# 英文翻译

## Angular vs. React vs. Vue: A 2017 comparison

Deciding on a JavaScript framework for your web application can be overwhelming. Angular and React are very popular these days, and there is an upstart which has been getting a lot of traction lately: VueJS. What’s more, these are just a few of the new kids on the block.So, how are we supposed to decide? A pros-and-cons list never hurts.

为你的web应用程序决定一个JavaScript框架是非常重要的。在最近Angular和React是非常流行的，而且近些时间有个新兴的框架非常引人注意：VueJS。更重要的是，这些这是这个街区里的新的年轻成员。所以我们该如何决定？一份优缺点列表是没有伤害的。

Before we start — SPA or not?

You should first make a clear decision as to whether you need a single-page-application (SPA) or if you’d rather take a multi-page approach.

The starters today: Angular, React and Vue

First, I’d like to discuss lifecycle & strategic considerations. Then, we’ll move to the features & concepts of the three javascript frameworks. Finally, we’ll come to a conclusion.

Here are the questions we’ll address today:

How mature are the frameworks / libraries?

Are the frameworks likely to be around for a while?

How extensive and helpful are their corresponding communities?

How easy is it to find developers for each of the frameworks?

What are the basic programming concepts of the frameworks?

How easy is it to use the frameworks for small or large applications?

What does the learning curve look like for each framework?

What kind of performance can you expect from the frameworks?

Where can you have a closer look under the hood?

How can you start developing with the chosen framework?

Ready, set, GO!

1. Lifecycle & strategic considerations

在我们开始之前—是否是SPA

您应该首先明确您是需要单页面应用程序（SPA）还是希望采用多页面方法。

现在开始：Angular，React，Vue

首先，我想讨论生命周期和战略考虑。 然后，我们将转向三个JavaScript框架的功能和概念。 最后，我们会得出结论。

以下是我们今天要解决的问题：

框架/库有多成熟？

框架可能会在一段时间内出现吗？

他们相应的社区有多广泛和有帮助的？

为每个框架找到开发人员有多容易？

框架的基本编程概念是什么？

为小型或大型应用程序使用框架有多容易？

每个框架的学习曲线是什么样的？

你可以从框架期望什么样的性能？

你在哪里可以仔细观察下引擎盖？

你如何开始用选择的框架进行开发？

预备，准备，开始！

1. 生命周期和战略考虑

1.1历史

Angular是一个基于TypeScript的Javascript框架。 由Google开发和维护，它被描述为“超级英雄JavaScript MVW框架”。 Angular（也称为“Angular 2+”，“Angular 2”或“ng2”）是AngularJS（也称为“Angular.js”或“AngularJS 1.x”）的改写，是其大多不兼容的继承者。 尽管AngularJS（旧版本）最初于2010年10月发布，但它仍然在修复bug等问题 - 新的Angular（sans JS）于2016年9月推出，版本号为2。最新的主要版本为版本4， 版本3被跳过。 Angular被Google，Wix，weather.com，healthcare.gov和Forbes（根据madewithangular，stackshare和libscore.com）等使用。

React被描述为“用于构建用户界面的JavaScript库”。 React最初于2013年3月发布，由Facebook开发并维护，后者在多个页面上使用React组件（但不是作为单页面应用程序）。 根据Chris Cordle撰写的这篇文章，React在Facebook上的使用远远多于Angular在Google上的使用。 React也被Airbnb，Uber，Netflix，Twitter，Pinterest，Reddit，Udemy，Wix，Paypal，Imgur，Feedly，Stripe，Tumblr，Walmart和其他人使用（根据Facebook，stackshare和libscore.com）。

Facebook正在开发React Fiber。 它会改变子例程下的React - 渲染速度应该更快 - 但是在变化之后，框架会向后兼容。 Facebook在2017年4月的开发者大会上讨论了这些变化，并发布了一篇关于新架构的非官方文章。 React Fiber于2017年9月发布了React 16。

Vue是2016年发展最迅速的JS框架之一.Vue将自己描述为“用于构建交互式界面的直观，快速和可组合的MVVM”。该框架于2014年2月首次由前Google员工Evan You（提一下：Evan发布的时候写了一篇关于营销活动和数字的有趣博客文章）。 这是相当成功的，尤其是考虑到Vue在没有大公司支持的情况下获得如同一个人的表演一样的吸引力。 Evan目前拥有一批核心开发人员。 2016年，版本2发布。 Vue被阿里巴巴，百度，Expedia，任天堂，GitLab使用 - 小型项目列表可以在madewithvuejs.com上找到。

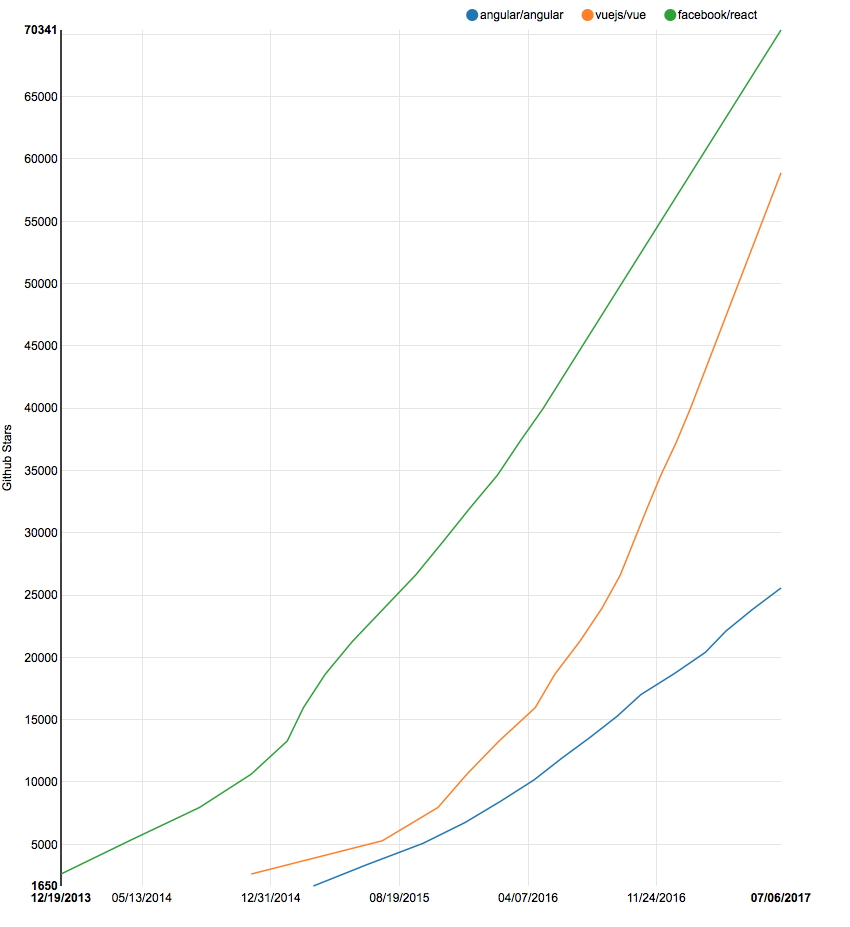
所有三种框架都可以在MIT许可下获得。

React附带特殊的BSD3许可证文件，直到2017年9月。有关专利文件的讨论很多。 如果您对这段历史感兴趣，您可以阅读Github问题讨论，专利文件背后的原因和历史（由Facebook前工程师James Ide撰写），《为什么您不应该害怕》（由Dennis Walsh提供）， 《警告不要用于初创公司》（由RaúlKripalani提供）以及Facebook上关于此主题的陈述：解释React的许可。 无论如何 - 所有这些都不应该像Facebook最终宣布的那样，React将获得MIT许可。

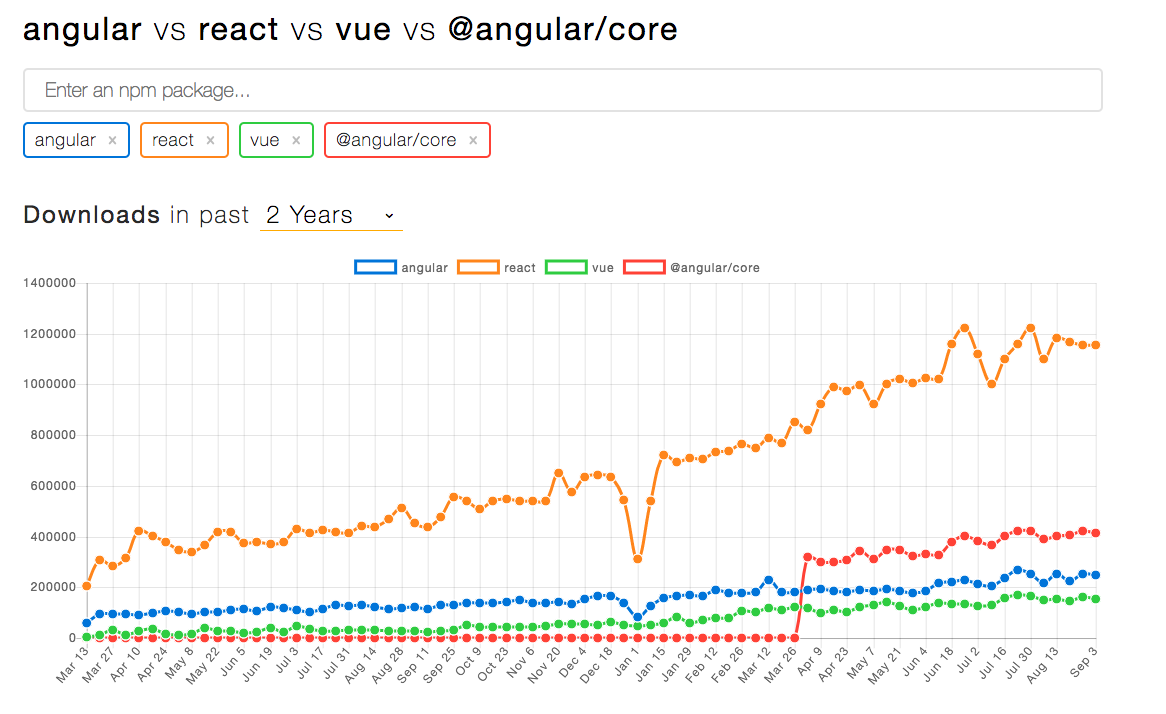
1.2核心发展

如前所述，Angular和React得到了大公司的支持和使用。 Facebook，Instagram和Whatsapp将其用于他们的页面。 谷歌在很多项目中使用它，例如，新的Adwords UI是使用Angular＆Dart实现的。 再次，Vue是由一群通过Patreon和其他赞助方式支持其工作的个人实现的。 您可以自行决定这是积极的还是消极的。 MatthiasGötzke认为Vue的小团队是一个好处，因为它会导致更简洁，更少的过度设计的代码/ API。

让我们来看看一些统计数据：Angular在他们的团队页面上列出了36个人，Vue列出了16个人，而React没有团队页面。 在Github上，Angular拥有> 25,000星级和463名贡献者，React拥有> 7万星级和> 1,000个贡献者，而Vue拥有近6万名明星和120名贡献者。 您还可以查看Github Stars的Angular，React和Vue历史。 再次，Vue似乎趋势非常好。 根据bestof.js的数据，在过去三个月中，Angular 2的平均每天获得31颗星星，React 是74颗星星，同时Vue.JS是107颗星星。

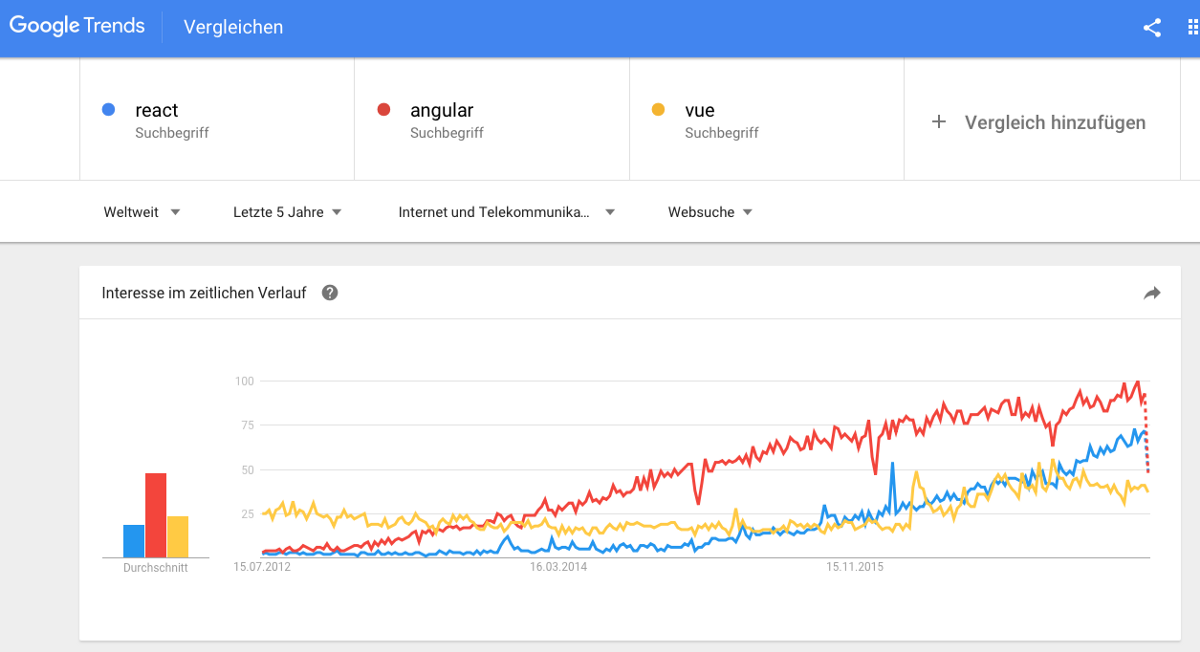


更新：感谢Paul Henschel给出的npm趋势。 它们显示给定npm软件包的下载次数，对比Github星星数量的纯粹看法更有帮助：



1.3市场生命周期

由于各种名称和版本，很难在Google趋势中比较Angular，React和Vue。 一种近似的方法是在“Internet＆technologies”类别中搜索。 结果如下：



好吧。 Vue并非在2014年之前创建的 - 所以这里有些不对劲。 La Vue是法语中的“观看”，“视线”或“意见”的意思。 也许就是这样。 “VueJS”和“Angular”及“React”的比较也不公平，因为VueJS几乎没有任何结果。

那么，我们试试其他的东西吧。 ThoughtWorks的技术雷达给人留下了技术如何随着时间演变的良好印象。 Redux正处于采用阶段（将在项目中采用！），并且它在许多ThoughtWorks项目中具有无可估量的价值。 Vue.js正处于试用阶段（试用！）。 它被描述为Angular的轻量级和灵活的替代品，学习曲线较低。 Angular 2处于评估阶段 - 它已被ThoughtWork团队成功使用，但尚未获得强有力的推荐。

根据上一次的Stackoverflow 2017调查，React被67％的受访开发者喜爱，AngularJS则是52％。 “没有兴趣继续开发”这一项AngularJS（48％）要高于React（33％）。 在这两项调查中，Vue都没有进入前十名。 接下来是statejs.com调查，比较了“前端框架”。 最有趣的事实是：React和Angular有100％的知名度，而23％的受访者不了解Vue。 关于满意度，React因“可重复使用”而获得92％的满意度，而Vue是 89％，Angular 2仅获得65％。

另一个客户满意度调查呢？ Eric Elliott于2016年10月开始评估Angular 2和React。 只有38％的受访者表示会再次使用Angular 2，而84％的人会选择再次使用React。

1.4长期支持和迁移

正如Facebook在其设计原则中所述，React API非常稳定。 还有一些脚本可以帮助您从当前的API转移到更新的API：检查react-codemod。 迁移非常简单，并且没有（需要）作为长期支持的版本。 在这篇Reddit文章中，人们注意到升级从未成为一个真正的问题。 React团队撰写了一篇关于他们的版本控制计划的博客文章。 当他们添加弃用警告时，他们会在下一个主要版本中的行为更改之前保留当前版本的其余部分。 目前没有更改为新的主要版本的计划 - v14于2015年10月发布，v15于2016年4月发布，而v16尚未发布日期。 正如React核心开发人员最近指出的，升级不应该是一个问题。

关于Angular，有一篇关于版本控制和发布Angular的博客文章，从版本2发布开始。 每六个月会有一次重大更新，并且至少有六个月的弃用期（两个主要版本）。 在文档中标记了一些实验性的API，并使用较短的弃用期。 目前还没有官方声明，但根据这篇文章，Angular团队已经宣布了从Angular 4开始的长期支持版本。这些版本将在下一个主要版本发布之后至少支持一年。 这意味着至少在2018年9月之前Angular 4将有bug修复和重要补丁的支持。 在大多数情况下，将Angular从v2更新到v4与更新Angular依赖关系一样简单。 Angular还提供了有关是否需要进一步更改的信息指南。

Vue 1.x到2.0的更新过程对于一个小应用程序来说应该很容易 - 开发人员团队声称90％的API保持不变。 在控制台上有一个很好的升级 - 诊断迁移 - 辅助工具。 一位开发人员指出，从v1到v2的更新在大型应用程序中很无趣。 不幸的是，关于下一个主要版本或关于LTS版本计划的信息没有清晰的（公共）路线图。

还有一件事：Angular是一个完整的框架，并提供了很多捆绑在一起的东西。 React比Angular更灵活，您可能会使用更独立，不稳定，快速移动的库 - 这意味着您需要自行处理相应的更新和迁移。 如果某些包不再被保留，或者某些其他包在某些时候成为事实上的标准，这对于框架来说也是不利的。

1.5人力资源和招聘