Basho levelDB
实践

360-陈宗志

- 背景介绍
- 原理
- 改进点
- 新的挑战

背景

- 360 的KV服务是基于Dynamo实现
- 底层的存储引擎使用的是LevelDB

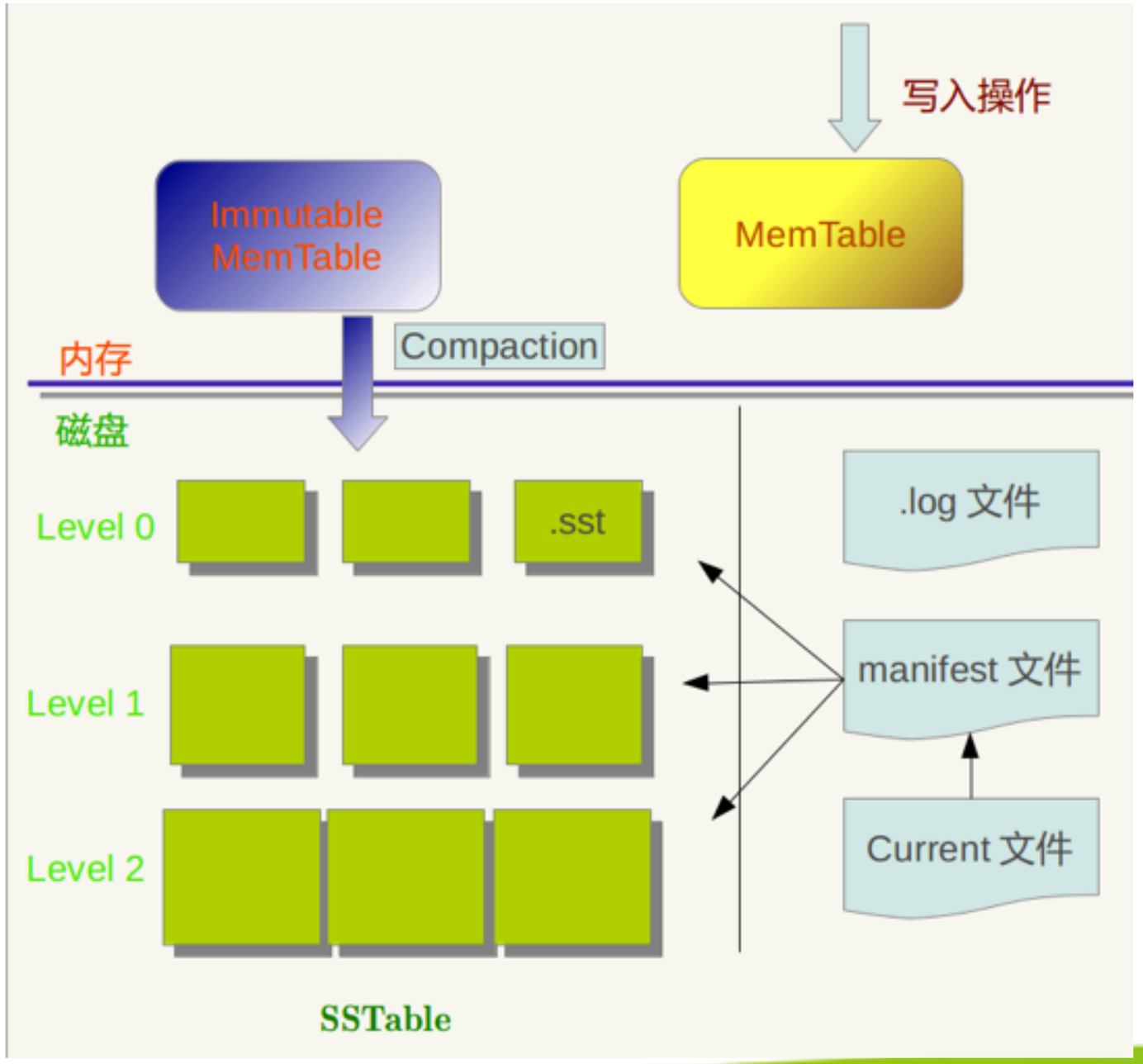
背景

- LevelDB is a fast key-value storage library written at Google that provides an ordered mapping from string keys to string values.
- LevelDB 单机可以支持百万级别的数据量

原理

- 整体是一个LSM Tree的实现
- 插入: 先写内存, 然后将内存中的数据Dump 成一个静态文件
- 读取: 先读内存, 然后一次读取静态文件
- 整体通过Compaction 将文件从内存到磁盘

原理


写入操作
Immutable
MemTable
MemTable
Compaction
内存
磁盘
Level 0
.sst
Level 1
Level 2
.log 文件
manifest 文件
Current 文件
SSTable

改进点1

- 官方: 只限制sst文件的大小
- Basho: 限制sst文件的大小同时限制sst文件key的个数<75000
- 原因: 为了控制bloom filter中key的个数, key过多bloom filter的误判率增高

改进点2

- 官方: 当Compaction线程落后很多的时候, 会不可写
- Basho: 增加Compaction线程, 每个线程有优先级. 优先级最高的是imm_ 到 Level 0的Compaction
- 原因: 因为当imm_满的时候, 写入是不允许的. 增加Compaction的优先级, 可以优先满足imm_到Level 0 的Compaction

改进点3

- 官方: 没有统计当前DB的key个数等方法

- Basho: 增加统计工具. 通过在sst文件的头部添加统计结构, 可以统计每一个sst文件中key 的个数

- 原因: 方便管理统计

改进点4

- 官方: 没有DB的操作数的记录统计
- Basho: 在Leveldb进程加入shared memory segment, 用来统计Get, Put, OpenFile 等当前信息
- 原因: 方便实时观察DB的运行情况

改进点5

- 官方: 所有Level文件放在同一目录
- Basho: 将相应级别的文件放入的相应目录
- 原因: 在Linux下面一个目录下面文件过多对文件的打开速率有略微的影响. 方便对不同级别统计

改进点6

- 官方: 每个级别的sst文件大小2M
- Basho: 定制了每个级别的sst文件大小
- 原因: 因为一个进程需要打开64个levelDB实例, 所以需要限制levelDB单个实例的open_files.

改进点7

- 官方: 不支持key的expire time 设置
- 360: 支持expire time设置. 设置了expire time的 key 会在超时后, 自动在Compaction的时候被删除
- 原因: 业务方有key过期时间的要求

新的挑战

- 加入动态分库功能, 目前分库需要手动迁移并导入数据
- 支持multi-get 和 multi-put 功能

Q & A

Thanks

我们正在招聘, 欢迎加入我们

chenzongzhi@360.cn