5月15日

web

全文搜索

Problem1: 文件格式异构

2:索引效率

3:weight

关系型数据库 sql within

Lucene

复杂度从O（I\*n）i为文章数 n为长度

降到了O(N+m+n)

中文分词？ 中科院

同义词搜索

reader.close()删除写入硬盘 光delete 其他线程访问依然没删掉

删除是懒惰的 加标签 因此可以undelete

文件格式：jpg png等等

按颜色搜索 抽取位图的颜色信息 大于阈值

人 风景 动物 神经网络训练 深度学习-》中间节点层数多

google tensorflaw

5月17日 web

网格计算不可行 不愿意贡献

资源分配可伸缩

解决系统需求变化：建立虚拟机

google3篇论文

mapreduce：任务调度

分布式文件系统管理文件 big-table

数据存储扩展 不同租户

1 数据库隔离

2 表隔离

3 表外键扩展

前端定制：

ajax

（1）Ssd 当作缓存

Hdd 当作硬盘

问题

1 浪费 ssd空间

2 cache不同步

Ssd在文件小且访问随机时 性能高

HDD 文件大 访问连续

（2）混合存储：

一块放在ssd 另一块 hdd

动态分配

主控节点 控制 ssd和hdd的动态分配

1 备份

2 分布开

web 5月24日

mapreduce前提

批处理

数据线性可切 数据库外键关联不行

map-map最后一次map-reduce combine

shuffle 整理数据

Worker闲下来 做其他worker没有完成的数据

？能否接着其他worker完成一半的数据继续做

中间结果-》.out 分片写出

Out先不要合并 记录处理游标 中间结果在reduce那里

Worker间的负载不同

任务-》FIFO（默认） deadline 任务排队应是adaptive

Bigtable

记录不同的时间戳上的值

Familiy里的字段可以灵活添加

事务处理麻烦 因为分布式事务需要两阶段提交

Monogodb不适合存储事务型数据

Master写完了后立刻返回 向slave同步 同步记录游标 可以读slave已经同步好的数据

CAP 分区一致性

Mysql事务提交前放在缓存里

No-sql提交前写入硬盘

事务时间长因为副本多

最终一致性

当数据冗余时 spark会自动清理内存 .cache()不会被清理

RDD 键值对

数据划分粒度太大

工作流有向无环图

性能高原因：

1 spark比java执行效率高

2 内存比硬盘效率高

Iaas 虚拟机调度

Project：

日志文件

尝试登陆总是失败

用户使用网站的习惯：白天／晚上

**web 6月7**

Region靠行锁来分裂

大了分裂

副本 不一定只在一个server上有

memstore写不下了导出storefile

compaction 合并key应该是靠的很近

大文件不停的往硬盘上写

Hdfs可以用对象存储

Block过大拿出来效率低

client屏蔽namenode和datanode的操作 namenode是瓶颈

standby namenode上只跑应用，元数据共享

zookeeper监控namenode的健康状况

HDFS读

如果文件有备份

选取离他最近的datanode 靠距离做负载均衡

Datanode读的时候可以不加锁

写的时候需要加锁

HIVE底层是HBASE

MAPREDUCE2.0 任务与资源的管理分开 YARN

HDFS底层的文件系统可以更换 比如换成对象存储的文件系统，用散列来管理元数据，效率更高

在对象存储中，存储的不仅是数据，还有与丰富的数据相关的属性信息。系统会给每一个对象分配一个唯一的ID。对象本身是平等的，所有的ID都属于一个平坦的地址空间，而并非文件系统那样的树状逻辑结构。使用hash为新的对象分配id。

一致性Hash法

Block的size不能太大，如果只读一个小文件，需要把整个block都load进来，I/O的效率低

太小的话大文件寻址负担重，并且namenode种需要存大量的元数据。