

广告位招租

光荣在于平淡，卓越在于专著

首页 日志 相册 音乐 收藏 博友 关于我

日志

使用MyEclipse集成SSH和DWR(三)整合Spring和DWR - □

Hibernate缓存概述 - pansu的专栏 - CSDN 博客

Hibernate缓存机制 - Hibernate - Java - JavaEye论坛

J2EE 2009-06-18 19:54:35 阅读97 评论0

字号： 太 中 小

缓存是位于应用程序与物理数据源之间，用于临时存放复制数据的内存区域，目的是为了减少应用程序对物理数据源访问的次数，从而提高应用程序的运行性能。

Hibernate在查询数据时，首先到缓存中去查找，如果找到就直接使用，找不到的时候就会从物理数据源中检索，所以，把频繁使用的数据加载到缓存区后，就可以大大减少应用程序对物理数据源的访问，使得程序的运行性能明显的提升。

Session缓存，一级缓存。

SessionFactory的缓存分为内置缓存和外置缓存.内置缓存中存放的是SessionFactory对象的一些集合属性包含的数据(映射元素据及预定义SQL语句等),对于应用程序来说,它是只读的.外置缓存中存放的是数据库数据的副本,其作用和一级缓存类似.二级缓存除了以内存作为存储介质外,还可以选用硬盘等外部存储设备。

Hibernate的一级缓存和二级缓存都位于均位于持久层,且均用于存放数据库数据的副本,最大的区别就是缓存的范围各不一样。

缓存的范围分为3类:

- 1.事务范围
- 事务范围的缓存只能被当前事务访问,每个事务都有各自的缓存,缓存内的数据通常采用相互关联的对象形式.缓存的生命周期依赖于事务的生命周期,只有当事务结束时,缓存的生命周期才会结束.事务范围的缓存使用内存作为存储介质,一级缓存就属于事务范围.
- 2.应用范围
- 应用程序的缓存可以被应用范围内的所有事务共享访问.缓存的生命周期依赖于应用的生命周期,只有当应用结束时,缓存的生命周期才会结束.应用范围的缓存可以使用内存或硬盘作为存储介质,二级缓存就属于应用范围。
- 3.集群范围
- 在集群环境中,缓存被一个机器或多个机器的进程共享,缓存中的数据被复制到集群环境中的每个进程节点,进程间通过远程通信来保证缓存中的数据的一致,缓存中的数据通常采用对象的松散数据形式。



zhangqia

天生我材必有用！耶稣说的。。。

加博友 关注他

最新日志

hibernate 需要的包

个人总结flex各种用法 (□

C# 导入EXCEI

Flex 错误小结

网站优化

大学生必须知道20条生□

相关日志

getHibernateTemplate.I□

Hibernate应用 批量处理

Hibernate用后感一

hibernate 二级缓存

hibernate 二级缓存配置

hibernate 二级缓存配置 -□

查看更多>>

推荐阅读

“PÊ³¼ãÃ»”ÔÖÐÓŒ

À©xâ·¿±»ÄxÆúÊÊ°□

°ÃÖPÖ»Ö!âÄê»ýÐi³íÃ□

°£±âµÄ¹·ÊÇÖýxÙì!áíÁ¹·

±¼½i½ôíñ½±¶¼»²xÄ,Ü□

‘Óx”xÓÂÇÄiÔÖÁê,ÐÐ”□

更多>>

一级缓存的管理:

`evict(Object obj)` 将指定的持久化对象从一级缓存中清除,释放对象所占用的内存资源,指定对象从持久化状态变为脱管状态,从而成为游离对象.

`clear()` 将一级缓存中的所有持久化对象清除,释放其占用的内存资源

`contains(Object obj)` 判断指定的对象是否存在于一级缓存中.

`flush()` 刷新一级缓存区的内容,使之与数据库数据保持同步.

二级缓存的管理:

`evict(Class arg0, Serializable arg1)` 将某个类的指定ID的持久化对象从二级缓存中清除,释放对象所占用的资源.

```
sessionFactory.evict(Customer.class, new Integer(1));
```

`evict(Class arg0)` 将指定类的所有持久化对象从二级缓存中清除,释放其占用的内存资源.

```
sessionFactory.evict(Customer.class);
```

`evictCollection(String arg0)` 将指定类的所有持久化对象的指定集合从二级缓存中清除,释放其占用的内存资源.

```
sessionFactory.evictCollection("Customer.orders");
```

首先,不是所有的数据都适合放在二级缓存中,看一下,什么样的数据适合放在二级缓存中来?什么样的数据不适合放在二级缓存中来?

下面这几种情况就不适合加载到二级缓存中:

- 1.经常被修改的数据
- 2.绝对不允许出现并发访问的数据
- 3.与其他应用共享的数据

下面这几种情况合适加载到二级缓存中:

- 1.数据更新频率低
- 2.允许偶尔出现并发问题的非重要数据
- 3.不会被并发访问的数据
- 4.常量数据
- 5.不会被第三方修改的数据

Hibernate的二级缓存功能是靠配置二级缓存插件来实现的,Hibernate为了集成这些插件,Hibernate提供了`org.hibernate.cache.CacheProvider`借口,它充当缓存插件与Hibernate之间的适配器.

常用的二级缓存插件

`EHCache org.hibernate.cache.EhCacheProvider`

`OSCache org.hibernate.cache.OSCacheProvider`

`SwarmCahe org.hibernate.cache.SwarmCacheProvider`

`JBossCache org.hibernate.cache.TreeCacheProvider`

简单介绍一下EHCache的配置

`hibernate.cfg.xml`

```
<><propertyname="hibernate.cache.provider_class">    org.hibernate.cache.EhCacheProvider
<propertyname="hibernate.cache.use_query_cache">    true </>
```

`ehcache.xml`

<!-- maxElementsInMemory为缓存对象的最大数目, eternal设置是否永远不过期,timeToIdleSeconds对象处于空闲状态的最多秒数,timeToLiveSeconds对象处于缓存状态的最多秒数 --

><diskStorepath="java.io.tmpdir"/><defaultCache<maxElementsInMemory="10000"eternal="false"timeToIdleSeconds="300"

****.hbm.xml

<?xmlversion="1.0"encoding='UTF-8'?><!DOCTYPE hibernate-mapping PUBLIC
//Hibernate/Hibernate Mapping DTD 3.0//EN" <!-- 设置该持久化类的二级缓存并发访问策略 read-only read-
write nonstrict-read-write transactional--><cacheusage="read-write"/>

最近用上了,看看了,有什么不妥的地方,多谢指出.

引文来源 [Hibernate缓存机制](#) - [Hibernate](#) - [Java](#) - [JavaEye论坛](#)



阅读(97) | 评论(0) | 引用(0) | 举报

上一篇: [使用MyEclipse集成SSH和DWR\(三\)整合Spring和DWR](#) - [金庸Ematrix](#) - [CSDN博客](#)

下一篇: [Hibernate缓存概述](#) - [pansu的专栏](#) - [CSDN博客](#)

最近读者

评论

[点击登录](#) | 昵称: