

# 开篇词 | 从这里开始，带你走上硅谷一线系统架构师之路

2019-04-15 蔡元楠 来自北京

《大规模数据处理实战》



你好，我是蔡元楠。目前是 Google Brain 的软件工程师。

在接下来的 4 个月时间里，我会与你一起探索大规模数据处理的世界。

在开始我们的系统性学习之前，我想先和你分享两个我亲历的故事，借此告诉你，我为什么要开这个专栏。

2014 年，我刚开始在美国找工作，在一次面试中，面试官让我解释一下 C++ 的 smart pointer 和 string view。我完全回答不上来。

当时我面露难色怀疑人生，难道我的前半生学的是“假的” C++ 吗？

回想学习经历，虽然我在一个还可以的高校里学习了 C++ 课程，考试成绩也是 90 分以上。但我的学习资料只有按 C++ 98 标准编写的教材，和当时流行的 CSDN 论坛文章。

而当时的教材里根本没有提到过 smart pointer 和 string view。

没想到工业界早就进入了 C++ 0x/11 标准甚至是 C++ 17（当时是试验标准）。我后悔没有学习紧跟时代的最新技术，而信息的不对称就会造成巨大的认知偏差。

那学习“最新技术”一定就是好事情吗？我想通过第二个故事来回答这个问题。

2017 年，我帮 Google Ventures（Google 的风险投资基金，会寻找并帮助优秀初创公司）在投的初创公司做导师，那时候经常参加一些对方公司的技术架构评审。

一次评审中，对方的技术 VP 眉飞色舞地介绍他们的技术框架和××大厂一样，罗列了 Kylin 和 Tornado 等一些时髦的技术名词。

因为我并不了解提到的几个技术，就好奇地问为什么 Kylin 适合他们团队，Kylin 为他们解决了哪些独特的问题？

他当时的回答并没能说服我和别的同事：“因为××大厂也在用，这就是未来的技术方向”。

看得出来，这位技术 VP 还没有真的搞清楚使用一个技术的原因。

学会用一个技术只是第一步，最重要的是要追问自己：

这个技术解决了哪些痛点？

别的技术为什么不能解决？

这个技术用怎样的方法解决问题？

采用这个技术真的是最好的方法吗？

如果不用这个技术，你会怎样独立解决这类问题？

如果没有这些深层次的思考，你就永远只是在赶技术的时髦而已，不会拥有影响他人的技术领导力。

事实上在 Google，类似这样的“灵魂追问”每天都在发生。

这里敢于打碎任何权威，所有的技术设计都是从问题出发。每一个工程师都会独立思考究竟什么是最佳方案，而不是照搬现有结论。

正如乔布斯所说，过去的点最终会连成线。这两个经历让我深感有使命去帮助更多技术同行，比如第一个故事里的我，或者是第二个故事里的初创公司 VP。我想设计一个专栏，去解决故事里的问题。

### **第一，我想要介绍硅谷最前沿技术和真实的案例。**

比如，在大规模数据处理领域，MapReduce 或者 Apache Storm 的不少设计理念已经无法胜任最新的挑战。

所以，我会介绍最新的知识，例如框架层面的前后端分离理念，和批处理流处理统一的思想。

### **第二，我不想只停留在照本宣科的层面。**

正如上文所说，学会用一个或者两个技术是不够的。

更重要的是，我会剖析技术框架产生的原因和它们解决的问题。这样当下一次你再碰到相似的问题时，就不用照搬别人的方法。

## **为什么写大规模数据处理？**

为什么会选择大规模数据处理这个主题呢？并不是因为我觉得这个主题受众多、销量好，相反，我认为大部分人都还没有正确理解数据处理技术，常常见到的误区有如下几种。

### **第一，低估了数据处理的重要性。**

因为我在 Google Brain 的 AI 应用领域工作，切身感受到过，没有高质量的数据处理的话，人工智能是只有人工没有智能的。

Google 也曾在很长的一段时间里低估过数据处理。

例如，在语义理解上，Google 认为自己有最多的搜索文本数据，最好的算法，那就一定能把语义理解做的最好。

可是到 2016 年左右，一个名不见经传的德国小公司却一举超过了 Google，大家都很惊讶。后来发现原来他们凭借的是高质量的数据标注和处理。

## **第二，低估了数据处理工程师在组织架构上的重要性。**

许多工程师都喜欢自嘲自己的工作“搬砖”，事实也正是如此。

包括我在内，很多人的工作内容都避不开数据的搬运和处理。把数据从这个格式处理成那个格式，把数据从这个数据库搬到那个数据库，这个服务器搬到那个服务器，这个客户端搬到那个客户端。

可能连你自己都还没有意识到，即使是一个写前端的工程师，他的很多工作还是数据处理。

大数据领域泰斗级人物 Jesse Anderson 曾做过一项研究，一个人工智能团队的合理组织架构，需要 4/5 的数据处理工程师。很不幸，很多团队没有认识到这一点。

## **第三，低估了数据处理规模变大带来的复杂度。**

我把这个专栏定位在“大规模”数据处理，因为我想着重在数据规模变大时需要的技术思想。

很多人可能还没有遇到过“大规模”数据的问题，容易把问题想简单了。

我在 Google 面试过很多优秀的候选人，应对普通的编程问题，他们能够用算法和数据结构解决得很好。可是当我追问数据规模变大时怎么设计系统，他们的回答却并不让人满意。

当你的产品从 1 万用户到 1 亿用户，技术团队从 10 个人到 1000 个人，之前的方法还能奏效吗？

#### 第四，高估了上手数据处理的难度。

一方面我们需要认识到大规模的数据处理是有复杂的因素的。但另一方面，我想在这个专栏里教会你，有了正确的工具和技术理念，现在上手数据处理并不困难。

在 Google，我见到很多应届生来了半年后也能轻松应对上亿的数据量。

我给开篇词起名为《从这里开始，带你走上硅谷一线系统架构师之路》，就是为了给你设计切实可操作的学习路径，让你比别人更准确深入地掌握实用的大规模数据处理技术，最终通往硅谷一线系统架构师的水平。

因此，我们的学习路径会是这样的。

第一部分，先会用原汁原味**最实际的硅谷一线大厂的案例**，向你解释 MapReduce 为什么不能应对最新的技术挑战。然后我会从实际的问题出发，从头开始引导你怎样从顶层设计一个数据处理框架。

第二部分，同样是结合实战案例，来讲解在数据处理框架的使用和设计中必需的一些基础知识。这些案例紧贴应用，可能就是**你的团队明天会碰到的问题**。

第三、第四部分深入拆解了 Apache Spark 和 Apache Beam。不仅会用实际的案例教会你如何使用，还要教会你为什么它们这么设计。你会发现它们的设计其实大致和第一部分的顶层设计是一致的。**下一次，即使这个世界一无所有，你也能构建类似的框架解决一系列问题。**

第五部分按 Google T6 级别设计，是带着代码的真枪实弹的架构设计。毫不夸张地说，能完整掌握第五部分的思想精髓，你就能比肩硅谷一线大规模数据处理架构师。

第六部分着重培养你的技术远见。因为，**是否能现在就开始准备应对 10 年后人类社会的技术挑战，是你拉开与别人差距的重要一站。**

## 《大规模数据处理实战》课程目录



- ① 从这里开始，带你走上硅谷一线系统架构师之路

## 模块一：直通硅谷大规模数据处理技术

- ② 为什么 MapReduce 会被硅谷一线公司淘汰？
- ③ MapReduce 后谁主沉浮：怎样设计下一代数据处理技术？
- ④ 大规模数据处理初体验：怎样实现大型电商热销榜？

## 模块二：实战学习大规模数据处理基本功

- ⑤ 分布式系统（上）：如何用服务等级协议 SLA 来评估你的系统？
- ⑥ 分布式系统（下）：架构师不得不知的三大指标
- ⑦ 如何区分批处理还是流处理？
- ⑧ Workflow 设计模式：让你在大规模数据世界中君临天下
- ⑨ 发布 / 订阅模式：流处理架构中的瑞士军刀
- ⑩ CAP 定理：三选二，架构师必须学会的取舍
- ⑪ Lambda 架构：Twitter 千亿级实时数据分析架构背后的倚天剑
- ⑫ Kappa 架构：使用 Kafka 锻造的屠龙刀

## 模块三：抽丝剥茧剖析 Apache Spark 设计精髓

- ⑬ Spark 的独有优势：为什么世界需要 Spark？
- ⑭ 弹性分布式数据集：Spark 大厦的地基为何如此设计（上）
- ⑮ 弹性分布式数据集：Spark 大厦的地基为何如此设计（下）

- 15 Spark SQL: Spark 数据查询的利器
- 16 Spark Streaming: Spark 的实时流计算 API
- 17 如何用 Spark DataFrame API 进行实时数据分析?
- 18 WordCount: 从零开始运行你的第一个 Spark 应用
- 19 综合案例实战: 加州房屋信息的线性回归模型 & 成年人收入的预测模型
- 20 更多、更快、更好、更省: 实例解析 Spark 应用程序性能优化
- 21 深入对比 Flink 与 Spark: 帮你的系统设计两开花

## 模块四: Apache Beam 为何能一统江湖

- 22 Apache Beam 的前世今生
- 23 站在 Google 的肩膀上学习 Beam 编程模型
- 24 PCollection: 为何 Beam 要如此抽象封装数据?
- 25 Beam 数据转换操作的抽象方法
- 26 Pipeline: Beam 如何抽象多步骤的数据流水线?
- 27 Pipeline I/O: Beam 数据中转的设计模式
- 28 如何设计好一个 Beam Pipeline?
- 29 如何测试 Beam Pipeline?

## 模块五: 决战 Apache Beam 真实硅谷案例

- 30 Apache Beam 实战冲刺: Beam 如何 run everywhere?

30 Apache Beam 实战/平利: Beam 如何 run everywhere?

31 WordCount Beam Pipeline 实战

32 Beam Window: 打通流处理的任督二脉

33 横看成岭侧成峰: 再战 Streaming WordCount

34 Amazon 热销榜 Beam Pipeline 实战

35 Facebook 游戏实时流处理 Beam Pipeline 实战 (上)

36 Facebook 游戏实时流处理 Beam Pipeline 实战 (下)

## 模块六: 大规模数据处理的挑战与未来

37 5G 时代, 如何处理实时超大规模物联网数据?

38 大规模数据处理在深度学习中如何应用?

39 从 SQL 到 Streaming SQL: 突破静态数据查询的次元

40 大规模数据处理未来之路

在刚开始动笔写专栏的时候, 我就在想, 什么样的人会是这个专栏的目标读者呢?

直到专栏快上线, 我写下这篇开篇词, 我才真正定义读者的标签——应该就是跟我一样渴望成长的人。是的, 我和你一样, 都渴望成长——渴望知识的成长, 渴望经验的成长, 渴望财富的成长。

所以我想把这个专栏设计成一份共同的成长规划, 而不是一本死板的教材。

正如我在开头的小故事里所说的, 这个世界没有谁是绝对的权威。

我希望你每一期都能在留言栏里质疑、提问和讨论。这些互动能帮助我和别的同学一起提高。



最后，我期待和你一起开始学习，共同成长！

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

## 精选留言 (118)



风之伤 置顶

2019-04-15

学习这专栏需要什么基础知识

作者回复：很好的问题。设计时并没有对读者基础作任何假设，所以碰到任何技术概念，都会举例解释一下。可能需要些编程基础会学的快一点，专栏里一些示例代码是Python。如果有哪里觉得不清楚的后面可以再提出来，我们可以再调整。谢谢提问！

共 2 条评论 >

👍 32



暮光之城 置顶

2019-04-16

老师你好，你上文提到的AI落地的基础是大规模的数据和高质量的标注，目前能满足的这个条件是否只有一些超大规模的一线互联网公司。那是否意味着大数据处理也只有在这些公司中才能发挥真正的价值，那对于在小型互联网公司工作的程序员，学习大数据处理的意义在哪里呢？

作者回复：我觉得这个问题很好啊。我一部分同意大规模的互联网公司天生数据量大一点。另一方面，1. 对于公司来讲小型互联网公司甚至是传统企业，并不是不需要数据处理技能，而是他们还没有从数据中挖掘business insight的意识，没有数据驱动决策的意识，甚至没有收集数据的意识。举个我工作中见到的例子，比如有奶牛的农户几十年来根本不知道什么是数据，但是当我们帮他们细致的搜集牛的每天的数据，比如饮食，运动，作息，产奶，他们能从中找到最经济最优的饲料投放。2. 对于个人来讲一定要看长期的职业发展，公司会从小变大，职位会从低变高，当你能更多影响决策当你数据量变多，当你跳槽之后，数据的处理能力都是至关重要的。我们可以继续就这个问题探讨！

💬

👍 24



钱 置顶

2019-04-16

这个技术解决了哪些痛点？  
别的技术为什么不能解决？

这个技术用怎样的方法解决问题？

采用这个技术真的是最好的方法吗？

如果不用这个技术，你会怎样独立解决这类问题？

做了太多的CRUD，越来越觉得自己没什么核心竞争力，好的问题解决思路都具有通用性，希望跟着牛人走上一程，随便聊聊涨涨见识也好。

大数据多大算大？

处理量大的思路，感觉主要就是分治的思想？

需要多台机器来扛，如果单台机器没有容量和性能局限是不是就不需要各种大数据的处理思想了？

我不是从事大数据工作的，只是好奇，之前面试时也遇到过类似处理大量数据存储的问题，存起来容易，一台机器容量不够，就两台，两台不行就继续加，不过怎么查询？怎么统计分析就费劲了？

希望，听到不一样的声音喝令人惊叹的文婷姐姐思路。

作者回复：首先感谢你对这门课程的支持！你所提出的问题都非常好！

大数据多大算大，其实我觉得对数据量并没有一个硬性的上限或者下限的要求。一个有几Pb的数据集是大数据，那一个只有几条数据的数据集算吗？其实也算。我们看大数据背后的本质其实是希望我们不要纠结于数据量的多少，抽象出来看的话其实大数据平台希望能有处理无限大或者无限小数据集的能力的。

第二个问题你已经看到问题的本质了。没错，你所说的分治的思想其实就是MapReduce里面Map方法的一个抽象。

第三个问题我们可以这样看，在单台机器下，所有的数据处理操作其实都是由CPU完成的。而站在更高的角度上看，一个计算机的集群我们也可以把这个集群看作是“一台计算机”，而底下每台计算机都是一个“CPU”。只不过在计算机集群这种分布式的环境下我们还要保证其它例如数据一致性这种单机环境下不需要特别考虑的东西。

你说的一台不够就多加机器来处理这种操作是有专业名词的，叫作Horizontal Scaling。我会在第二章里面有实例讲解。

希望在后面的课程里还能看到你的提问留言，让我们一起学习进步！



15



2019-04-17

我是一个蹉跎了多年的三流程序员，目前放弃了一切在脱产学习大数据，过程很辛苦，但我渴望成长，很幸运遇到了蔡老师，希望能跟随您的轨迹，通过这段时间的学习达到自己新的高度，to get my life back!

作者回复: 你好啊坤剑同学! 我觉得不要给自己定下一个标签, 每个人都是慢慢学习成长起来的, 包括我自己。我也希望你能通过我的课程, 学习到数据处理上, 架构设计上的思想精髓。大数据在技术平台上虽然日新月异, 但其实很多背后的设计思想都是融会贯通的, 当你掌握了本质, 很多东西学习起来就得心应手了。希望这也对你日后的学习有所帮助, 我们一起加油!



👍 8



**hua168** 置顶

2019-04-15

老师, 学习这个需要什么知识为提前?

作者回复: 很好的问题, 另一个同学也提到了类似问题。我们在内容设计时并没有对读者对知识背景作任何假设, 所以即使一些基础的技术概念都会举例解释一下(如果你会了可能会觉得啰嗦)。有一些任何语言的编程经验会看起来快一点, 因为有一些示例代码是Python的。但是设计类型的案例, 我不觉得有特别的技术要求。希望你后面继续跟踪一下吧, 如果有哪些讲的不清楚, 或者解释的过多, 后面可以调整内容。谢谢提问!



👍 6



**大王叫我来巡山**

2019-04-16

很多时候公司淘汰一个人的原因不是因为他年龄大了, 而是他的技术没有随着年龄增长

作者回复: 很多时候是因为那个公司傻逼, 没有意识到程序员的价值在于经验, 下次解决相似的问题知道哪些路可以哪些路不行。年轻干的动只是一小部分。



👍 45



**Sapph**

2019-04-15

高效的数据处理和高质量的标注是数据分析的前提, 在AI战场厮杀的不仅仅是复杂的算法, 还要依托于背后的大数据处理能力。看了目录, 内容很干货。

话说, 这又是一个亲身上阵自己录音频的老师, 作者本人读出来的文章是有灵魂的~

作者回复: 谢谢鼓励



👍 21



**coder**

2019-04-15

请问老师Google T6是什么概念？

作者回复: 相当于阿里P9吧

共 2 条评论 >

👍 17



**硅谷居士**

2019-04-19

补充关于 Google T6 的说明。“如今的谷歌工程师们身处从 1 级开始的庞大存在链当中。最底层的 1 级代表 IT 技术支持人员，2 级为大学新生，3 级则通常是拥有硕士学位的员工。达到 4 级往往需要数年实践周期，或者拥有博士学位。大多数员工的职业晋升都停留在第 5 级。6 级为工程师，代表谷歌公司前 10% 的卓越人才，他们的技术能力直接决定着项目的成败。7 级则代表着拥有长期实践经验的 6 级工程师。8 级为首席工程师，他们的工作与各主要产品及基础设施紧密关联。9 级为杰出工程师，在很大程度上已经成为一种尊称。而 10 级代表谷歌院士，这是一种终身性的荣誉头衔。谷歌院士是全球所在领域内毋庸置疑的佼佼者，Jeff 与 Sanjay 则是谷歌高级院士——该公司最初也是唯二的 11 级员工。”



👍 16



**听水的湖**

2019-04-15

又是一个Google大佬，大佬是南方人吧。带着耳机听的，专栏用作者本人的声音真是很良心了，更有代入感。不过感觉有点难度，希望文章内容有深度的同事能兼顾一下宽度.....学渣倒地不起.....

作者回复: 哈哈，确实是南方口音。是会兼顾各方面同学的需求，不过难度和宽度并不冲突。比如在第二篇里面，我们分析一个案例，会看数据量100的时候怎么解决，1亿又是怎么解决。我希望展现一个问题解决的立体全景。



👍 12





2019-04-15

灵魂追问需要通过阅读文档和源码，并加上自己实践和思考才能够回答。

作者回复: 的确是互为补充，专栏讲解的案例是有限的，但我希望在有限的案例里把思考方式讲清楚。另外相比文档，这里设计的案例会更实际一点。也欢迎你把自己专栏外的学习收获在这里分享。



👍 11



而立斋

2019-04-15

听起来老师很安静，喜欢您的分享。鼓励自己多留言

作者回复: 是的海明同学，希望多看到你的留言



👍 7



3SKarl

2019-04-18

元楠老师，您好！请问在阅读开源框架（如hadoop，hive...）的源码方面有没有什么经验可以说说呢？我有以下疑惑：

1. 当我打开一个类开头就是二三十个变量的时候就不知该怎么读下去
2. 有些方法看着有几百行代码，当我从头到尾都弄明白每一段的逻辑之后感觉并没有学到什么，阅读源码的意义究竟是为了什么呢
3. 想请教正确的阅读源码的方式是怎样的

作者回复: 我和你感同身受，纯粹从头读很容易睡着。我觉得比较有效的方式是带着问题去看，比如我今天帮hadoop修一个bug，我去看究竟怎么回事；比如我今天想搞懂hadoop的fingerprint究竟怎么工作的，怎么配置嵌入的。其实这种阅读代码能力在工作中也很重要，比如在工作中，每个组的代码量级都和hadoop之类差不多，漫无目的去读很难找到方向。

希望对你有帮助，如果有收获欢迎分享给朋友。



👍 6

渡码



2019-04-16

您觉得研究Mapreduce或者hdfs框架有必要吗

作者回复: Mapreduce和hdfs都解决了很多问题。但是后面一篇会提到mapreduce本身的局限性。我看一个技术不会拘泥于“现在还有必要学吗”。任何技术产生都是有原因的, 肯定能解决一些问题, 肯定也有被更好技术取代的一天, 但更重要的是明白技术怎么产生怎么设计的。



👍 6



李

2019-04-15

深度拷问灵魂中技术的本质, 做一个有技术远见的工程师

作者回复: 是, 李同学, 你理解的很对



👍 6



buptfb

2019-04-16

很认同作者的观点, 没有高效的数据处理, 哪来的算法的快速迭代?

作者回复: 谢谢! 希望后面的课程能继续看到你的留言!



👍 5



kylin

2019-04-16

目前在一家外包公司写hive, 做批处理。很想了解一下大数据处理的全部过程。目前国内对于技术名词的操作让我迷失了方向, 像elasticsearch做即席查询, flink一统流处理和批处理, 而且很多写hive的工程师写习惯了sql, 也不想用编程语言写框架提供的各种算子。请问老师, 做大数据处理, 应该更关注哪些方面, 目前写了1个多月的hive, 感觉都是按照业务分层处理, 我这边是DWD层, 无法了解到很全面的业务流程, 有些迷茫。感谢老师

作者回复: 首先感谢你的提问! 大规模数据处理其实涉及了很多方面, 例如从原始数据集到结果数据集的转换涉及了数据处理Workflow的设计, 处理架构上的设计, 甚至到数据库Schema的设计优化, 数据服务API的设计都会影响大数据处理的性能好坏。像hive的话更加多是属于data analysis了, 因为已经是在结果数据集上做分析, 自己需要做的数据转换不会需要特别多。

我在第二章会结合硅谷这边的实际业务场景介绍大规模数据处理上的设计架构思想，希望能看到你的留言与提问，让我们一起进步！



👍 5



**阿童木**

2019-04-16

大神，有机会可以写一篇您是如何从毕业生一步步走到google T6的，谢谢！



👍 5



**jiji**

2019-04-16

作为数据行业从业者表示深深同意开篇词，非常实用的提纲挈领。现在的人工智能如果还是只关注算法那说明应用者太浮躁了，喜欢这份专栏

作者回复：谢谢你的肯定！让我们一起成长



👍 5



**SpanningWings**

2019-04-16

我觉得老师的五个问题问得好：这个技术解决哪些痛点，为什么别的技术不能解决，它是如何解决的，是最好的方法吗，不用它我如何来独立解决。回答类似问题在我司别名叫讲清楚，实在很不容易。谢谢。

作者回复：很高兴看到你也是类似追问



👍 5