



作者 mindwind (/users/50355ce4a05b) 2016.06.05 20:27

写了94734字，被369人关注，获得了892个喜欢

(/users)

+ 添加关注 (/sign_in)

适合程序员的画图技法

字数2813 阅读2280 评论7 喜欢107



之前写一些技术文章时，经常有读者留言问我是用什么工具画图的。其实我感觉他们很可能问错了问题，因为我曾经为了画好图尝试过各种不同的画图工具软件，但最后发现能不能画好图和工具的关系并不大。

为何？

程序员不是写代码的么，为什么需要画图？很多程序员会认为写好代码就好，画好图有什么用？程序员成为架构师后是不是就天天画架构图，成为了所谓的 PPT 架构师？如上这些疑问，好几年前也曾让我困惑过。

在一篇文章《在首席架构师眼里，架构的本质是...》提到了一个架构师能力模型（下图），结合我自己的经历和经验，这个能力模型针对架构师这个岗位来说还是比较符合的。

简 (/)

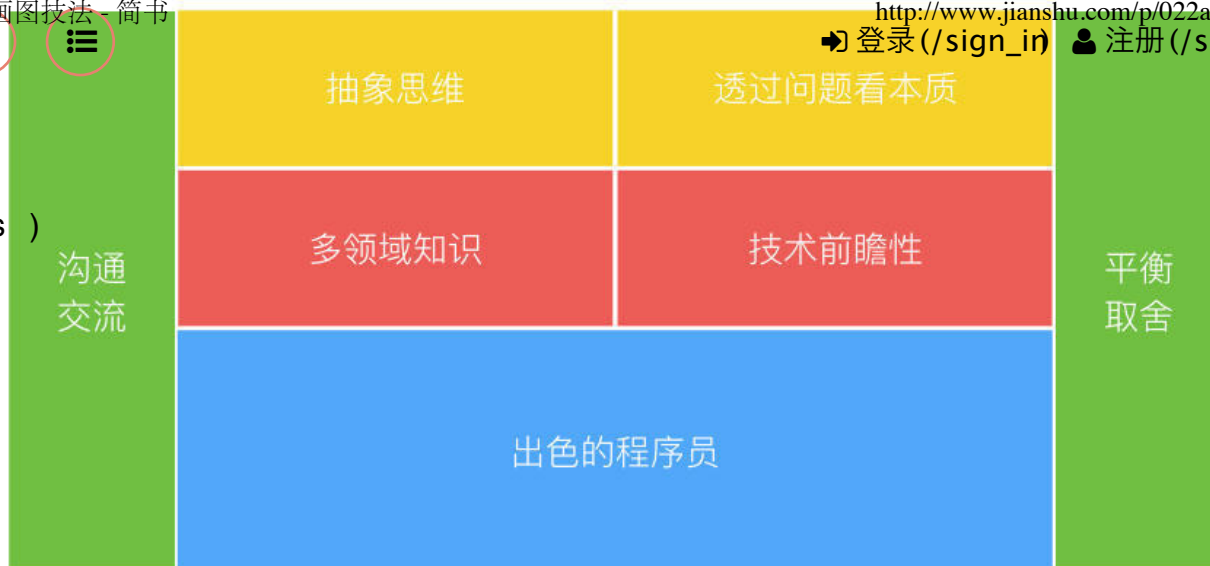
⋮

(/collections)

☎

(/apps)

登录 (/sign_in) 注册 (/sign_up)



一个程序员做了很多项目，写了多年代码逐步成长为一名出色的程序员。从上面的能力图中可以看出，一个出色的程序员离一个架构师还差好多其他方面的能力。我们以前以为程序员积累了足够经验就会自然成长为一名架构师，但其实架构师并不是程序员自然成长的一个延续，只是因为架构师的工作相对管理岗而言离程序员和技术更近，所以我们对它产生了这样的错觉。不断在「出色的程序员」这块领地内不断的耕耘和出色下去会让你成为该垂直领域内的技术专家，这才是程序员自然成长的延续。

因而，程序员出色到了一定程度后想成长为一名架构师，就需要看看能力模型中的其他方面。而掌握好画图技法，我觉着至少对其中的抽象思维、沟通交流、平衡取舍与透过问题看本质都有帮助。至于多领域知识和技术前瞻性这两方面好像确实和画图的相关性不强，但如果多领域知识不限于程序技术领域，画图也算一个领域的知识吧。

今天这个时代的地图软件我们都用过，一个国家、一个城市、一个街区，地图软件总是在不同的抽象维度上来展示地图。而对于一个复杂的软件系统，也需要类似的不同抽象维度，系统的全貌，不同子系统间的关联和交互，子系统内部模块间的接口和调用，某个关键实现点的处理流程。一个架构师应该可以在这些不同的抽象维度上把系统或系统的一部分清晰地描绘出来。

当在不同抽象维度上描绘了系统的各个重要方面，我们才可以更好的发现和找到系统的症结。如果解决系统的问题就像走迷宫，你是直接钻进去反复尝试寻找出路，还是站在更高的维度俯视迷宫再找到最佳的问题解决路径。这就是透过问题看本质领域一个方面的体现。

关于沟通交流，俗话说，有图有真相，哦，不对，是一图胜千言。有些程序员写技术文档啪啦啪啦的写一大堆，有时真不如一张清晰的架构图或交互图让人更快速清晰的理解到。在对系统有了抽象全面的多维度呈现，清晰准确的交流，直击了问题本质，那么正

A



如何?

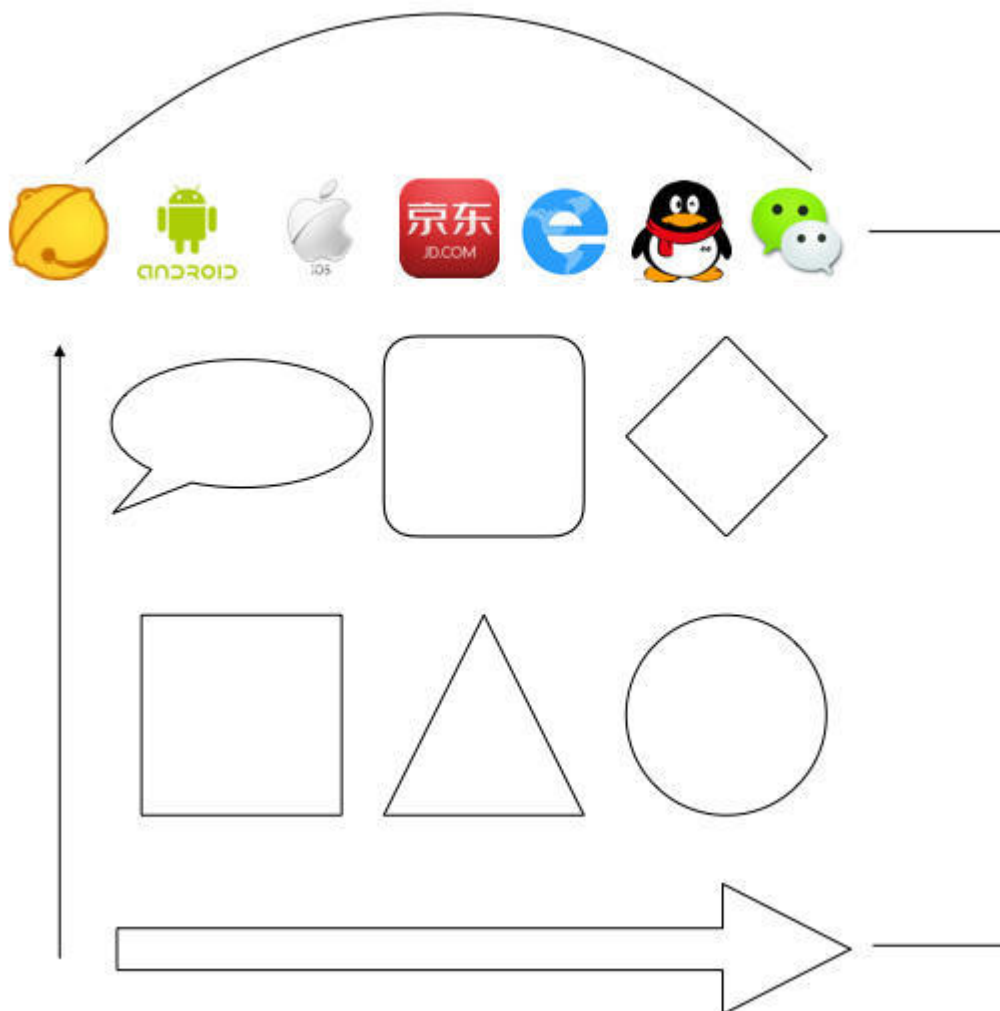
上一节探讨了画好图会带来什么样的收益，这一节我们看下如何画好图？画一个清晰易懂的技术架构或交互流程的说明图例需要什么专门的绘图知识与技巧么？另外为了画好图会花费大量的时间么？

在过去几年关于如何画好图这个课题上我做了好些摸索和实践，想取得效率（画图花费的时间不会多于比用文字来描述同样的内容更多）和效果（图例表达的效果应该比文字描述更好）的平衡，在这个过程中收获了下面一些基本认知和自我感觉还不错的实践方式。

图形

我画技术图例时只会使用一些最基础的图形，比如：矩形、圆、三角、菱形、气泡、箭头，这些最基本的图形几乎所有的画图软件都会自带的，所以工具的依赖性很低，但选择的效率很高。当然如果有时为了表达和一些著名外部系统间的交互，这些著名外部系统可能都有各自著名的 Logo 图标，也会直接使用它们的 Logo 图标。

像下面图示，就是我常用的一些画图图形元素。



简

颜色

(/)

:::

(/collections)

📱

(/apps)

有时系统的组成比较复杂，只用基本图形不足以表达所有不同的系统组成部件，这时就需要用颜色来区分了。那么下一个问题就来了，该用哪些颜色呢？我的答案是使用大部分人觉得美的颜色。那大部分人喜欢什么颜色呢？当然我没有作过任何调查，全凭脑袋拍的。我觉得大部分会觉得彩虹是美的，所以我一般用得颜色就是彩虹七色外加两种经典色：黑、白。这样就有九种颜色加上几种基本图形，可以组合出几十种表达不同组件的图形元素，基本够用了。

彩虹七色包括：红、橙、黄、绿、青、蓝、紫。但七种颜色的选择也是有优先级，在一本讲设计的书中 *Designing with the Mind in Mind*（中文译名《认知与设计》，其实我觉得译名没有原名那么的有感觉）提出了下面一些色彩使用准则：

- 使用饱和度、亮度以及色相区分颜色，确保颜色的高反差，因为人的视觉是为边缘反差而优化的。
- 使用独特的颜色，因为人最容易区分的颜色包括：红、绿、黄、蓝、白和黑。
- 避免使用色盲无法区分的颜色对，比如：深红—黑，深红—深绿，蓝色—紫色，浅绿—白色。
- 使用颜色之外的其他提示，对有颜色视觉障碍的人友好，而且也增强了可理解性。
- 避免强烈的对抗色，比如：红黑，黄黑

所以你看为什么交通灯是：红、黄、绿。为什么乔布斯选择这三个颜色作为 Mac OS X 操作系统中所有应用窗体的按钮颜色，这也是暗合人类的视觉认知原则的。所以我现在多选择是白底、黑字、黑色线条、色块优先选择：红、绿、黄、蓝，实在不够用了才会选择：橙、青、紫。

当然红有好多种红，绿有好多种绿，我用哪种？看下图所示，给出了 RGB 三原色的配色数值（别相信自己的眼睛，不同显示器上看到的会有差异，作为程序员需要精确点）。至于为什么是这个选择，后面再说。

A



简
(/)



(/collections)



(/apps)



审美

除了基本的图形和颜色选择之外，另外一个关注点是审美。审美对最终的效果呈现有很大影响，这得感谢苹果总设计师 Jonathan Ive 把大众的审美倾向全部带入到扁平化时代，所以实际我只需要把图形弄得扁平，去掉立体、阴影什么的，看起来就还不错了。



几何？

探讨了如何，我们接着看看几何。此「几何」不是数学里的几何，而是曾几何时，我们想象中很麻烦的事原来如此简单。掌握画图技法到底代价几何？又价值几何呢？

三年前，我画的技术图示（来自以前一个分享 PPT）大概是下面这样的，总是觉得不好，不太满意，却又不知道不好在哪里，该怎么改进。然后就归罪于工具不好用，从一开始用 Visio 画，后来尝试了 Mac 下的专业绘图工具 OmniGraffle，觉得太复杂后又找到个在线绘图网站 draw.io，感觉还可以，但由于是国外网站，访问效率不太好没多久就放弃了。之后需要做一些 Slide 演示时，用了 Mac 下的 Keynote（相当于 Win 下的 PPT），需要画技术图示时想如果直接在 Keynote 里画最省事了，然后就开始用 Keynote 画了。

A



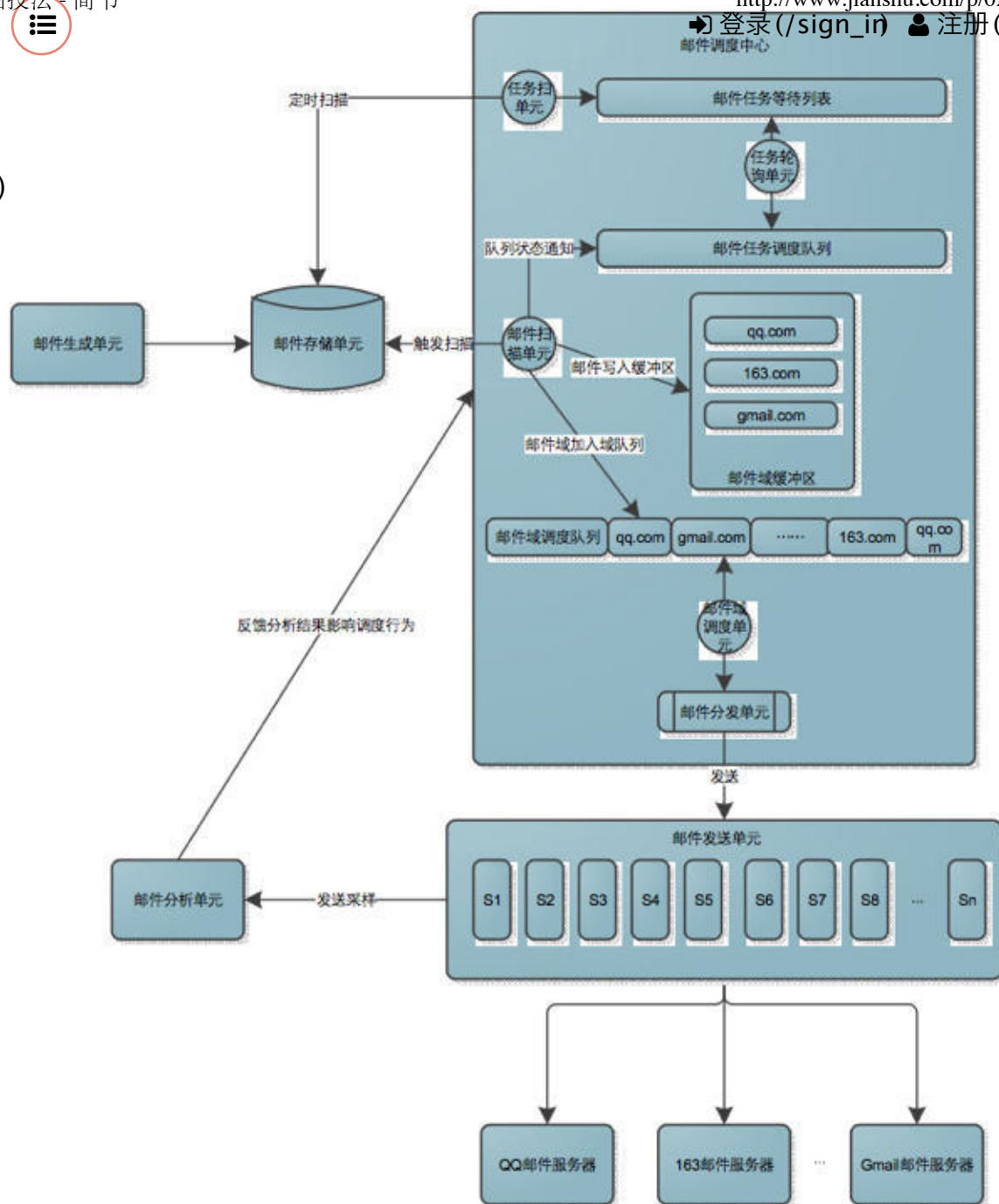
简 (/)

⌘

(/collections)

📱

(/apps)

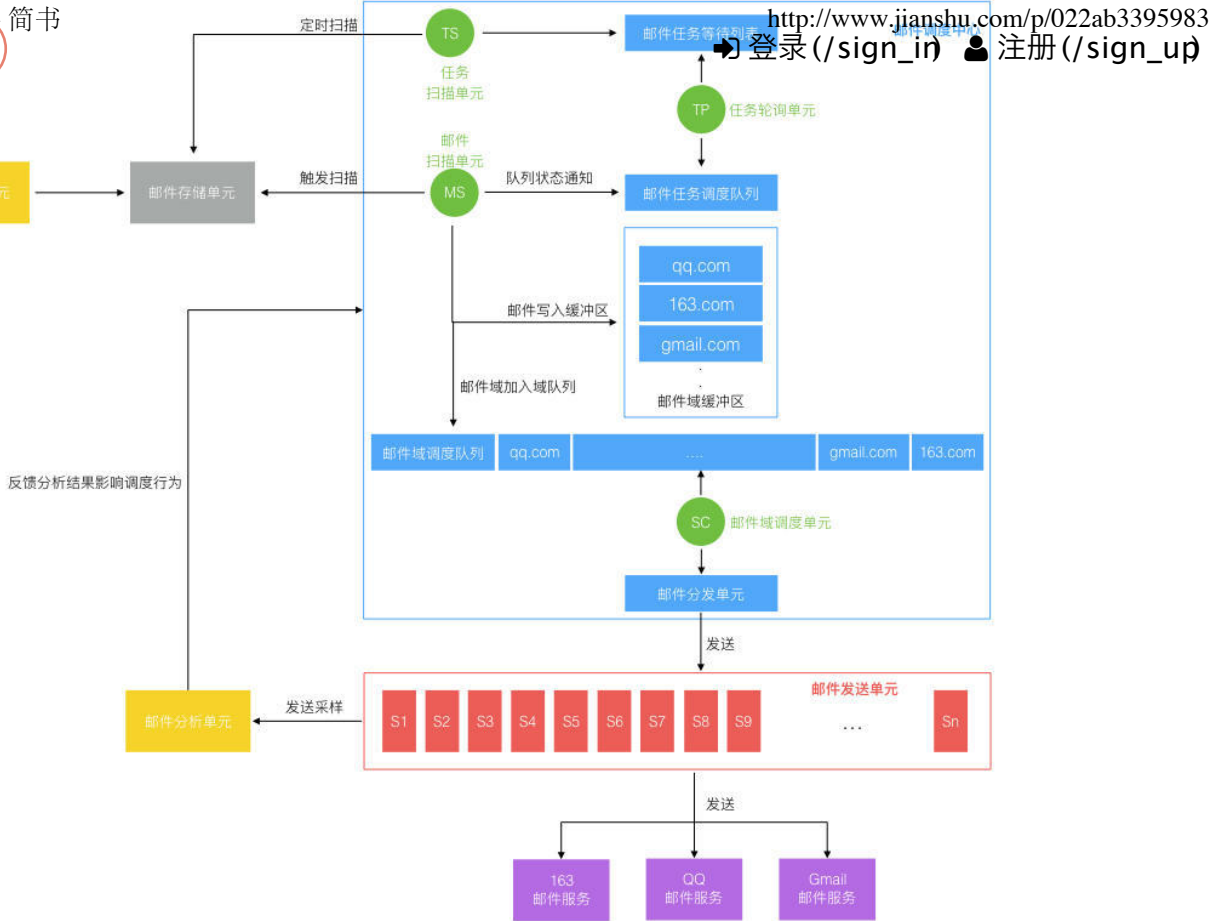


前面写到颜色选择时，提到为什么就选这种红而不是其他红，其实这就是 Keynote 在白色背景下默认提供的调色板色块选择。作为程序员提供一个软件的默认参数，通常的道理是这个参数可能在大部分场景下都是最优的。所以我倾向于认为 Keynote 的默认色块是这个背景下的一些最优纯色选择，而且我自己肉眼看起来也还感觉不错，这样就省了打开更高级的配色参数界面，绘图的效率又更高了，代价也更小了。所以按这个指导原则，重新画下上面这个技术图示大概就像下面这样，花费的时间绝对不会比画上面那个更多，但呈现效果自我感觉确实要好多了。

A



简 (/)
 (/collections)
 (/apps)



所以，学会使用一种简单的软件，使用简单的图形和配色，在最有效率的情况下画出一幅效果还不错的图例，也是很有价值的。当然很多程序员会认为只有写出的代码才有价值，其实这里可能忽视了一个大部分程序员都认同的观点：代码也是写给人看的。程序员不会认为一份机器能运行而人很难看懂的代码是好代码，而画好图能更好的帮助你去思考代码的组织 and 呈现方式。

本文只是介绍了一种极简的绘制技术图例的技法，毕竟我们画图只是为了追求讲清楚一个技术或展示一个系统，不需要考虑任何多余的艺术性。最低的代价，还不错的效果，在效率和效果之间取得性价比最高的平衡。

...

程序员的工作性质和习惯更关注实现而忽略呈现，有时酒香也怕巷子深啊，何况还未必是茅台酒。

推荐拓展阅读 (/sign_in) © 著作权归作者所有

简 (/)



打赏支持



(/users/0d9a530160211a8)

登录 (/sign_in) 注册 (/sign_up)



(/collections)



(/apps)

喜欢 | 107

分享到微博 分享到微信 更多分享 ▾

7条评论 (按时间正序 · 按时间倒序 · 按喜欢排序)

添加新评论 (/sign_in)



p了个f (/users/40fb2e2a8047)

5楼 2016.06.06 23:42 (/p/022ab3395983/comments/2668886#comment-2668886)

写的太好了

喜欢(1)

回复



小黑眼圈 (/users/71f422bb224f)

5楼 2016.06.07 11:23 (/p/022ab3395983/comments/2672800#comment-2672800)

一个即将成为程序猿的读者

喜欢(0)

回复

mindwind (/users/50355ce4a05b): @小黑眼圈 (/users/71f422bb224f) 😊😊

2016.06.07 14:52 (/p/022ab3395983/comments/2675143#comment-2675143)

回复

小黑眼圈 (/users/71f422bb224f): @mindwind (/users/50355ce4a05b) 哈哈 我会好好学习的

2016.06.08 08:13 (/p/022ab3395983/comments/2683365#comment-2683365)

回复

添加新回复



向右奔跑 (/users/54b5900965ea)

5楼 2016.06.07 18:02 (/p/022ab3395983/comments/2677301#comment-2677301)

可视化展现非常重要!

喜欢(0)

回复

A



齐刘海姑娘在DevStore (/users/9f5f02c04052)

7楼 2016.06.08 13:26 (/p/022ab3395983/comments/2687175#comment-2687175)




简书 (/)
📑 喜欢(0)


📑 (/collections)
田靖荣 (/users/84739978a5b6)
📅 2016-06-10 03:57 (/p/022ab3395983/comments/2702503#comment-2702503)


📱 写得挺好
(/apps) 📄 喜欢(0) 回复

登录后发表评论 (/sign_in)

被以下专题收入，发现更多相似内容：

 **首页投稿 (/collection/bDHhpK)**
玩转简书的第一步，从这个专题开始。想上首页热门榜么？好内容想被更多人看到么？来投稿吧！如果被拒也不要灰心哦~入选文章会进一个队...
📄 64488篇文章 (/collection/bDHhpK) · 106096人关注 [+ 添加关注 \(/sign_in\)](#)

 **程序员 (/collection/NEt52a)**
简书程序员大本营 投稿须知： 1.本专题仅收录与程序有关的文章。 2.请在代码框里写代码，尽量保证可看性。关注简书官...
📄 15252篇文章 (/collection/NEt52a) · 96462人关注 [+ 添加关注 \(/sign_in\)](#)

 **每天写1000字 (/collection/723de9bac3cd)**
如果你要练习每天写一千字，这就是你打卡的地方。
📄 100840篇文章 (/collection/723de9bac3cd) · 77707人关注 [+ 添加关注 \(/sign_in\)](#)