

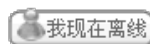
hanfei114

永久域名 <http://hanfei114.javaeye.com>**hanfei114**

浏览: 1326 次

性别:

来自: 南京

[详细资料](#)[留言簿](#)

搜索本博客

最近访客 [>>更多访客](#)[一段沧桑](#)[543229143](#)[ren008598](#)[susam](#)

博客分类

- [全部博客 \(14\)](#)

其他分类

- [我的收藏](#) (0)
- [我的论坛主题贴](#) (0)
- [我的所有论坛贴](#) (0)
- [我的精华良好贴](#) (0)

最近加入圈子

存档

- [2010-03](#) (5)
- [2010-01](#) (2)
- [2009-12](#) (7)
- [更多存档...](#)

最新评论

[数据库对象缓存](#) | [如何高效删除Oracle数据库中的重复数据](#)

2010-03-01

Hibernate缓存

文章分类: [Java编程](#)

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.

Hibernate缓存分类:

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的缓存范围

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.

Hibernate的缓存管理

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.

Java代码

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

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

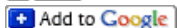
evict(Class arg0) 将指定类的所有持久化对象从二级缓存中清除,释放其占用的内存资源.

Java代码

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

评论排行榜

- [Eclipse快捷键大全](#)
- [Hibernate缓存](#)
- [Hibernate QBC 查询](#)
- [javascript时区转换](#)
- [如何高效删除Oracle数据库中的重复数据](#)



[\[什么是RSS?\]](#)

```
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.
Java代码
```

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

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

Hibernate的二级缓存的配置

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

Xml代码

```
<hibernate-configuration>
  <session-factory>
    <!-- 设置二级缓存插件EHCache的Provider类-->
    <property name="hibernate.cache.provider_class">
      org.hibernate.cache.EhCacheProvider
    </property>
    <!-- 启动"查询缓存" -->
    <property name="hibernate.cache.use_query_cache">
      true
    </property>
  </session-factory>
</hibernate-configuration>
```

```
<hibernate-configuration>
  <session-factory>
    <!-- 设置二级缓存插件EHCache的Provider类-->
    <property name="hibernate.cache.provider_class">
      org.hibernate.cache.EhCacheProvider
    </property>
    <!-- 启动"查询缓存" -->
    <property name="hibernate.cache.use_query_cache">
      true
    </property>
  </session-factory>
</hibernate-configuration>
```

ehcache.xml

Xml代码

```
<ehcache>
```

```
<diskStore path="java.io.tmpdir"/>
  <defaultCache maxElementsInMemory="10000" eternal="false" timeToIdleSeconds="300" timeToLiveSeconds="600"
overflowToDisk="true"/>
</ehcache>
```

```
<diskStore path="java.io.tmpdir"/>
  <defaultCache maxElementsInMemory="10000" eternal="false" timeToldSeconds="300" timeToLiveSeconds="600"
overflowToDisk="true"/>
</ehcache>
```

Xml代码

<hibernate-mapping>

</hibernate-mapping>

[数据库对象缓存](#) | [如何高效删除Oracle数据库中的重复数据](#)

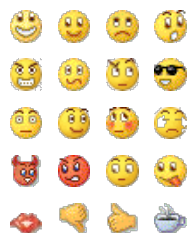
23:02 | 浏览 (69) | [评论 \(0\)](#) | [相关推荐](#)

评论

发表评论

表情图标

字体颜色: 字体大小: 对齐:



提示：选择您需要装饰的文字，按上列按钮即可添加上相应的标签

您还没有登录，请[登录](#)后发表评论(快捷键 **Alt+S** / **Ctrl+Enter**)