

# Spring——Java程序员的春天

永久域名 <http://hanqunfeng.iteye.com>



hanqunfeng

浏览: 40236 次

性别:

来自: 北京

我现在离线

详细资料 留言簿

搜索本博客

最近访客 [>>更多访客](#)



macula



joeyhacker



Altered



arele

博客分类

- 全部博客 (135)
- JAVA (19)
- Spring (29)
- Linux (18)
- JavaScript (7)
- Scrum (23)
- IDE (3)
- JSON (1)
- Solr (0)
- Webharvest (0)
- Hibernate (6)
- 杂谈 (3)
- Windows 7 (1)
- 持续集成 (23)

2011-08-26

◀ [SpringSecurity3.X--Cas client 配置](#) | [SpringSecurity3.X--验证码](#) ▶

## Spring3.1 Cache注解

博客分类: [Spring](#)

[spring](#) [ehcache](#)

需要感慨一下，spring3.0时丢弃了2.5时的spring-modules-cache.jar，致使无法使用spring来方便的管理cache注解，好在3.1.M1中增加了对cache注解的支持，可喜可贺啊！

希望了解spring2.5的cache注解，可以参考如下内容：

[Spring基于注解的缓存配置--EHCache AND OSCache](#)

[Spring基于注解的缓存配置--web应用实例](#)

2.5时，spring没有自己的解决方案，都是采用对许多第三方cache框架的支持，比如EHCache和OSCache等等，不过到了3.1，spring就只提供EHCache的支持了，不过spring3.1还给出了自己的解决方案。

下面简单介绍一下spring3.1.M1中的cache功能。

spring3.1.M1中负责cache的模块是org.springframework.context-3.1.0.M1.jar

与2.5时的modules模块类似，3.1的注解缓存也是在方法上声明注解，3.1同样提供了两个注解：

@Cacheable：负责将方法的返回值加入到缓存中

@CacheEvict：负责清除缓存

@Cacheable 支持如下几个参数：

value：缓存位置名称，不能为空，如果使用EHCache，就是ehcache.xml中声明的cache的name

key：缓存的key，默认为空，既表示使用方法的参数类型及参数值作为key，支持SpEL

condition：触发条件，只有满足条件的情况才会加入缓存，默认为空，既表示全部都加入缓存，支持SpEL

例如：

Java代码

```
1. //将缓存保存进andCache，并使用参数中的userId加上一个字符串(这里使用方法名称)作为缓存的key
2. @Cacheable(value="andCache",key="#userId + 'findById'")
3. public SystemUser findById(String userId) {
4.     SystemUser user = (SystemUser) dao.findById(SystemUser.class, userId);
5.     return user ;
6. }
7. //将缓存保存进andCache，并当参数userId的长度小于32时才保存进缓存，默认使用参数值及类型作为缓存的key
8. @Cacheable(value="andCache",condition="#userId.length < 32")
9. public boolean isReserved(String userId) {
10.     System.out.println("hello andCache"+userId);
11.     return false;
12. }
```

@CacheEvict 支持如下几个参数：

- [tomcat \(2\)](#)
- [Android \(1\)](#)
- [SpringSecurity \(5\)](#)

我的留言簿 [>>更多留言](#)

- 用eclipse开发android 右键run as 怎么只有run config ...  
-- by [ergouge](#)
- 好厉害 我是王晨，来顶顶=  
-- by [kitta](#)
- 如何让redis在Wondows中开机自启动？我用如下的指令没有成功 sc cr ...  
-- by [kane\\_easy](#)

其他分类

- [我的收藏 \(22\)](#)
- [我的代码 \(0\)](#)
- [我的书籍 \(2\)](#)
- [我的论坛主题帖 \(16\)](#)
- [我的所有论坛帖 \(30\)](#)
- [我的精华良好帖 \(0\)](#)

最近加入群组

存档

- [2011-10 \(1\)](#)
- [2011-09 \(2\)](#)
- [2011-08 \(5\)](#)
- [更多存档...](#)

评论排行榜

- [Eclipse开发环境搭建](#)
- [Spring2.5MVC请求处理时序图](#)
- [spring2.5注解介绍 \(3.0通用\)](#)
- [SpringSecurity3.X--Cas client 配置](#)
- [SpringSecurity3.X--一个简单实现](#)



value：缓存位置名称，不能为空，同上

key：缓存的key，默认为空，同上

condition：触发条件，只有满足条件的情况才会清除缓存，默认为空，支持SpEL

allEntries：true表示清除value中的全部缓存，默认为false

例如：

Java代码



```
1. //清除掉指定key的缓存
2. @CacheEvict(value="andCache",key="#user.userId + 'findById'")
3. public void modifyUserRole(SystemUser user) {
4.     System.out.println("hello andCache delete"+user.getUserId());
5. }
6.
7. //清除掉全部缓存
8. @CacheEvict(value="andCache",allEntries=true)
9. public final void setReservedUsers(String[] reservedUsers) {
10.     System.out.println("hello andCache deleteall");
11. }
```

一般来说，我们的更新操作只需要刷新缓存中某一个值，所以定义缓存的key值的方式就很重要，最好是能够唯一，因为这样可以准确的清除掉特定的缓存，而不会影响到其它缓存值，比如我这里针对用户的操作，使用(userId+方法名称)的方式设定key值，当然，你也可以找到更适合自己的方式去设定。

SpEL：Spring Expression Language

关于SpEL的介绍，可以参考如下地址：

<http://static.springsource.org/spring/docs/3.1.0.M1/spring-framework-reference/html/expressions.html>

了解了cache的注解之后，接下来说说如何使注解生效，其实就是需要在spring的配置文件中增加一些配置。

## 1.spring-cache

首先我们来看一下如何使用spring3.1自己的cache，需要在命名空间中增加cache的配置

Xml代码



```
1. <beans xmlns="http://www.springframework.org/schema/beans"
2.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:p="http://www.springframework.org/schema/p"
3.     xmlns:cache="http://www.springframework.org/schema/cache"
4.     xsi:schemaLocation="
5.         http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-3.1.xsd
6.         http://www.springframework.org/schema/cache http://www.springframework.org/schema/cache/spring-cache-3.1.xsd">
```

之后添加如下声明：

Xml代码




```
1. <!-- 启用缓存注解功能，这个是必须的，否则注解不会生效，另外，该注解一定要声明在spring主配置文件中才会生效 -->
2. <cache:annotation-driven cache-manager="cacheManager"/>
3.
4.
```

```
5.      <!-- spring自己的换管理器，这里定义了两个缓存位置名称，既注解中的value -->
6.      <bean id="cacheManager" class="org.springframework.cache.support.SimpleCacheManager">
7.          <property name="caches">
8.              <set>
9.                  <bean
10.                      class="org.springframework.cache.concurrent.ConcurrentCacheFac
11.                      toryBean"
12.                      p:name="default" />
13.                  <bean
14.                      class="org.springframework.cache.concurrent.ConcurrentCacheFac
15.                      toryBean"
16.                      p:name="andCache" />
17.              </set>
18.          </property>
19.      </bean>
```


2.spring-ehcache

接下来说说对ehcache的支持，其实只需要把cacheManager换成EhCache的cacheManager即可，如下：

Xml代码 

```
1.      <!-- 启用缓存注解功能，这个是必须的，否则注解不会生效，另外，该注解一定要声明在spring主配置文件中才会生效 -->
2.      <cache:annotation-driven cache-manager="cacheManager"/>
3.
4.      <!-- cacheManager工厂类，指定ehcache.xml的位置 -->
5.      <bean id="cacheManagerFactory" class="org.springframework.cache.ehcache.EhCacheManagerFactoryBean"
6.          p:configLocation="classpath:/config/ehcache.xml" />
7.
8.      <!-- 声明cacheManager -->
9.      <bean id="cacheManager" class="org.springframework.cache.ehcache.EhCacheCacheManager"
10.         p:cacheManager-ref="cacheManagerFactory" />
```

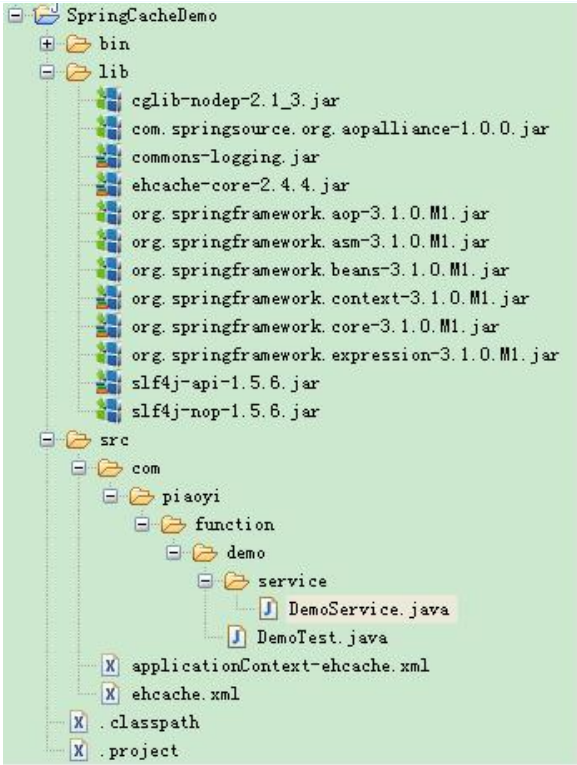
ehcache.xml

Xml代码 

```
1.      <?xml version="1.0" encoding="UTF-8"?>
2.      <ehcache xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3.          xsi:noNamespaceSchemaLocation="ehcache.xsd" updateCheck="true"
4.          monitoring="autodetect">
5.          <!--
6.          <diskStore path="java.io.tmpdir" /> -->
7.          <diskStore path="E:/cachetmpdir"/>
8.          <defaultCache maxElementsInMemory="10000" eternal="false"
9.              timeToIdleSeconds="120" timeToLiveSeconds="120" overflowToDisk="true"
10.             maxElementsOnDisk="10000000" diskPersistent="false"
11.             diskExpiryThreadIntervalSeconds="120" memoryStoreEvictionPolicy="LRU" />
12.
13.          <cache name="andCache" maxElementsInMemory="10000"
14.              maxElementsOnDisk="1000" eternal="false" overflowToDisk="true"
15.              diskSpoolBufferSizeMB="20" timeToIdleSeconds="300" timeToLiveSeconds="600"
16.              memoryStoreEvictionPolicy="LFU" />
17.      </ehcache>
```

ok，这样注解缓存就生效了。

附件中是我自己写的一个小例子，工程结构如下所示，运行com.piaoyi.function.demo下的DemoTest即可



参考资料:

<http://blog.springsource.com/2011/02/23/spring-3-1-m1-caching/>

<http://hi.baidu.com/coolcooldool/blog/item/3b541533c72b40e21a4cffda.html>

[SpringCacheDemo.rar](#) (3.1 MB)

下载次数: 305

[查看图片附件](#)



分享到:

◀ [SpringSecurity3.X--Cas client 配置](#) | [SpringSecurity3.X--验证码](#) ▶

15:46 | [评论 / 浏览 \(0 / 246\)](#) | [论坛回复 / 浏览 \(8 / 1983\)](#) | [分类:编程语言](#) | [相关推荐](#) [▶ MORE](#)

评论

发表评论



[您还没有登录,请您登录后再发表评论](#)

声明：ITeye 文章版权属于作者，受法律保护。没有作者书面许可不得转载。若作者同意转载，必须以超链接形式标明文章原始出处和作者。  
© 2003-2011 ITeye.com. All rights reserved. [ 京ICP证110151号 京公网安备110105010620 ]