# 二级缓存

## 什么是二级缓存

* hibernate 提供缓存机制：一级缓存、二级缓存

一级缓存：session级别缓存，在一次请求中共享数据。

二级缓存：sessionFactory级别缓存，整个应用程序共享一个会话工厂，共享一个二级缓存。

## 应用场景

* 适合放入二级缓存中的数据:

很少被修改

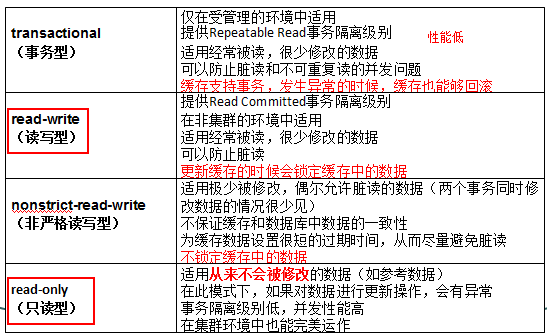
不是很重要的数据, 允许出现偶尔的并发问题

* 不适合放入二级缓存中的数据:

经常被修改

财务数据, 绝对不允许出现并发问题

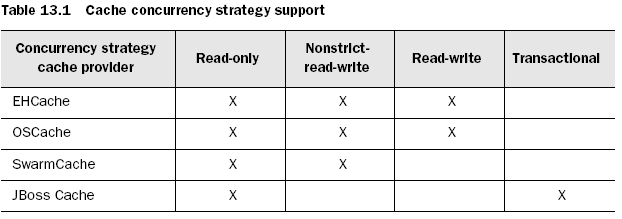
## 并发访问策略



* 访问策略：读写型（read-write）、只读型（read-only）

## 二级缓存提供商

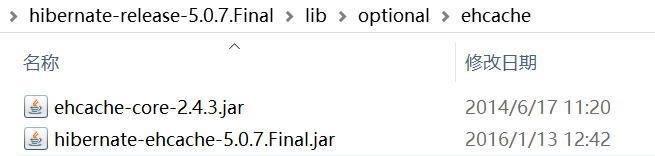
* **EHCache**: 可作为进程（单机）范围内的缓存, 存放数据的物理介质可以是内存或硬盘, 对 Hibernate 的查询缓存提供了支持。--支持集群。
* OpenSymphony `:可作为进程范围内的缓存, 存放数据的物理介质可以是内存或硬盘, 提供了丰富的缓存数据过期策略, 对 Hibernate 的查询缓存提供了支持
* SwarmCache: 可作为集群范围内的缓存, 但不支持 Hibernate 的查询缓存
* JBossCache:可作为集群范围内的缓存, 支持 Hibernate 的查询缓存



X表示支持

## 配置二级缓存（操作）

### 1. 导入jar包：



### 2.开启二级缓存(我要使用二级缓存)

|  |
| --- |
| <!--开启二级缓存-->  <property name=*"hibernate.cache.use\_second\_level\_cache"*>true</property> |

### 3.确定二级缓存提供商(我要使用哪个二级缓存)

|  |
| --- |
| <!--配置使用的二级缓存 -->  <property name=*"cache.region.factory\_class"*>  org.hibernate.cache.ehcache.EhCacheRegionFactory</property> |

### 4.确定需要缓存内容

#### 1>配置需要缓存的类

|  |
| --- |
| <!-- 配置二级缓存的类 -->  <class-cache usage=*"read-only"* class=*"pojo.User"*/>  <class-cache usage=*"read-only"* class=*"pojo.Order"*/> |

#### 2>配置需要缓存的集合

|  |
| --- |
| <!-- 配置二级缓存的集合 缓存集合要把该类先缓存了 不然不生效-->  <collection-cache usage=*"read-only"* collection=*"pojo.User.orders"* /> |

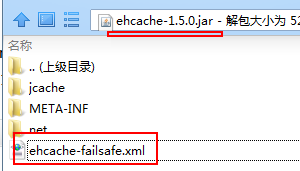
### 5.测试

|  |
| --- |
|  |



### 6.配置ehcache自定义配置文件

步骤1：从jar包复制xml文件



步骤2：将xml重命名“ehcache.xml”



步骤3：将修改后的xml，拷贝到src下

|  |
| --- |
| <ehcache xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xsi:noNamespaceSchemaLocation=*"../config/ehcache.xsd"*>  <diskStore path=*"D:/cache"*/>  <!-- 配置属性          timeToIdleSeconds 当缓存闲置n秒后销毁          timeToLiveSeconds 当缓存存活n秒后销毁          maxElementsInMemory：缓存最大个数。          eternal:对象是否永久有效，一但设置了，timeout将不起作用。          timeToIdleSeconds：设置对象在失效前的允许闲置时间（单位：秒）。仅当eternal=false对象不是永久有效时使用，可选属性，默认值是0，也就是可闲置时间无穷大。          timeToLiveSeconds：设置对象在失效前允许存活时间（单位：秒）。最大时间介于创建时间和失效时间之间。仅当eternal=false对象不是永久有效时使用，默认是0.，也就是对象存活时间无穷大。          overflowToDisk：当内存中对象数量达到maxElementsInMemory时，Ehcache将会对象写到磁盘中。          diskSpoolBufferSizeMB：这个参数设置DiskStore（磁盘缓存）的缓存区大小。默认是30MB。每个Cache都应该有自己的一个缓冲区。          maxElementsOnDisk：硬盘最大缓存个数。 diskPersistent：是否缓存虚拟机重启期数据 Whether the          disk store persists between restarts of the Virtual Machine. The          default value is false.          diskExpiryThreadIntervalSeconds：磁盘失效线程运行时间间隔，默认是120秒。          memoryStoreEvictionPolicy：当达到maxElementsInMemory限制时，Ehcache将会根据指定的策略去清理内存。默认策略是LRU（最近最少使用）。你可以设置为FIFO（先进先出）或是LFU（较少使用）。          clearOnFlush：内存数量最大时是否清除。          -->  <defaultCache  maxElementsInMemory=*"10000"*  eternal=*"false"*  timeToIdleSeconds=*"120"*  timeToLiveSeconds=*"120"*  overflowToDisk=*"true"*  maxElementsOnDisk=*"10000000"*  diskPersistent=*"false"*  diskExpiryThreadIntervalSeconds=*"120"*  memoryStoreEvictionPolicy=*"LRU"*    />  </ehcache> |

### 7.查询缓存

* 查询缓存默认不使用。需要手动开启
* 查询缓存：将HQL语句与 查询结果进行绑定。通过HQL相同语句可以缓存内容。

默认情况Query对象只将查询结果存放在一级和二级缓存，不从一级或二级缓存获取。

查询缓存就是让Query可以从二级缓存获得内容。

步骤一：开启查询缓存

|  |
| --- |
| <!-- 开启查询缓存 -->  <property name=*"hibernate.cache.use\_query\_cache"*>true</property> |

步骤二：在查询query对象，设置缓存内容（注意：存放和查询 都需要设置）

