# 论文阅读摘要

## MapReduce: Simplified Data Processing on Large Clusters

MapReduce是一种编程模型，用于大规模数据集的并行运算。用户指定一个map函数处理一个key/value对,从而产生中间的key/value对集.然后再指定一个reduce函数合并所有的具有相同中间key的中间value。

MapReduce通过把对数据集的大规模操作分发给网络上的每个节点实现可靠性；每个节点会周期性的返回它所完成的工作和最新的状态。如果一个节点保持沉默超过一个预设的时间间隔，主节点（类同Google File System中的主服务器）记录下这个节点状态为死亡，并把分配给这个节点的数据发到别的节点。每个操作使用命名文件的原子操作以确保不会发生并行线程间的冲突；当文件被改名的时候，系统可能会把他们复制到任务名以外的另一个名字上去。（避免副作用）。

化简操作工作方式与之类似，但是由于化简操作的可并行性相对较差，主节点会尽量把化简操作只分配在一个节点上，或者离需要操作的数据尽可能近的节点上；这个特性可以满足Google的需求，因为他们有足够的带宽，他们的内部网络没有那么多的机器。

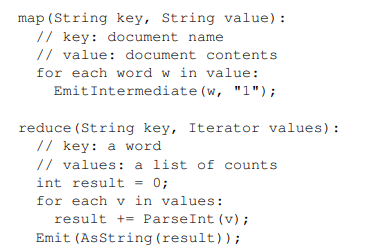
MapReduce提供了以下的主要功能：

1）数据划分和计算任务调度

2）数据/代码互定位

3）系统优化

4）出错检测和恢复



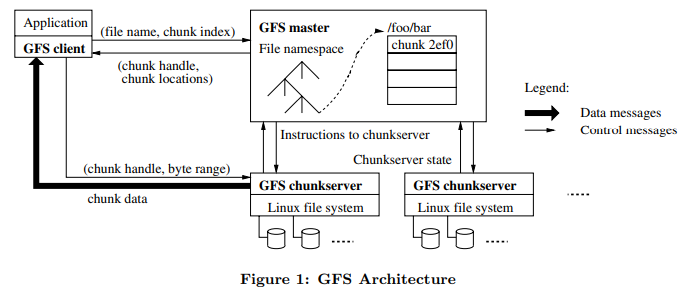
map函数emit每个word加上相关的出现计数（此处为1）。reduce函数将特定单词发出的所有计数汇总在一起。此外，用户可以编写代码，使用输入文件和输出文件的名称以及可选的调整参数，来填充mapreduce的规范对象（mapreduce specification object）。然后，用户调用MapReduce函数，将其传递给规范对象。用户的代码与MapReduce库（用C ++实现）链接在一起。

MapReduce优点：提供了抽象接口，分布式过程完全隐藏，使得程序员容易使用；使得大规模处理数据变得可能；自动的负载均衡；自动的容错性；

MapReduce缺点：极其严格的数据流；很多常见的操作也必须手写代码；程序内部实现隐藏，优化比较困难。

## The Google File System

谷歌文件系统是一个可扩展的分布式文件系统，用于大型的、分布式的、对大量数据进行访问的应用。它运行于廉价的普通硬件上，并提供容错功能。它可以给大量的用户提供总体性能较高的服务。



在谷歌文件系统中文件被切割为固定大小的Chunk然后分散存储，典型的Chunk大小为64MB。Chunk也是数据复制的基本单位，每个Chunk默认复制三份存放在不同的地方（机架、机器、磁盘）。

元数据节点：也称为Master。存储系统元数据信息，主要包括namespace、文件chunk信息以及chunk多副本位置信息。Master是系统的中心节点，所有客户端的元数据访问，如列举目录下文件，获取文件属性等操作都是直接访问Master。除此之外，还承担了系统诸多的管理工作，我们会在后面详细说明。

数据节点：文件chunk的存储位置。每个数据节点挂载多个磁盘设备并将其格式化为本地文件系统（如XFS）。将客户端写入数据以Chunk为单位存储，存储形式为本地文件。

客户端：提供类POSIX文件接口，应用程序使用客户端与GFS交互。

## FastSGG: Efficient Social Graph Generation Using a Degree Distribution Generation Model

随着社交媒体的快速发展，数十亿个节点在实际应用中变得越来越普遍。真实世界图和合成图对于评估社交网络分析任务的各种算法的性能和可扩展性都是必要的。但是，获取真实世界网络图有三个主要挑战，特别是对于大图。真实世界图和合成图对于评估上述SNA任务的各种算法的性能和可扩展性都是必要的。但是，获取真实世界网络图有三个主要挑战，特别是对于大图。首先，通常需要花费大量时间来分析和处理。其次，很难实现理想的大真实网络。第三，大型复杂网络的特征在爬行的真实网络*（？）*中经常被破坏。

关于社交图谱生成的架构有以下四个定义：

顶点架构：*顶点架构 VS* = （*lbl，amount，attr*），*其中 amount 是标记为"lbl"的顶点的数量，attr 是属性信息。*

边架构：*边架构 ES* = (*lbl, lbls, lblt, amount, distrin, distrout, attr*)

社区架构：*社区架构 CS* = (*lble,amount, λs,λt,ρ*)

社交图生成架构：*社交图生成架构 SGS* = （*VSS，ESS，CSS），其中 VSS、ESS 和 CSS 分别表示一组顶点架构、一组边缘架构和一组社区架构。*

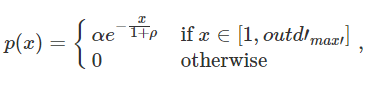
生成社交图的算法

Table 2- 
Social Graph Generation

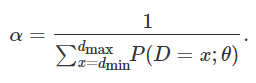

度分布生成模型

1. The D2G model

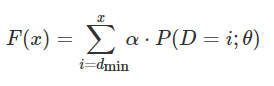
使用概率密度函数在x度邻域上的积分来代表度数x的存在概率



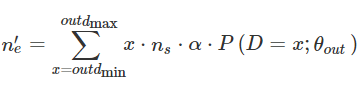
其中α的计算公式



累积分布函数



1. Out-degree Generation



1. Edge Generation

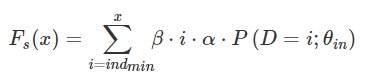
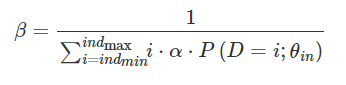
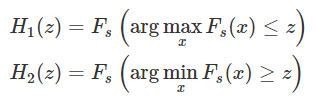
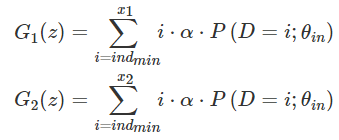


Table 5- 
DetermineTargetVertex





性能分析：

建立G（z）和 F（x）的时间和空间复杂度均为 O(*nv*).

建立H的时间和空间复杂性*H*1(*z*)*, H*2(*z*)*, G*1(*z*)*, and G*2(*z*)都是O(*knv*).

生成一般图形的时间复杂度为*O*(*ne*)，而它的空间复杂度是*O*(*knv*).

生成社交图谱的时间复杂度为*O*(*ne*)，而它的空间复杂度是*O*(*knv*).