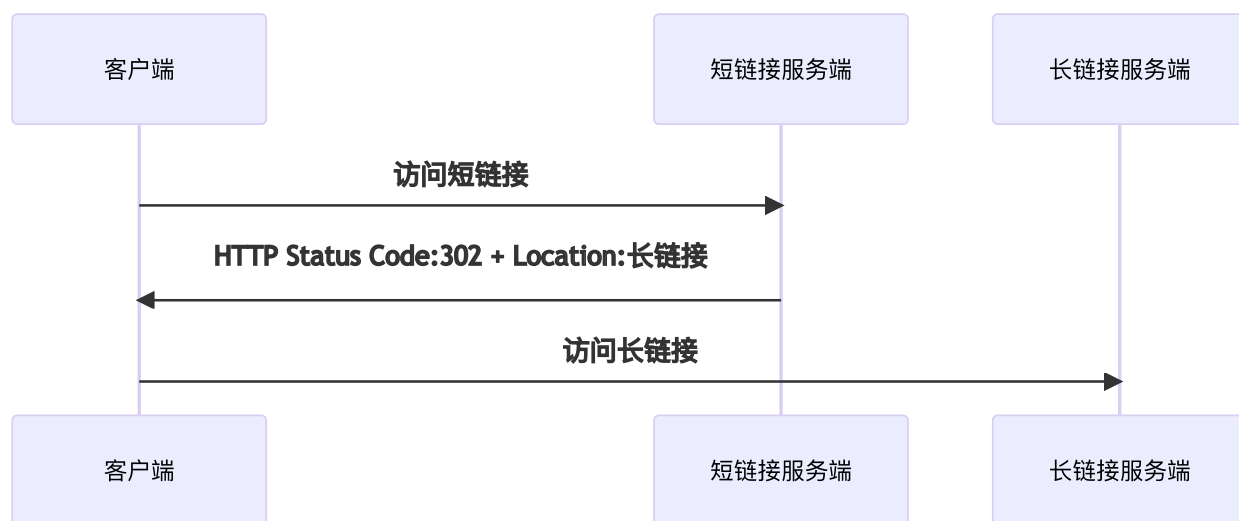


背景

实现一个短域名服务，要求：

- 短域名存储接口：接受长域名信息，返回短域名信息
- 短域名读取接口：接受短域名信息，返回长域名信息
- 短域名长度最大为 8 个字符
- 采用SpringBoot，集成Swagger API文档
- JUnit编写单元测试，使用Jacoco生成测试报告(测试报告提交截图)
- 映射数据存储在JVM内存即可，防止内存溢出(因此本方案只讨论单机服务架构，不涉及集群)

短链接的原理



如图，访问短链接后服务会重定向到长链接。考虑到短链接与长链接的对应关系可能会变，同时方便服务端统计，返回长链接时采用临时重定向302。

短链接的生成

任意长度的长链接得到限定长度的短链接，首先想到的就是Hash算法：

- MD5：128bit
- SHA-2：256bit
- Murmur3：32bit和128bit
- SipHash-2-4：64bit

1. 相较于MD5和SHA等加密Hash算法，MurmurHash和SipHash等非加密Hash的运算速度快n倍，并且SipHash能有效减缓hash flooding攻击。
2. 若采用SipHash-2-4算法，最大值为2的64次方：18446744073709551616，十进制20位，可转换为62进制，得到IYGhA16ahyg，大于需求中要求的8个字符。
3. 若采用Murmur3_32算法，最大值为2的32次方：4294967296，十进制10位，转换为62进制，得到4GFfc4，符合需求短链接长度需求。即使MurmurHash的冲突概率再低，也不可避免Hash冲突，此时只能人为的给长链接加一些规则，重新计算Hash，必要时可循环处理，直到没有冲突为止。
4. 每次生成短链接时都需要判断短链是否存在冲突，当数据量很大时会造成读压力，因此可以采用BloomFilter优化，以空间换时间。

短链接的存储

由于只能采用内存存储，并且数据是kv结构，可以选择本地缓存方案：

缓存	Encache	Guava Cache	Caffeine
读写性能	一般	一般	RingBuffer,很好
淘汰策略	LRU,LFU,FIFO	LRU,一般	W-TinyLFU,很好
是否持久化	是	否	否
是否支持集群	是	否	否

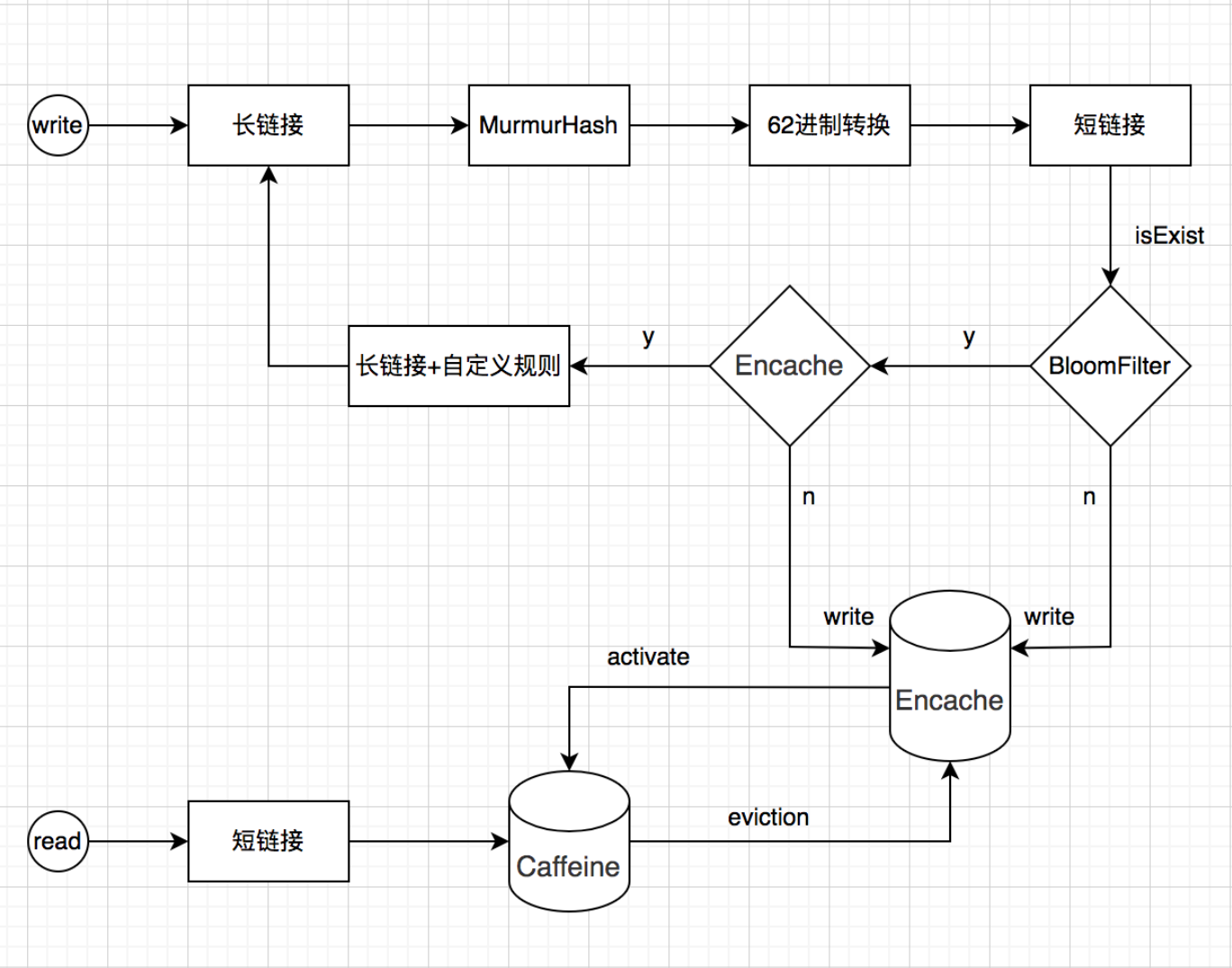
- 1. 由于Caffeine采用了最先进的淘汰算法W-TinyLFU，融合了LRU和LFU，性能无疑是最好的。
- 2. 但是当存储数据量很大时，为防止内存溢出，还是需要将kv数据持久化到磁盘文件。
- 3. 那么可以将Caffeine作为一级缓存，存储热点数据，再选择可持久化的Encache作为二级缓存，存储全量数据。

短链接的访问

当访问短链接时，先读取一级缓存，若没命中，再读取二级缓存，并将数据load到一级缓存中。

架构设计图

综上，整体的系统架构图如下：




Jacoco测试报告

Current scope: all classes

Overall Coverage Summary

Package	Class, %	Method, %	Line, %
all classes	100% (17/17)	91.4% (64/70)	86.9% (172/198)

Coverage Breakdown

Package 	Class, %	Method, %	Line, %
com.eagle.shorturl.util	100% (4/4)	66.7% (8/12)	77.8% (21/27)
com.eagle.shorturl.service.impl	100% (3/3)	100% (17/17)	80.3% (61/76)
com.eagle.shorturl.result	100% (2/2)	100% (11/11)	100% (21/21)
com.eagle.shorturl.param	100% (1/1)	100% (4/4)	100% (4/4)
com.eagle.shorturl.interceptor	100% (1/1)	100% (3/3)	90.9% (10/11)
com.eagle.shorturl.exception	100% (2/2)	90.9% (10/11)	92.6% (25/27)
com.eagle.shorturl.controller	100% (1/1)	100% (4/4)	100% (4/4)
com.eagle.shorturl.config	100% (2/2)	100% (6/6)	100% (25/25)
com.eagle.shorturl	100% (1/1)	50% (1/2)	33.3% (1/3)