LevelDB 介绍

陈宗志

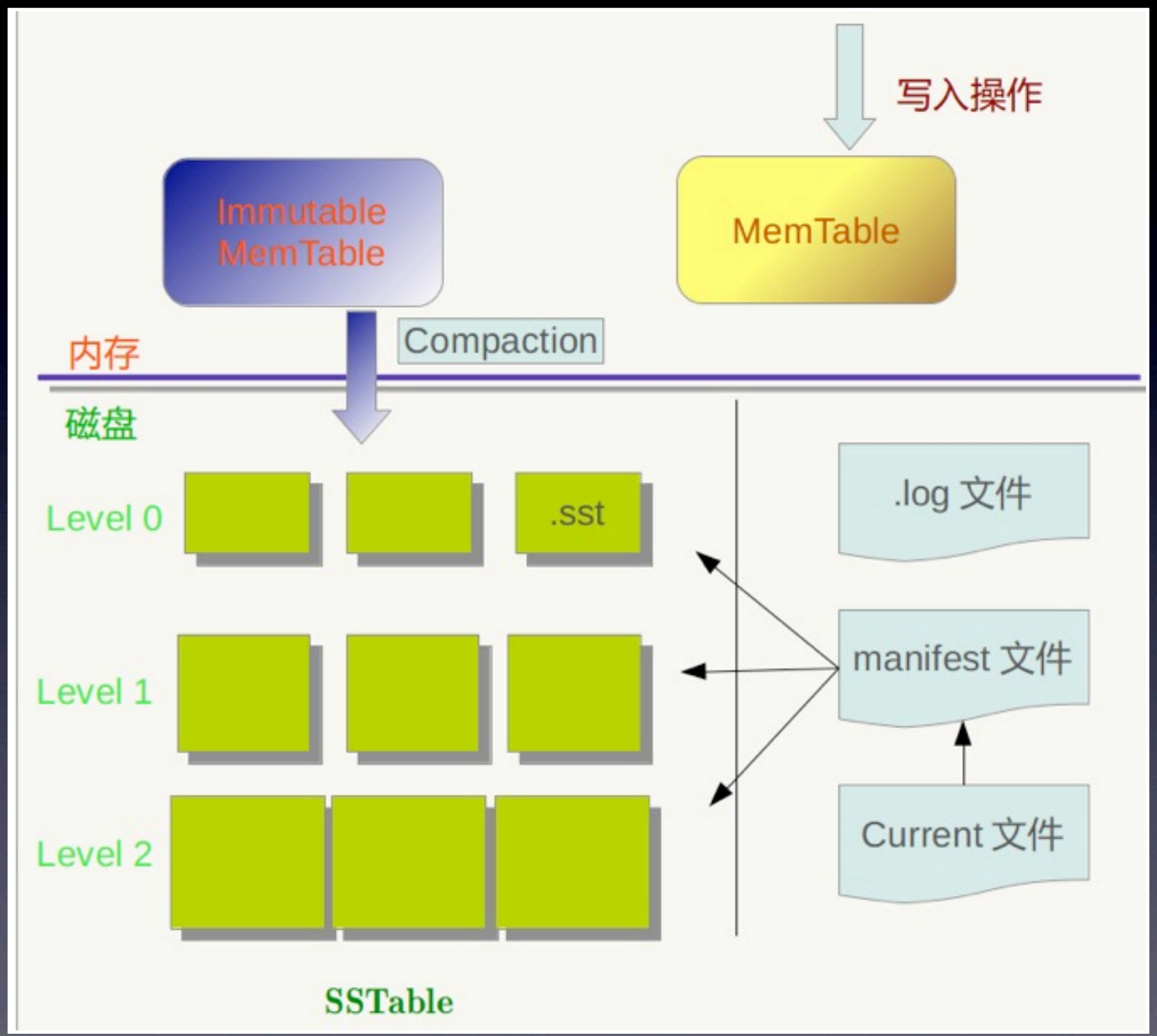
简介

- 背景介绍
- 结构介绍
- 读写过程
- 可改进点

背景介绍

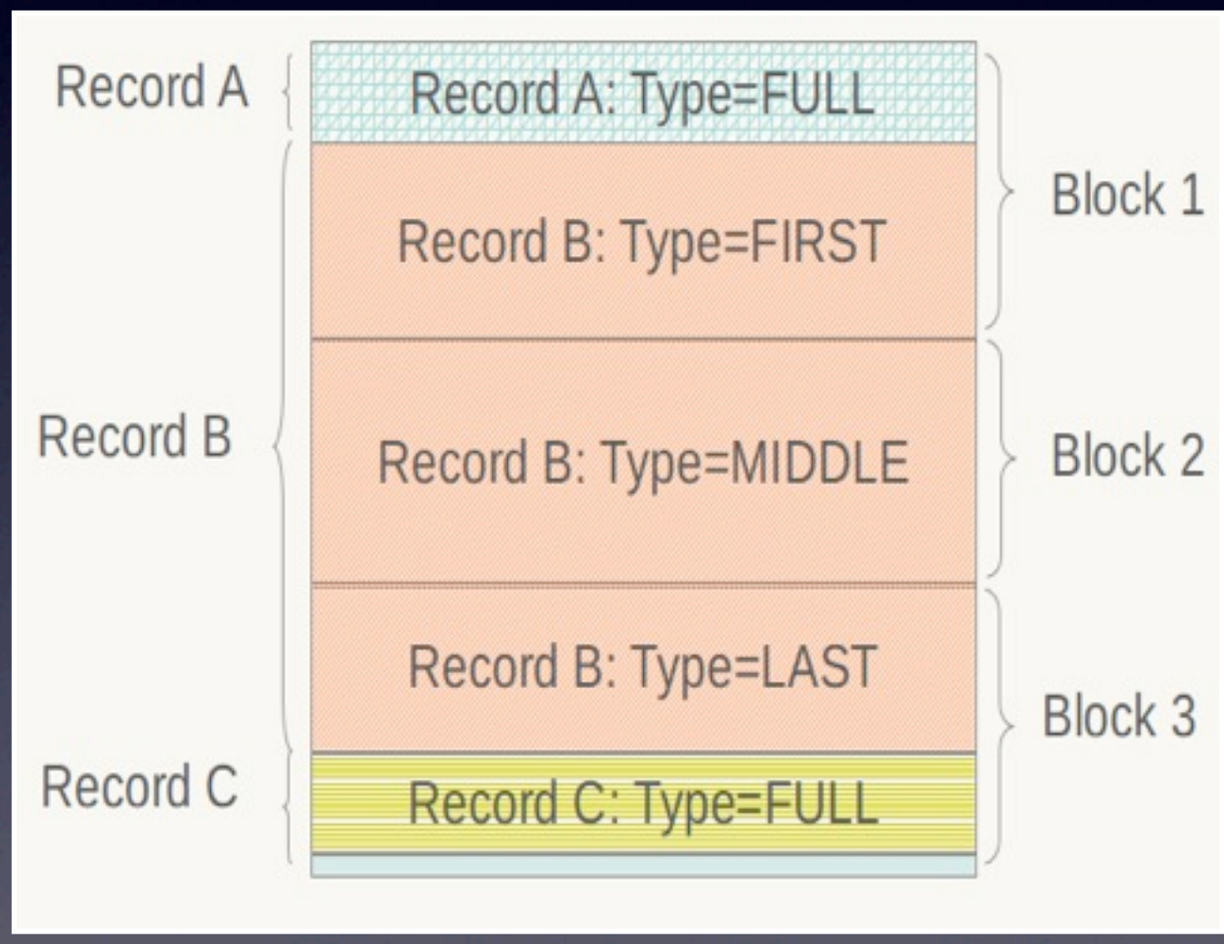
- Leveldb是google开源的一个单机K/V存储系统。从项目的contributor(sanjay,jeff beans)来看，很多人认为它是bigtable在单个节点上的实现,即类似于tablet

整体结构


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

log file

- Log file: 32KB, 以Block为单位



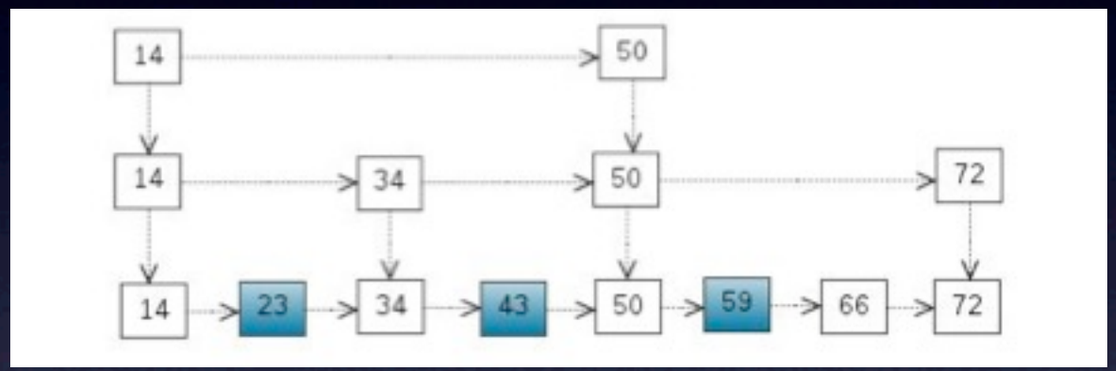
Sst file

- 一个Block的大小是4k
- 一个sst文件的大小可以配置.默认都是2M

Block 1	Type	CRC
Block 2	Type	CRC
Block 3	Type	CRC
Block 4	Type	CRC
Block 5	Type	CRC
Block 6	Type	CRC
Block 7	Type	CRC
Block 8	Type	CRC

Memtable

- 一个Skiplist的实现



- Immutable 与 Memtable 是一样的

Manifest

- Manifest文件列出了每个level的排序表，对应的key范围和其他重要的metadata. 每次db被重新打开的时候，生成一份新的MANIFEST. manifest的格式类似log

CURRENT

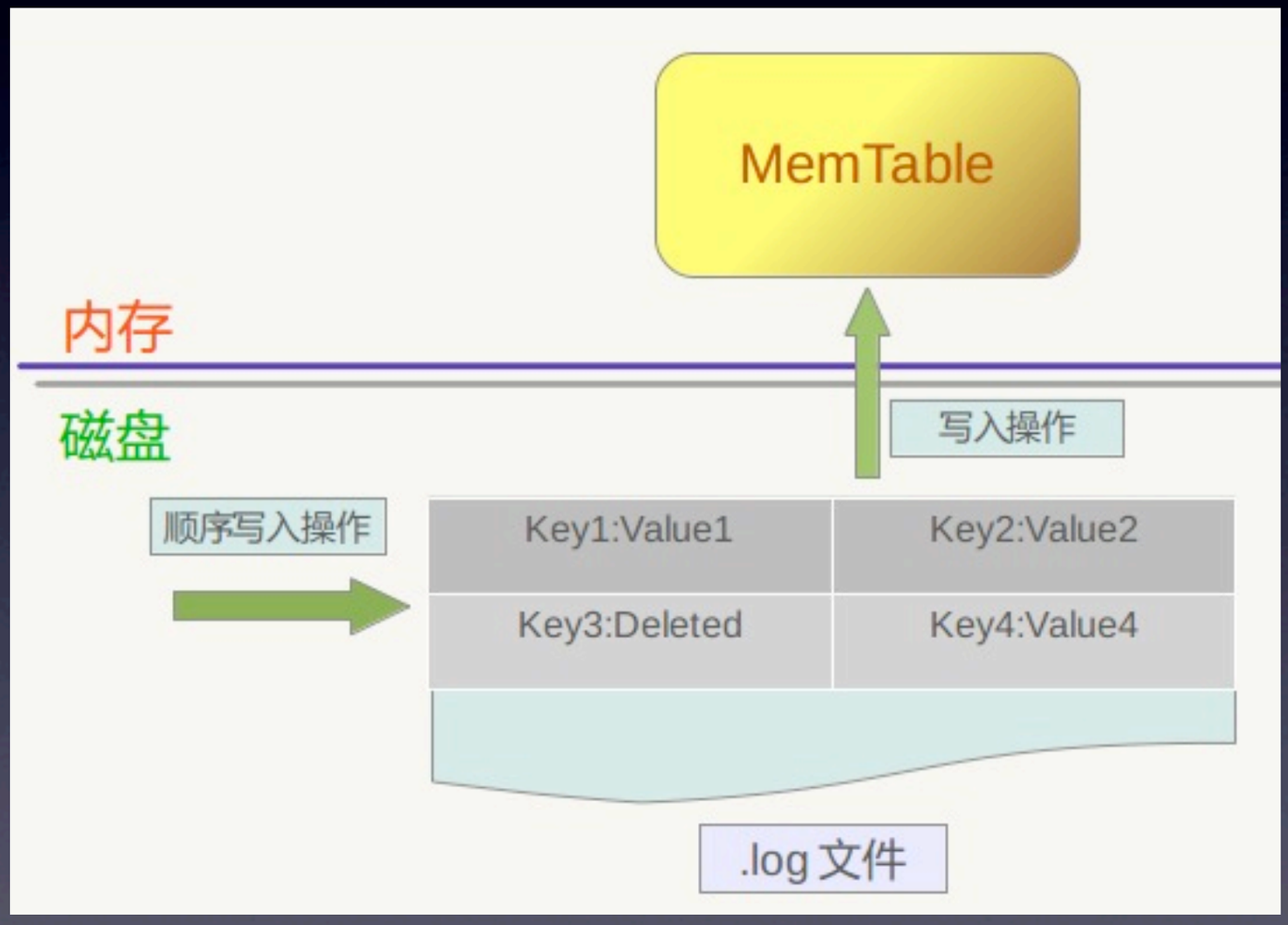
- 列出了最新的manifest名字

LOCK

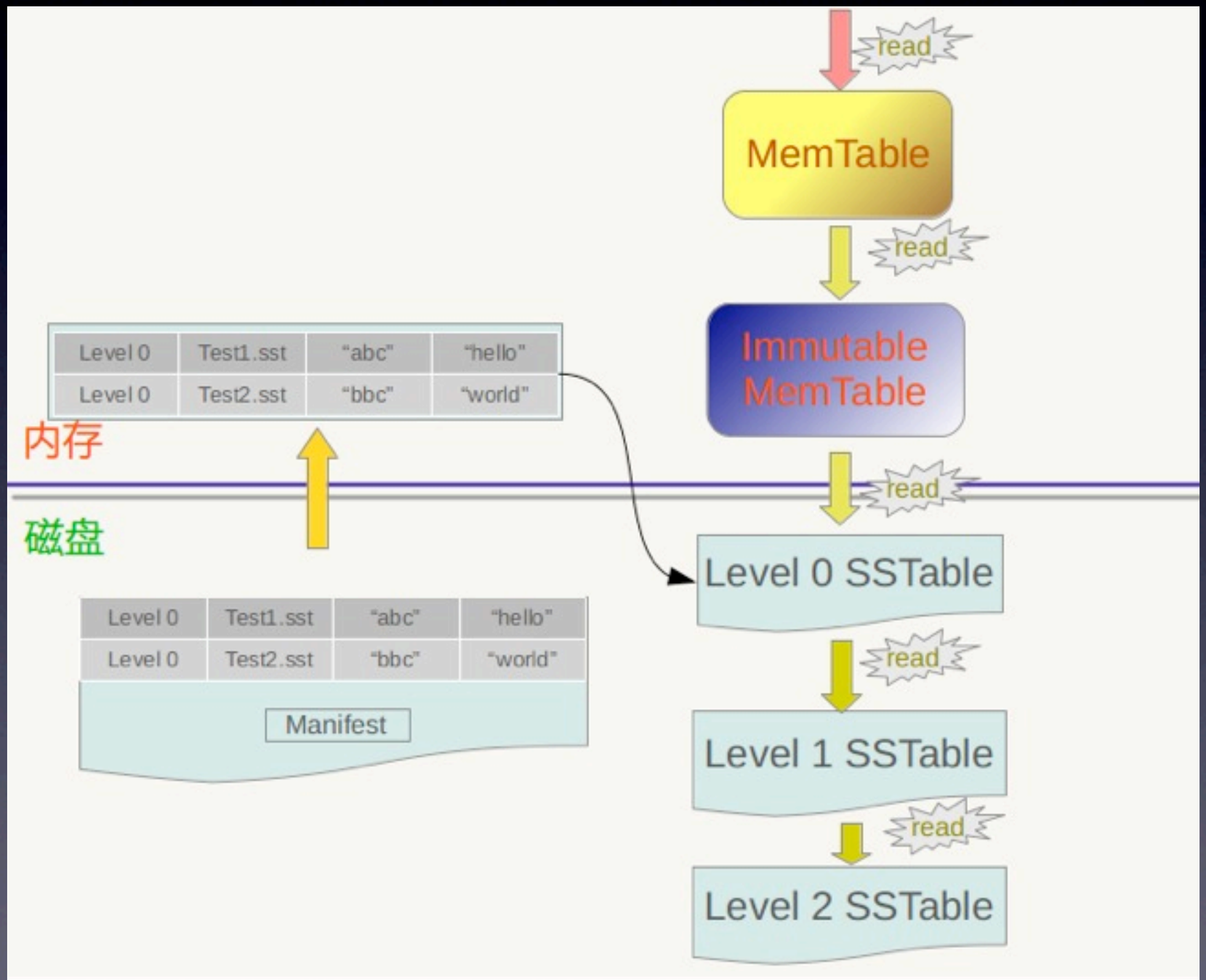
- 保证一个进程只有一个levelDB实例的文件
- 通过fcntl判断这个文件是否打开来实现

读写过程

Write


MemTable
内存
磁盘
写入操作
顺序写入操作
Key1:Value1
Key2:Value2
Key3:Deleted
Key4:Value4
.log 文件

Read


read
MemTable
read
Immutable
MemTable
read
Level 0 SSTable
read
Level 1 SSTable
read
Level 2 SSTable
Level 0 Test1.sst "abc" "hello"
Level 0 Test2.sst "bbc" "world"
内存
磁盘
Level 0 Test1.sst "abc" "hello"
Level 0 Test2.sst "bbc" "world"
Manifest

Compaction

- Compaction机制是leveldb的核心, leveldb之所以叫”leveldb” 就是因为compaction带来的层级结构

Compaction

- Level 0: 无序的多路归并
- 其他Level: 有序的多路归并
- 策略: 轮流合并, 限制级别大小

可改进点

- 可以提供自己的比较函数来重载原有的比较函数
- 封装了底层的文件系统, 可以用任意的文件系统替换
- 一些参数的配置，级别文件大小, 应该根据SSD盘进行优化

Thank You