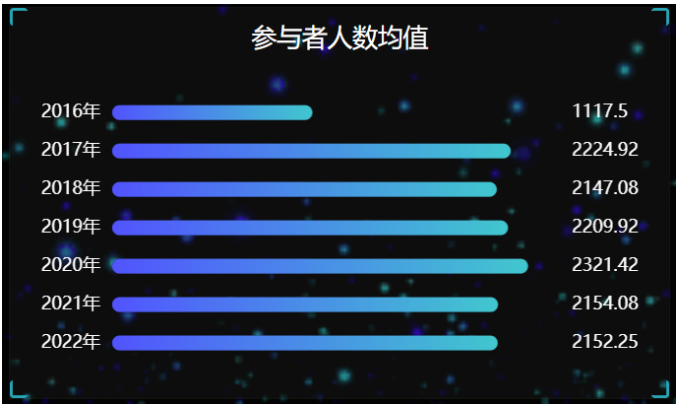
**与行业发展的关联推测：**在技术快速发展的编程领域，VSCode 项目 openrank 值总体上升，暗示其能较好适应行业变化。随着新编程语言和框架不断涌现，VSCode 可能及时提供支持，保持对开发者的吸引



力。另一方面，VSCode 由于其创新的代码编辑功能和开发流程优化方式，被其他项目借鉴学习，进一步提升了项目影响力，从而推动 openrank 值上升。

**未来趋势：**在没有外部干预的情况下保持相对稳定，未来可能会呈持续上升趋势。

**参与者人数均值条形图：**



**整体趋势：**

从2016年的1117.5人到2022年的2152.25人，整体呈现出上升趋势。尽管过程中存在波动，但总体表明在这期间，平均参与VSCode项目的人数有所增加，反映出项目对开发者的吸引力在逐步增强，越来越多的人愿意参与到项目的建设中来。这可能得益于项目本身的发展壮大、功能的不断完善、应用场景的拓展等。

**年度变化及分析：**

**2016-2017 年：**人数均值从1117.5增长到2224.92，增长幅度显著，近乎翻倍。这可能是因为2016-2017年间，VSCode项目取得了重大进展，比如发布了具有里程碑意义的版本，新增了许多开发者急需的功能，对热门编程语言的更好支持、代码智能提示功能的大幅优化等。这些改进吸引了大量新开发者加入项目，使得平均参与人数大幅提升。

**2017-2018 年：**均值从2224.92下降到2147.08，出现了一定程度的下滑。这或许是由于在2017年快速增长后，项目进入了一个相对调整期。在这一年，项目的发展重点可能从功能拓展转向了内部优化，新功能推出速度放缓，对部分开发者的吸引力暂时降低，导致参与人数略有减少。

**2018-2019 年：**均值从2147.08上升至2209.92，有所回升。这可能是因为VSCode团队针对上一年的调整做出了积极回应，在这一年改进了开发工具的易用性、优化了代码调试流程等，重新吸引了开发者的兴趣，使得参与人数再次增加。

**2019-2020 年：**人数均值从2209.92增长到2321.42，保持上升态势。可能是因为VSCode在这一年对新兴技术和框架提供了更好的支持，例如对新出现的热门前端框架的集成，吸引了更多专注于这些领域的开发者参与项目，进一步扩大了参与者群体。

**2020-2021 年：**均值从2321.42下降到2154.08，再次出现下降。这可能与当时的行业竞争环境有关，其他类似的开源项目在这一年可能推出了更具吸引力的功能或举措，吸引了部分原本参与VSCode项目的开

发者，导致参与人数减少。

**2021-2022 年：**均值从 2154.08 微降至 2152.25，基本保持稳定。这表明项目在这一年处于相对平稳的发展阶段，虽然人数略有下降，但幅度极小，说明 VSCode 在开发者群体中已经拥有了较为稳定的用户基础，即使面临一些小的挑战，也能维持相对稳定的参与人数。

**影响：**

**积极影响：**总体上升的参与者人数均值对 VSCode 项目具有积极意义。更多的开发者参与意味着更多的创意、技术和资源的汇聚，能够加速项目的开发进程，促进功能的丰富和完善。不同背景的开发者可以带来多样化的思路和解决方案，有助于 VSCode 更好地适应不同的开发场景和需求，提升项目的整体质量和竞争力。

**需关注的问题：**尽管整体呈上升趋势，但期间的波动也需要关注。波动可能反映出项目在发展过程中面临的一些挑战和不稳定因素，如市场竞争、自身发展策略的调整等。项目团队需要密切关注这些波动背后的原因，以便及时调整策略，保持项目对开发者的持续吸引力，确保参与人数能够稳定增长。

**未来趋势：**参与者人数均值可能会在 2023 年保持在 2150 左右或者略有下降。

**Issue 数量变化折线图：**



**整体趋势分析：**

**创建数量趋势：**issue 创建数量整体呈现波动状态，在 2015 年 11 月-2023 年 3 月期间，数值从 731 到 2561 不等。虽然有起伏，但没有明显的单调递增或递减趋势，表明 VSCode 项目在不同阶段面临的新问题数量不稳定，可能受到项目开发节奏、功能更新、用户群体变化等多种因素影响。例如在某些功能集中开发阶段，新问题会大量涌现，导致创建数量升高。

**关闭数量趋势：**issue 关闭数量同样波动较大，从 369 到 4043 不等。整体上也无明显的单一走向，反映出项目团队处理问题的进度并非匀速，可能根据项目重点、资源分配等因素，在不同时期处理问题的能力 and 效率有所不同。

**对比分析：**

**月度对比：**多数月份中，issue 创建数量高于关闭数量。这意味着在大多数时间段内，项目产生新问题的



速度快于解决问题的速度，可能会导致问题积压。

**特殊月份：**然而，存在一些特殊月份，关闭数量远高于创建数量，如 2018 年 9 月（创建 1749，关闭 3274）和 2019 年 10 月（创建 1999，关闭 4043）。这可能是项目团队在这些时期集中精力处理积压问题，或是进行了大规模的代码审查与优化，使得问题解决效率大幅提升。

**差值分析：**计算每月创建数量与关闭数量的差值，能更清晰地看到问题积压情况。差值为正且数值较大时，表明问题积压增多；反之，差值为负且绝对值较大时，说明积压问题得到缓解。例如 2018 年 9 月差值为-1525，问题积压显著减少；而 2016 年 2 月差值为 244，问题积压有所增加。

**波动原因探讨：**

**项目开发节奏：**当 VSCode 进行重大功能更新或新模块开发时，会引入新的代码逻辑和功能特性，开发者在使用过程中容易发现更多问题，导致 issue 创建数量上升。例如在添加对新编程语言的支持后，可能出现语法高亮、代码补全等方面的问题。而在项目维护阶段，团队专注于修复已有问题，关闭数量可能相对增加。

**团队资源与策略：**项目团队的资源分配和处理问题的策略对 issue 关闭数量影响较大。如果团队在某一时期集中人力和时间解决问题，优化处理流程，如采用更高效的问题跟踪系统、加强开发人员之间的协作等，会提高问题关闭效率，使关闭数量上升。相反，若团队资源分散在多个项目任务中，对 issue 处理的优先级降低，关闭数量可能减少。

**影响：**

**积极影响：**较高的 issue 创建数量反映出社区对 VSCode 项目的关注度高，开发者积极参与项目反馈，有助于项目团队发现更多潜在问题，不断优化和完善项目功能。而 issue 关闭数量的提升，表明团队具备处理问题的能力，能够及时解决用户反馈的问题，提高项目质量和稳定性，增强用户对项目的信任。

**潜在挑战：**持续的创建数量高于关闭数量可能导致问题积压，影响项目的开发进度和用户体验。大量未解决的问题可能使新用户望而却步，也会给开发团队带来较大压力。此外，issue 数量的大幅波动可能意味着项目开发过程不够稳定，需要项目团队更好地规划开发流程、平衡功能开发与问题修复的关系，以确保项目的健康发展。

**未来趋势：**创建数量可能会继续呈现波动上升的趋势。

**巴士系数均值饼状图：**



巴士系数（Bus Factor）是指如果团队中不幸有一定数量的人同时离开项目，项目是否还能继续正常推进的一种衡量指标。较高的巴士系数意味着项目对关键人员的依赖程度较低，即使部分人员离开，项目仍能较好地继续。

### **整体趋势：**

整体体现为前期上升，后期波动。从 2016 年的 784 增长到 2020 年的 1608.17，呈现出明显的上升趋势，表明在这段时间内，VSCode 项目对少数关键人员的依赖程度逐渐降低，项目的可持续性和稳定性在增强，团队结构可能变得更加多元化和分散化，更多的人具备足够的知识和技能来维持项目的推进。然而，2020 年后数据出现波动，2021 年降至 1508.58，2022 年进一步降至 1452.75，这显示项目在保持团队稳定性和知识分散方面面临一定挑战，可能出现了关键人员变动、知识传递受阻等情况。

### **年度变化分析：**

**2016-2017 年：**巴士系数均值从 784 大幅增长到 1400.75，增长幅度显著。这可能是由于在 2016-2017 年间，VSCode 项目团队采取了一系列措施来加强团队建设和知识共享，使得更多成员能够熟悉项目的各个方面，减少了对少数核心人员的依赖。同时，随着项目的发展壮大，吸引了更多有能力的开发者加入，进一步提升了团队整体的能力储备。

**2017-2018 年：**均值从 1400.75 增长到 1412，增长幅度较小但仍保持上升。这表明项目在持续巩固和优化团队结构，虽然增长幅度不大，但团队的稳定性和可持续性在稳步提升。可能在这一年，项目团队继续加强知识传承和技能培养，确保新老成员之间的知识交流顺畅，使得更多成员能够在项目中发挥重要作用。

**2018-2019 年：**巴士系数均值上升至 1456.67，增长较为明显。这可能得益于项目在这一年的进一步发展，更多的开发者参与到项目中，形成了更广泛的知识网络。此外，团队可能实施了一些激励机制，鼓励成员之间分享知识和经验，从而提升了整体的巴士系数。

**2019-2020 年：**数值增长到 1608.17，达到一个较高水平。这或许是因为 VSCode 项目在 2019-2020 年间进行了大规模的开源协作拓展，吸引了全球范围内众多开发者的参与，这些开发者带来了丰富的经验和技能，使得项目知识更加分散，对个别人员的依赖进一步降低。同时，项目可能加强了对开发者社区的建设，促进了开发者之间的交流与合作，提高了团队的整体协作能力和应对人员变动的能力。

**2020-2021 年：**巴士系数均值降至 1508.58，出现下降。这可能是由于项目在 2020 年的快速发展后，部分核心开发者因为各种原因逐渐减少了对 VSCode 项目的投入，而新加入的开发者在知识和技能的积累上尚未完全弥补这一空缺，导致巴士系数有所下降。此外，可能项目在这一年的发展过程中，出现了一些复杂的技术难题或业务调整，使得部分知识和技能集中在少数人员手中，增加了对这些关键人员的依赖。

**2021-2022 年：**均值进一步降至 1452.75，继续下降。这可能暗示项目在这一年面临着更严峻的团队稳定性挑战。可能是因为团队内部的协作机制出现了一些问题，导致知识共享和传承受到阻碍，新老成员之间的沟通不够顺畅，使得项目推进对少数关键人员的依赖程度再次上升。另外，行业竞争的加剧可能导致部分有经验的开发者被竞争对手吸引走，进一步削弱了团队的整体实力。

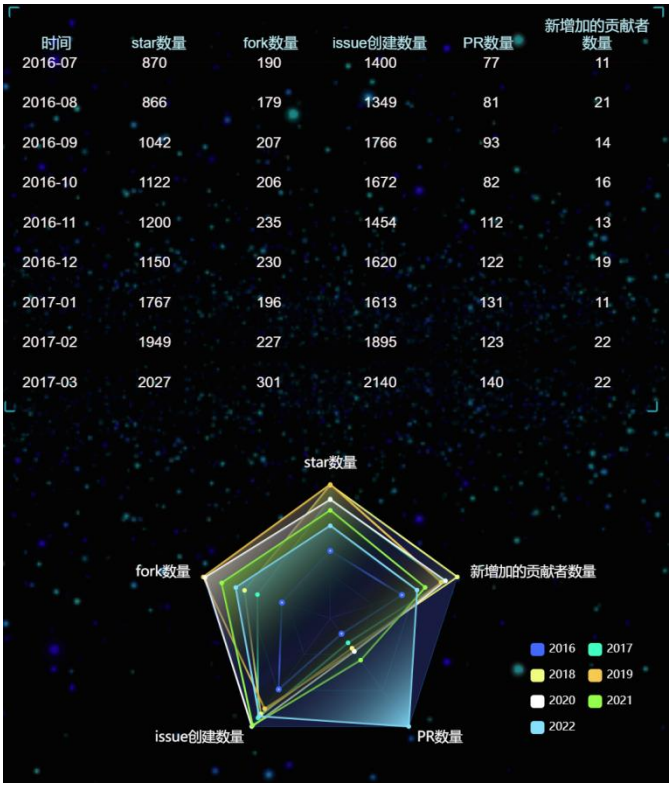
### **影响：**

**积极影响：**在 2016-2020 年巴士系数的上升阶段，较高的巴士系数意味着项目具有较强的抗风险能力。即使遇到部分核心人员离开的情况，项目也能够凭借众多其他开发者的知识和技能继续推进，保障项目的稳定性和连续性。这对于一个开源项目来说至关重要，有助于吸引更多的开发者参与、企业的采用以及用户的信任。

**潜在风险：**2020 年后巴士系数的波动下降则提示项目存在潜在风险。较低的巴士系数意味着项目对少数关键人员的依赖增加，如果这些关键人员因意外情况无法继续参与项目，可能会导致项目进度延迟、技术难题难以解决等问题。这可能影响项目的发展速度和质量，对项目的声誉造成一定损害。因此，项目团队需要关注巴士系数下降的原因，采取相应措施，如加强团队培训、优化知识管理系统、完善人才储备计划等，以提高项目的稳定性和可持续性。

**未来趋势：**可能会延续相对稳定趋势。

**滚动屏：**



**Star 数量：**

项目受欢迎程度维度。

**增减态势：**2016-2019 年，Star 数量从 13264 稳步增长至 26304（总数），这一阶段 VSCode 项目的受欢迎程度显著提升，在开源社区中获得了越来越多开发者的认可与喜爱。然而，自 2019 年起，Star 数量开始逐渐下滑，到 2022 年降至 18196。

**维度意义：**Star 数量直接反映了 VSCode 项目在开源社区中的受欢迎程度。它代表着开发者对项目的整体喜爱与关注程度，是一种对项目价值的直观认可。高 Star 数量意味着项目在技术领域具有较高的知名度

和影响力，吸引着更多开发者的目光，为项目带来更多潜在的参与者和用户。

**未来趋势：**可能会小幅度下降。

#### **Fork 数量：**

项目扩展性与二次开发潜力维度。

**波动情况：**Fork 数量呈现出较为明显的波动。2016 年为 2499，之后逐年上升，2019 年达到峰值 6562，随后在 2020-2022 年逐渐下降，2022 年为 4881。

**维度意义：**Fork 数量体现了 VSCode 项目的扩展性与二次开发潜力。较多的 Fork 数量表明项目的架构和代码具有一定的灵活性和可扩展性，吸引了众多开发者尝试在此基础上进行创新和拓展。这一维度反映了 VSCode 项目在技术层面的吸引力和适应性，为项目的多元化发展提供了可能。

**未来趋势：**可能会呈波动平稳状态。

#### **Issue 创建数量：**

项目发展中的问题与需求反馈维度。

**变化趋势：**整体波动频繁。2016 年为 16117，2017 年增长至 22579，后续年份有增有减，2020-2022 年维持在较高水平。

**维度意义：**Issue 创建数量反映了 VSCode 项目在发展过程中所面临的问题以及开发者对项目的改进需求。当开发者在使用项目过程中遇到功能缺陷、兼容性问题或者有新的功能需求时，他们会通过创建 Issue 来反馈。这一维度为项目团队提供了宝贵的信息，帮助他们了解项目的不足之处，明确改进方向。

**未来趋势：**可能呈现稳定发展趋势。

#### **PR (Pull Request) 数量：**

社区活跃度与开发者贡献维度。

**增长趋势：**自 2016 年的 922 起，PR 数量整体呈增长态势，尤其是在 2022 年急剧增长至 6444。

**维度意义：**PR 数量衡量了社区的活跃度以及开发者对 VSCode 项目的实际贡献程度。开发者通过提交 PR，将自己编写的代码贡献给项目，用于修复问题、添加新功能或优化现有代码。较高的 PR 数量表明社区内开发者积极参与项目开发，不仅关注项目的发展，还愿意投入时间和精力来推动项目的进步。这一维度体现了 VSCode 项目社区的活力和凝聚力，众多开发者的贡献共同促进项目不断迭代和完善。

**未来趋势：**可能会呈现缓慢增长趋势。

#### **新增加的贡献者数量：**

项目人才吸引力与可持续发展维度。

**稳定态势：**数据相对稳定且规模较小，2016-2022 年期间，从 2016 年的 176 到 2022 年的 213，波动幅度不大。

**维度意义：**新增加的贡献者数量反映了 VSCode 项目在吸引新人才方面的能力以及项目的可持续发展潜力。新贡献者的加入为项目带来新的思路、技术和经验，是项目保持创新活力的重要因素。虽然数量相对稳定，但持续有新贡献者加入，表明 VSCode 项目具有一定的吸引力，能够不断吸纳新鲜血液，为项目的长期发展提供支持。

**未来趋势：**可能会呈现波动稳定趋势。

#### **数据关联分析：**

**Star 与其他指标：**前期 Star 数量的增长，同步带动了 Fork 与 PR 数量的上升。这表明 VSCode 项目的高受欢迎程度吸引了更多开发者关注，进而激发他们参与二次开发（Fork）和代码贡献（PR）的热情。然而，后期 Star 数量的下降，可能与 Issue 数量的波动以及市场竞争等因素有关。当 VSCode 项目中存在较多未解决的 Issue 时，可能会影响开发者对项目的好感度，从而导致 Star 数量下滑。

**Fork 与 PR：**Fork 数量的变动对 PR 数量有一定影响。更多的开发者 Fork 项目，意味着有更多潜在的代码贡献者。2019 年 Fork 数量大增后，后续 PR 数量也呈现出上升趋势，说明基于 Fork 项目进行开发的开发者更有可能将自己的改进和创新以 PR 的形式反馈给原项目。

**Issue 与 PR：**Issue 揭示了 VSCode 项目存在的问题和改进需求，而 PR 则是解决这些问题和实现改进的具体行动。当 Issue 数量较多时，往往会激发开发者通过提交 PR 来解决。