岁寒

(https://lvwenhan.com/) **任何事情，从现在开始做，都不晚！**

性能之殇（七）-- 分布式计算、超级计算机与神经网络共同的瓶颈

2018-11-22 / 阅读数：14545 / 分类： 操作系统 (https://lvwenhan.com/sort/36)

分布式计算是这些年的热门话题，各种大数据框架层出不穷，容器技术也奋起直追，各类数据库（Redis、ELasticsearch、MongoDB）也大搞分布式，可以说是好不热闹。分布式计算在大热的同时，也存在着两台机器也要硬上 Hadoop 的“面向简历编程”，接下来我就剖析一下分布式计算的本质，以及我的理解和体会。

分布式计算的本质

分布式计算来源于人们日益增长的性能需求与落后的 x86 基础架构之间的矛盾。恰似设计模式是面向对象对现实问题的一种妥协。

x86 服务器

x86 服务器，俗称 PC 服务器、微机服务器，近二十年以迅雷不及掩耳盗铃之势全面抢占了绝大部分的服务器市场，它和小型机比只有一个优势，其他的全是缺点，性能、可靠性、可扩展性、占地面积都不如小型机，但是一个优势就决定了每年 2000 多亿美元的 IDC 市场被 x86 服务器占领了 90%，这个优势就是**价格**。毕竟有钱能使磨推鬼嘛。

现有的分布式计算，无论是 Hadoop 之类的大数据平台，还是 HBase 这样的分布式数据库，无论是 Docker 这种容器排布，还是 Redis 这种朴素分布式数据库，其本质都是因为 x86 的扩展性不够好，导致大家只能自己想办法利用网络来自己构建一个宏观上更强性能更高负载能力的计算机。

x86 分布式计算，是一种新的计算机结构。

基于网络的 x86 服务器分布式计算，其本质是把网络当做总线，设计了一套新的计算机体系结构：

- 每一台机器就等于一个运算器加一个存储器
- master 节点就是控制器加输入设备、输出设备

x86 分布式计算的弱点

上古时代，小型机的扩展能力是非常变态的，到今天，基于小型机的 Oracle 数据库系统依旧能做到惊人的性能和可靠性。实际上单颗 x86 CPU 的性能已经远超 IBM 小型机用的 PowerPC，但是当数量来到几百颗，x86 服务器集群就败下阵来，原因也非常简单：

1. 小型机是专门设计的硬件和专门设计的软件，只面向这种规模（例如几百颗 CPU）的计算
2. 小型机是完全闭源的，不需要考虑扩展性，特定的几种硬件在稳定性上前进了一大步
3. x86 的 IO 性能被架构锁死了，各种总线、PCI、PCIe、USB、SATA、以太网，为了个人计算机的便利性，牺牲了很多的性能和可靠性
4. 小型机使用总线通信，可以实现极高的信息传递效率，极其有效的监控以及极高的故障隔离速度
5. x86 服务器基于网络的分布式具有天然的缺陷：
    1. 操作系统决定了网络性能不足
    2. 网络需要使用事件驱动处理，比总线电路的延迟高几个数量级
    3. PC 机的硬件不够可靠，故障率高
    4. 很难有效监控，隔离故障速度慢

x86 分布式计算的基本套路

Google 系大数据处理框架

2003 年到 2004 年间，Google 发表了 MapReduce、GFS（Google File System）和 BigTable 三篇技术论文，提出了一套全新的分布式计算理论。MapReduce 是分布式计算框架，GFS（Google File System）是分布式文件

系统，BigTable 是基于 Google File System 的数据存储系统，这三大组件组成了 Google 的分布式计算模型。

Hadoop、Spark、Storm 是目前最重要的三大分布式计算系统，他们都是承袭 Google 的思路实现并且一步一步发展到今天的。

MapReduce 的基本原理也十分简单：将可以并行执行的任务切分开来，分配到不同的机器上去处理，最终再汇总结果。而 GFS 是基于 Master-Slave 架构的分布式文件系统，其 master 只扮演控制者的角色，操控着所有的 slave 干活。

Redis、MongoDB 的分布式

Redis 有两个不同的分布式方案。Redis Cluster 是官方提供的工具，它通过特殊的协议，实现了每台机器都拥有数据存储和分布式调节功能，性能没有损失。缺点就是缺乏统一管理，运维不友好。Codis 是一个非常火的 Redis 集群搭建方案，其基本原理可以简单地描述如下：通过一个 proxy 层，完全隔离掉了分布式调节功能，底层的多台机器可以任意水平扩展，运维十分友好。

MongoDB 官方提供了一套完整的分布式部署的方案，提供了 mongos 控制中心，config server 配置存储，以及众多的 shard（其底层一般依然有两台互为主从强数据一致性的 mongod）。这三个组件可以任意部署在任意的机器上，MongoDB 提供了 master 选举功能，在检测到 master 异常后会自动选举出新的 master 节点。

问题和瓶颈

人们费这么大的劲研究基于网络的 x86 服务器分布式计算，目的是什么？还不是为了省钱，想用一大票便宜的 PC 机替换掉昂贵的小型机、大型机。虽然人们已经想尽了办法，但还是有一些顽固问题无法彻底解决。

master 失效问题

无论怎样设计，master 失效必然会导致服务异常，因为网络本身不够可靠，所以监控系统的容错要做的比较高，所以基于网络的分布式系统的故障恢复时间一般在秒级。而小型机的单 CPU 故障对外是完全无感的。

现行的选举机制主要以节点上的数据以及节点数据之间的关系为依据，通过一顿猛如虎的数学操作，选举出一个新的 master。逻辑上，选举没有任何问题，如果 master 因为硬件故障而失效，新的 master 会自动顶替上，并在短时间内恢复工作。

而自然界总是狠狠地打人类的脸：

1. 硬件故障概率极低，大部分 master 失效都不是因为硬件故障

2. 如果是流量过大导致的 master 失效，那么选举出新的 master 也无济于事：提升集群规模才是解决之道
3. 即使能够及时地在一分钟之内顶替上 master 的工作，那这一分钟的异常也可能导致雪崩式的 cache miss，从磁盘缓存到虚拟内存，从 TLB 到三级缓存，再到二级缓存和一级缓存，全部失效。如果每一层的失效会让系统响应时间增加五倍的话，那最终的总响应时长将是惊人的。

系统规模问题

无论是 Master-Slave 模式还是 Proxy 模式，整个系统的流量最终还是要落到一个特定的资源上。当然这个资源可能是多台机器，但是依旧无法解决一个严重的问题：系统规模越大，其本底性能损失就越大。

这其实是我们所在的这个宇宙空间的一个基本规律。我一直认为，这个宇宙里只有一个自然规律：熵增。既然我们这个宇宙是一个熵增宇宙，那么这个问题就无法解决。

超级计算机

超级计算机可以看成一个规模特别巨大的分布式计算系统，他的性能瓶颈从目前的眼光来看，是超多计算核心（数百万）的调节效率问题。其本质是通信速率不够快，信息传递的太慢，让数百万核心一起工作，传递命令和数据的工作占据了绝大多数的运行时间。

神经网络

深度学习这几年大火，其原因就是卷积神经网络（CNN）造就的 AlphaGo 打败了人类，计算机在这个无法穷举的游戏里彻底赢了。伴随着 Google 帝国的强大推力，深度学习，机器学习，乃至人工智能，这几个词在过去的两年大火，特别是在中美两国。现在拿手机拍张照背后都有机器学习你敢信?

机器学习的瓶颈，本质也是数据交换：机器学习需要极多的计算，而计算速度的瓶颈现在就在运算器和存储器的通信上，这也是显卡搞深度学习比 CPU 快数十倍的原因：显存和 GPU 信息交换的速度极快。

九九归一

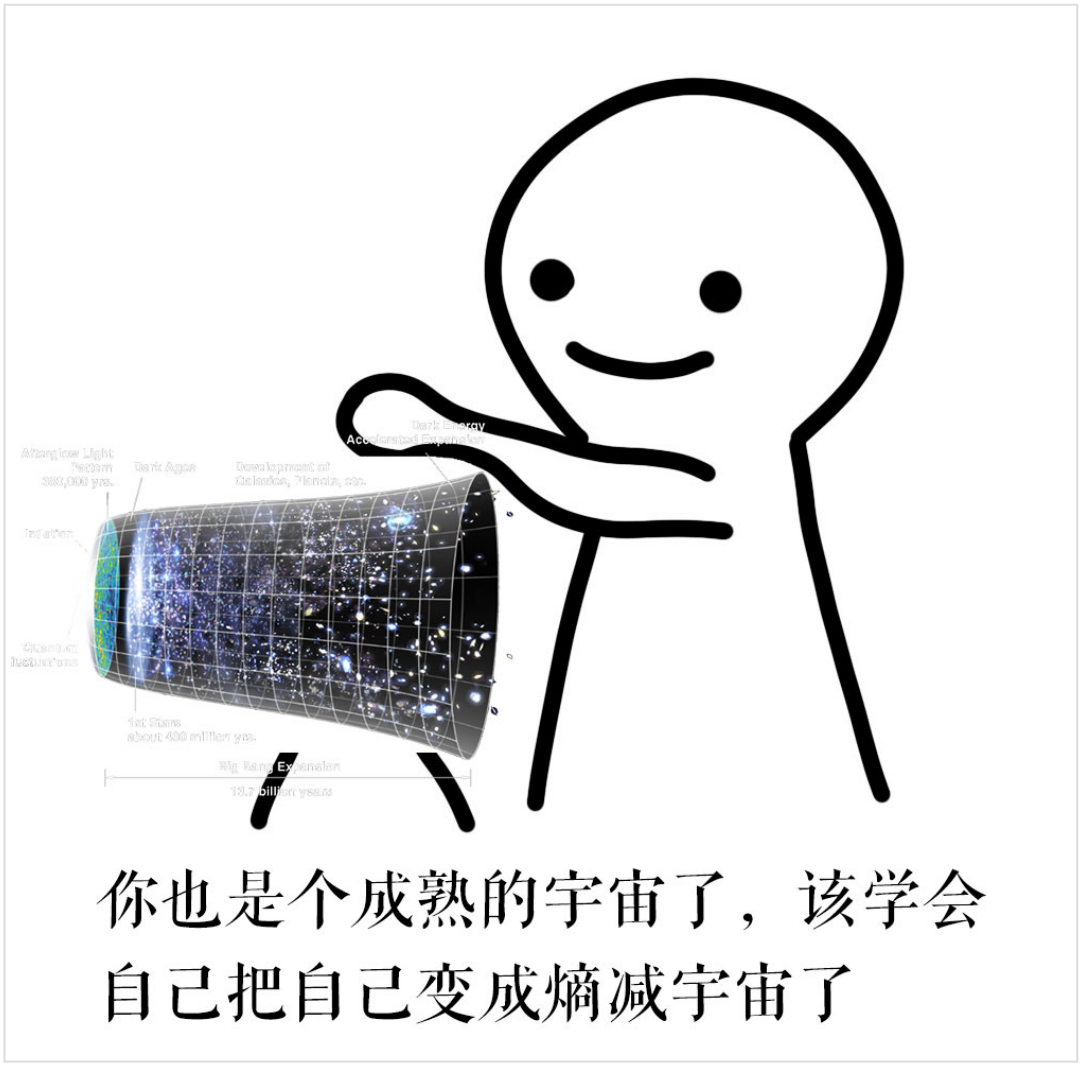
分布式系统的性能问题，表现为多个方面，但是归根到底，其原因只是一个非常单纯的矛盾：人们日益增长的性能需求和数据一致性之间的矛盾。一旦需要强数据一致性，那就必然存在一个限制性能的瓶颈，这个瓶颈就是信息传递的速度。

同样，超级计算机和神经网络的瓶颈也都是信息传递的速度。

那么，信息传递速度的瓶颈在哪里呢?

我个人认为，信息传递的瓶颈最表层是人类的硬件制造水平决定的，再往底层去是冯·诺依曼架构决定的，再往底层去是图灵机的逻辑模型决定的。可是图灵机是计算机可行的理论基础呀，所以，还是怪这个熵增宇宙吧，为什么规模越大维护成本越高呢，你也是个成熟的宇宙了，该学会自己把自己变成熵减宇宙了。

【全系列完结】



WRITTEN BY

JohnLui (https://github.com/johnlui)

程序员，Swift Contributor

相关日志：

性能之殇（六）-- 现代计算机最亲密的伙伴：局部性与乐观 (https://lvwenhan.com/操作系统/497.html)

性能之殇（四）-- Unix 进程模型的局限 (https://lvwenhan.com/操作系统/495.html)

软件工程师需要了解的网络知识：从铜线到HTTP（五）—— HTTP 和 HTTPS (https://lvwenhan.com/操作系统/489.html)

软件工程师需要了解的网络知识：从铜线到HTTP（四）—— TCP 和路由器 (https://lvwenhan.com/操作系统/488.html)

软件工程师需要了解的网络知识：从铜线到HTTP（三）—— TCP/IP (https://lvwenhan.com/操作系统/487.html)

标签: 性能之殇 (https://lvwenhan.com/tag/%E6%80%A7%E8%83%BD%E4%B9%8B%E6%AE%87) 性能 (https://lvwenhan.com/tag/%E6%80%A7%E8%83%BD) 分布式 (https://lvwenhan.com/tag/%E5%88%86%E5%B8%83%E5%BC%8F) 神经网络 (https://lvwenhan.com/tag/%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C) 超级计算机 (https://lvwenhan.com/tag/%E8%B6%85%E7%BA%A7%E8%AE%A1%E7%AE%97%E6%9C%BA)

性能之殇（六）-- 现代计算机最亲密的伙伴：局部性与乐观 → (https://lvwenhan.com/操作系统/497.html)

评论：

DHclly

2020-08-18 12:36

技术和文本功底都挺好的，硬核大佬

回复

名字君

2019-09-19 17:54

一口气看完，写的真是好，通俗易懂又能探寻问题本质，作为半个外行来都看得懂一些，收藏了以后慢慢理解

回复

游客

2018-12-20 15:16

没接触过量子通信，不敢多说，但我觉得现在的信息技术基于数字01的逻辑排列组合是一种非常低效的信息处理机制，虽然处理速度已经非常惊人。下一代信息技术不知是何，会不会是相对于数字的模拟，连续无极限。

回复

lly_0620

2018-12-17 15:11

一口气读完，爽歪歪

回复

文东

2018-12-04 19:26

非常赞的文章,楼主请继续多贡献·~

回复

Roidder

2018-11-29 12:00

精品文章
感谢作者大大
在功名浮躁的中国开发社区还可以潜心沉淀写这么深刻的底层东西很难得。

回复

StupidMan

2018-11-26 18:18

写的真好🤑🤑

回复

杜佳豪

2018-11-26 09:40

您好，我是机器之心运营，杜佳豪。希望能够转载这篇文章《性能之殇（七）-- 分布式计算、超级计算机与神经网络共同的瓶颈》到机器之心官网（jiqizhixin.com），希望能获得授权，我们会在文章开头着明来源和作者，并在文末着明原文链接，我的VX：xiaodenghuakai，希望能与您建立联系。

回复

JohnLui (https://lvwenhan.com/)

2018-11-26 10:11

@杜佳豪：没问题~

回复

发表评论：

昵称

邮件地址 (选填)

个人主页 (选填)

发表评论