Spring,hibernate整合ehcache使用：

实现方式：

注解方式，或者xml-aop方式；

xml-aop方式，

Spring配置文件，

<bean id=*"cacheManager"* class=*"org.springframework.cache.ehcache.EhCacheCacheManager"* p:cache-manager-ref=*"ehcache"*/>

<!-- EhCache library setup -->

<bean id="ehcache" class="org.springframework.cache.ehcache.EhCacheManagerFactoryBean" p:config-location="ehcache.xml">

<!-- 由于hibernate也使用了Ehcache, 保证双方都使用同一个缓存管理器 -->

<property name="shared" value="true"/>

</bean>

<!-- cache definitions -->

<cache:advice id=*"cacheAdvice"* cache-manager=*"cacheManager"*>

<cache:caching cache=*"findAllCache"*>

<cache:cacheable method=*"findAll"* />

<cache:cache-evict method=*"reload"* all-entries=*"true"*/>

</cache:caching>

</cache:advice>

<!-- apply the cacheable behavior to all BookService interfaces -->

<aop:config>

<aop:advisor advice-ref=*"cacheAdvice"* pointcut=*"execution(\* com.neirong.cms.service.\*.\*(..))"*/>

</aop:config>

Resources文件夹下建立*ehcache.xml文件，*

*Ehcache.xml*，

<ehcache>

<cache name=*"findAllCache"*

eternal=*"true"*

timeToIdleSeconds=*"900"*

timeToLiveSeconds=*"1800"*

maxElementsInMemory=*"2000"*

overflowToDisk=*"false"*

memoryStoreEvictionPolicy=*"LRU"* />

</ehcache>

name 缓存名称；

eternal="true" 缓存对象是否永久有效，false时，下面两条设置生效，true时，下面两条不生效；

timeToIdleSeconds="900" 闲置失效时间；0为无穷；

timeToLiveSeconds="1800" 存活时间；0为无穷；

maxElementsInMemory="1000" 内存缓存中最大存放数量，超过时：或者存放入磁盘缓存，或者替换已有

overflowToDisk="false" 内存不足时,是否启用磁盘缓存

memoryStoreEvictionPolicy=”LFU” 内存存储与释放策略,即达到maxElementsInMemory限制时,Ehcache会根据指定策略清理内存

共有三种策略,分别为LRU(最近最少使用，时间单位)、LFU(最常用的，清除点击最少的)、FIFO(先进先出)

~

maxElementsOnDisk 磁盘缓存中最多可以存放的元素数量,

**<diskStore** path="java.io.tmpdir"**/>** 磁盘缓存写入位置；

diskPersistent 是否持久化磁盘缓存

diskExpiryThreadIntervalSeconds 磁盘缓存的清理线程运行间隔,默认是120秒

diskSpoolBufferSizeMB 设置DiskStore（磁盘缓存）的缓存区大小,默认是30MB

数据操作类，

一切照旧，不需要特别标识cache；