

# 高速缓存系统的设计思路 -JAVA 版

作者: [yhm2](#)

原文链接: <https://ld246.com/article/1577782199548>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

高速缓存??

啥意思?

缓存本来就很快了, 还要高速?

对的!!!

下列代码实现了读和写的分离, 第一次读和写的混合, 其他读都需要等待的功能。细细斟酌

```
package com.yhm.testSpringMvc.readWriteLock;

import java.util.HashMap;
import java.util.Map;
import java.util.concurrent.locks.ReadWriteLock;
import java.util.concurrent.locks.ReentrantReadWriteLock;

public class ReadWriteLockCache {
    public static void main(String[] args) {
    }

    private Map<String, Object> cache = new HashMap<>();
    private Object returnObj = null;
    private ReadWriteLock readWriteLock = new ReentrantReadWriteLock();

    public synchronized Object getCacheBySynchronized(String key) {
        Object obj = null;
        if (returnObj == null) {
            returnObj = "查数据库";
        }
        return obj;
    }

    public Object getCacheByReadWriteLock(String key) {

        readWriteLock.readLock().lock();
        try {
            returnObj = cache.get(key);
            if (returnObj == null) {
                readWriteLock.readLock().unlock();
                try {
                    readWriteLock.writeLock().lock();

                    if (returnObj == null) {
                        returnObj = "查数据库";//做真正LOAD到缓存的事情
                    }
                } catch (Exception e) {
                } finally {

                    readWriteLock.writeLock().unlock();
                }
                readWriteLock.readLock().lock();
            }
        } catch (Exception e) {
```

```
    } finally {  
        readWriteLock.readLock().unlock();  
    }  
    return returnObj;  
}  
}
```