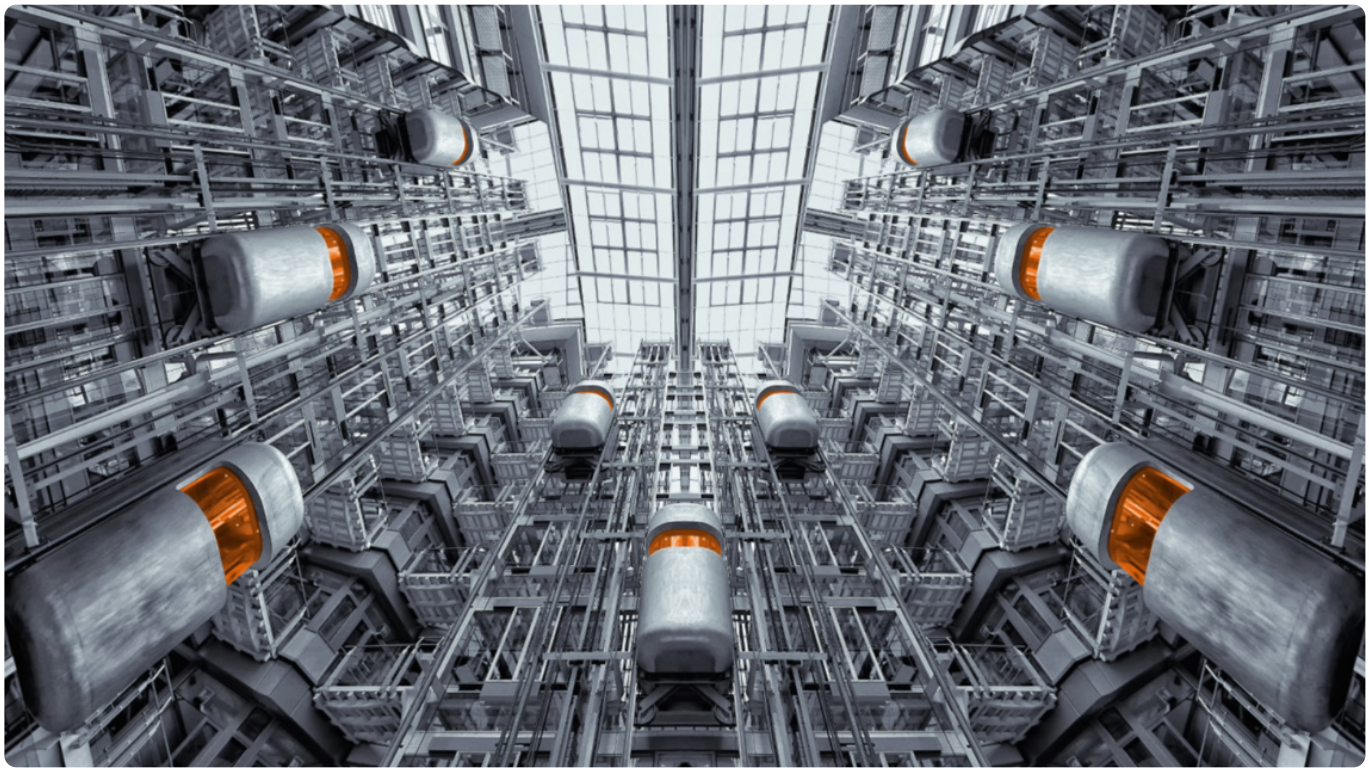
04 | 移动计算比移动数据更划算

2018-11-06 李智慧

从0开始学大数据 进入课程 >



讲述：李智慧
时长 10:31　大小 4.83M

大数据技术和传统的软件开发技术在架构思路上有很大不同，大数据技术更为关注数据，所以相关的架构设计也围绕数据展开，如何存储、计算、传输大规模的数据是要考虑的核心要素。

传统的软件计算处理模型，都是“输入 -> 计算 -> 输出”模型。也就是说，一个程序给它传入一些数据也好，它自己从某个地方读取一些数据也好，总是先有一些输入数据，然后对这些数据进行计算处理，最后得到输出结果。

但是在互联网大数据时代，需要计算处理的数据量急速膨胀。一来是因为互联网用户数远远超过传统企业的用户，相应产生了更大量的数据；二来很多以往被忽视的数据重新被发掘利用，比如用户在一个页面的停留时长、鼠标在屏幕移动的轨迹都会被记录下来进行分析。在稍微大一点的互联网企业，需要计算处理的数据量常常以 PB 计（10^{15} Byte）。

正因为如此，传统的计算处理模型不能适用于大数据时代的计算要求。你能想象一个程序读取 PB 级的数据进行计算是怎样一个场景吗？一个程序所能调度的网络带宽（通常数百 MB）、内存容量（通常几十 GB ）、磁盘大小（通常数 TB）、CPU 运算速度是不可能满足这种计算要求的。

那么如何解决 PB 级数据进行计算的问题呢？

这个问题的解决思路其实跟大型网站的分布式架构思路是一样的，采用分布式集群的解决方案，用数千台甚至上万台计算机构建一个大数据计算处理集群，利用更多的网络带宽、内存空间、磁盘容量、CPU 核心数去进行计算处理。关于分布式架构，你可以参考我写的《大型网站技术架构：核心原理与案例分析》这本书，但是大数据计算处理的场景跟网站的实时请求处理场景又有很大不同。

网站实时处理通常针对单个用户的请求操作，虽然大型网站面临大量的高并发请求，比如天猫的“双十一”活动。但是每个用户之间的请求是独立的，只要网站的分布式系统能将不同用户的不同业务请求分配到不同的服务器上，只要这些分布式的服务器之间耦合关系足够小，就可以通过添加更多的服务器去处理更多的用户请求及由此产生的用户数据。这也正是网站系统架构的核心原理。

我们再回过头来看大数据。**大数据计算处理通常针对的是网站的存量数据**，也就是刚才我提到的全部用户在一段时间内请求产生的数据，这些数据之间是有大量关联的，比如购买同一个商品用户之间的关系，这是使用协同过滤进行商品推荐；比如同一件商品的历史销量走势，这是对历史数据进行统计分析。**网站大数据系统要做的就是将这些统计规律和关联关系计算出来，并由此进一步改善网站的用户体验和运营决策。**

为了解决这种计算场景的问题，技术专家们设计了一套相应的技术架构方案。最早的时候由 Google 实现并通过论文的方式发表出来，随后根据这些论文，开源社区开发出对应的开源产品，并得到业界的普遍支持和应用。这段历史我们在前面的“预习”中已经讨论过了。

这套方案的核心思路是，既然数据是庞大的，而程序要比数据小得多，将数据输入给程序是不划算的，那么就反其道而行之，**将程序分发到数据所在的地方进行计算，也就是所谓的移动计算比移动数据更划算。**

有一句古老的谚语，说的是“当一匹马拉不动车的时候，用两匹马拉”。听起来是如此简单的道理，但是在计算机这个最年轻的科技领域，在很长一段时间里却并没有这样做。当一台

计算机的处理能力不能满足计算要求的时候，我们并没有想办法用两台计算机去处理，而是换更强大的计算机。

商业级的服务器不够用，就升级小型机；小型机不够用，就升级中型机；还不够，升级大型机，升级超级计算机。

在互联网时代之前，这种不断升级计算机硬件的办法还是行得通的，凭借摩尔定律，计算机硬件的处理能力每 18 个月增强一倍，越来越强大的计算机被制造出来。传统企业虽然对计算机的处理需求越来越高，但是工程师和科学家总能制造出满足需求的计算机。

但是这种思路并不适合互联网的技术要求。Google、Facebook、阿里巴巴这些网站每天需要处理数十亿次的用户请求、产生上百 PB 的数据，不可能有一台计算机能够支撑起这么大的计算需求。

于是互联网公司不得不换一种思路解决问题，当一台计算机的计算能力不能满足需求的时候，就增加一台计算机，还不够的话，就再增加一台。就这样，由一台计算机起家的小网站，逐渐成长为百万台服务器的巨无霸。Google、Facebook、阿里巴巴这些公司的成长过程都是如此。

但是买一台新计算机和一台老计算机放在一起，就能自己开始工作了吗？两台计算机要想合作构成一个系统，必须要在技术上重新架构。这就是现在互联网企业广泛使用的负载均衡、分布式缓存、分布式数据库、分布式服务等种种分布式系统。

当这些分布式技术满足互联网的日常业务需求时，对离线数据和存量数据的处理就被提了出来，当时这些分布式技术并不能满足要求，于是大数据技术就出现了。

现在我们来看，移动计算程序到数据所在位置进行计算是如何实现的呢？

1. 将待处理的大规模数据存储在服务器集群的所有服务器上，主要使用 HDFS 分布式文件存储系统，将文件分成很多块（Block），以块为单位存储在集群的服务器上。

2. 大数据引擎根据集群里不同服务器的计算能力，在每台服务器上启动若干分布式任务执行进程，这些进程会等待给它们分配执行任务。

3. 使用大数据计算框架支持的编程模型进行编程，比如 Hadoop 的 MapReduce 编程模型，或者 Spark 的 RDD 编程模型。应用程序编写好以后，将其打包，MapReduce 和 Spark 都是在 JVM 环境中运行，所以打包出来的是一个 Java 的 JAR 包。

4. 用 Hadoop 或者 Spark 的启动命令执行这个应用程序的 JAR 包，首先执行引擎会解析程序要处理的数据输入路径，根据输入数据量的大小，将数据分成若干片（Split），每一个数据片都分配给一个任务执行进程去处理。

5. 任务执行进程收到分配的任务后，检查自己是否有任务对应的程序包，如果没有就去下载程序包，下载以后通过反射的方式加载程序。走到这里，最重要的一步，也就是移动计算就完成了。

6. 加载程序后，任务执行进程根据分配的数据片的文件地址和数据在文件内的偏移量读取数据，并把数据输入给应用程序相应的方法去执行，从而实现在分布式服务器集群中移动计算程序，对大规模数据进行并行处理的计算目标。

这只是大数据计算实现过程的简单描述，具体过程我们会在讲到 HDFS、MapReduce 和 Spark 的时候详细讨论。

小结

移动程序到数据所在的地方去执行，这种技术方案其实我们并不陌生。从事 Java 开发的同学可能有过用反射的方式热加载代码执行的经验，如果这个代码是从网络其他地方传输过来的，那就是在移动计算。杀毒软件从服务器更新病毒库，然后在 Windows 内查杀病毒，也是一种移动计算（病毒库）比移动数据（Windows 可能感染病毒的程序）更划算的例子。

大数据技术将移动计算这一编程技巧上升到编程模型的高度，并开发了相应的编程框架，使得开发人员只需要关注大数据的算法实现，而不必关注如何将这个算法在分布式的环境中执行，这极大地简化了大数据的开发难度，并统一了大数据的开发方式，从而使大数据从原来的高高在上，变成了今天的人人参与。

思考题

互联网应用系统架构中有一种重要架构原则是尽量使用无状态的服务，不同服务实例之间不共享状态，也就是不持有数据，用户请求交给任何一个服务实例计算，处理的结果都是一样

的，为什么要这样设计？这种架构有什么好处？

欢迎你写下自己的思考或疑问，与我和其他同学一起讨论。

上一篇　预习 03 | 大数据应用领域：数据驱动一切

下一篇　05 | 从RAID看垂直伸缩到水平伸缩的演化

精选留言 (66)

写留言

ASCE1885
2018-11-06

54

无状态服务的主要好处是服务间无需同步状态或者数据，便于扩缩容。

展开

作者回复: 是的

不求
2018-11-07

24

针对于思考题的一些思考：
分布式架构的原则：尽量使用无状态的服务，不同服务实例之间不共享状态，也就是不持有数据。。。

这个问题我是这样考虑的，什么是无状态的服务？为什么需要它？它是在怎样的情况下...
展开

作者回复: 思考深入👍

jon
2018-11-06

9

移动数据的成本高且一台机器负载不了，所以用计算找数据的方式，让数据平均分布在集群中，把软件包发生到各台机器上并行对相对较小的数据计算，计算结果再合并起来。
无状态的原因是程序包可以负载均衡的分发到任何最优的节点进行计算，计算时宕机了也可以在另一台创新计算，各个节点都是一样的环境
展开

高国君
2018-11-15

5

互联网使用无状态服务的原则，主要目的是为了实现服务的低耦合高内聚的目标。一旦低耦合高内聚，服务就可以动态伸缩（放/换哪个机器上都可以运行），同时，也引申出另外一个要求：如何实现服务的发现、编排、调度？这就涉及一些微服务框架了。
对于大数据，如果不是无状态服务，那弹出一个服务，还要调它依赖的服务，那么这个处理过程，会有非常多非必要的开销，也有非常多的隐患。譬如，被调用的服务怎么保证...
展开

陈柏林
2018-11-07

4

每台服务器原本都不带有程序，但是调度服务器为处理服务器分发任务之后，处理服务器就执行任务并检查是否有该程序，没有就下载，下载之后从指定路径中读取数据进行处理，处理好之后统一存放处理结果，大概的执行流程是这样吗？

作者回复: 是的

没有枫树的...
2018-11-07

3

有点抽象，能举例子说明一下什么是无状态的什么是有状态的吗？移动计算不就是那个节点都进行相同的计算吗？是说计算过程需要依赖其他节点的数据叫有状态？

展开

作者回复: 大数据的分布式都是有状态的，这个无状态是网站架构里的无状态应用，抱歉思考题跳跃有点大。

关于无状态应用可以参考我的《大型网站技术架构》这本书

五岳寻仙
2018-11-06

3

刚接触这个领域，认知还比较浅显。我觉得成千上万的机器之间通信会很耗费时间，无状态能保证减少机器之间的耦合，提高效率。

作者回复: 即使有状态的服务器，也尽量做到share nothing，尽少通信，不然n对n通信，通信量巨大

老男孩
2018-11-06

2

期待后面的文章。无状态的服务是内聚的低耦合的，同一个用户的请求可以被分发到不同服务节点上进行处理。伸缩自如。

作者回复: 是的

公号-代码...
2018-11-06

2

无状态服务有利于提升分布式系统的可伸缩性。

展开

作者回复: 是的

丶请叫我七...
2019-01-08

1

大数据引擎根据集群里不同服务器的计算能力，在每台服务器上启动若干分布式任务执行进程，这些进程会等待给它们分配执行任务....
请问：分布式任务执行程序是指什么？

展开

作者回复: 继续学习，会有答案～

苏锐 | ...
2018-11-13

1

在 hadoop 提出的年代是百兆网卡为主，所以搬数据会出现网络带宽的瓶颈。今天，万兆网卡已经成为标配，data locality 已经没那么重要了吧，请教李老师。同时，下发计算，scale-out 的能力会受 data node 数量的限制，Facebook 等厂商开始实践存储与计算分离的架构，计算资源更容易在大任务执行时去临时扩展。希望和李老师探讨

展开

cellardoor
2018-11-06

1

无状态的好处：
1，伸缩性更好，应用之间无需同步状态，方便伸缩。
2，幂等，应用服务器之间都是对等的，请求落在哪里都可以得到相同的响应。
3，可用性更好，有状态，意味着有数据丢失的可能，在某些情况下，状态不一致容易造成可怕的结果。...

展开

作者回复: 赞

关于2幂等，应用服务器无状态，但是应用服务器依赖的数据库或者其他服务器可能有状态，无状态的应用服务器也无法幂等了。需要看情况。

杨杰。
2018-11-06

1

最近在自学一些大数据的组件，有个问题想请问一下老师，在不改写sqoop源码的情况下，有没有什么好的方法避免数据倾斜？

公号-代码...
2018-11-06

1

无状态服务有利于提升分布式系统的可伸缩性。

展开

作者回复: 是的

Zach_
2018-11-06

1

【自问自答】什么是无状态服务？无状态服务的好处是什么？无状态服务有没有缺点？

John Lau
2018-11-06

1

無狀態服務可以免除需要同步狀態的情況，系統可以跟據情況把計算自由分發到不同的機器上，而不用考慮同步問題。實現起來也會簡單很多。
但我有個問題:現實場景就是有很多時候需要有狀態，有沒有系統的思考方法，把需要狀態的工作方式，改寫成不需要狀態的工作方式?

作者回复: 状态委托给其他服务器，缓存或者数据库。
有状态是逻辑需要，可以委托出去，但是很难不要状态。

掌心童话
2019-04-22

将逻辑运算与存量数据分离。可以分而制之， 如逻辑运算可以优化时，无需考虑影响数据的存放；增强数据的一致性，减少存量数据再利用的难度（需要再经过至少一次ETL）等。

展开

songgoogle

2019-04-19

如果做不到，那就转移状态到计算框架之外，这样就会涉及到一致性，zookeeper

gkb111

2019-04-11

无状态服务，我（数据）就在这里，你索取也好，存储也罢，我始终在这里。哈哈😁

不似旧日

2019-01-21

对移动计算的理解: 当我自己想吃KFC时我自己去店里吃,这叫做移动数据;当1000个人要吃KFC时我让kfc来我家做这叫做移动计算。举个栗子,不接受反驳