项目是使用spring ，hibernate的java web项目，启动时报错：

Caused by: java.lang.NoClassDefFoundError: Lorg/hibernate/cache/CacheProvider;

at java.lang.Class.getDeclaredFields0(Native Method)

at java.lang.Class.privateGetDeclaredFields(Class.java:2291)

at java.lang.Class.getDeclaredFields(Class.java:1743)

at org.springframework.orm.jpa.support.PersistenceAnnotationBeanPostProcessor.findPersistenceMetadata(PersistenceAnnotationBeanPostProcessor.java:374)

at org.springframework.orm.jpa.support.PersistenceAnnotationBeanPostProcessor.postProcessMergedBeanDefinition(PersistenceAnnotationBeanPostProcessor.java:322)

at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.applyMergedBeanDefinitionPostProcessors(AbstractAutowireCapableBeanFactory.java:840)

at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.doCreateBean(AbstractAutowireCapableBeanFactory.java:495)

at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.createBean(AbstractAutowireCapableBeanFactory.java:458)

at org.springframework.beans.factory.support.AbstractBeanFactory$1.getObject(AbstractBeanFactory.java:295)

at org.springframework.beans.factory.support.DefaultSingletonBeanRegistry.getSingleton(DefaultSingletonBeanRegistry.java:223)

at org.springframework.beans.factory.support.AbstractBeanFactory.doGetBean(AbstractBeanFactory.java:292)

at org.springframework.beans.factory.support.AbstractBeanFactory.getBean(AbstractBeanFactory.java:198)

at org.springframework.beans.factory.support.AbstractBeanFactory.doGetBean(AbstractBeanFactory.java:270)

at org.springframework.beans.factory.support.AbstractBeanFactory.getBean(AbstractBeanFactory.java:198)

at org.springframework.context.annotation.CommonAnnotationBeanPostProcessor.autowireResource(CommonAnnotationBeanPostProcessor.java:442)

at org.springframework.context.annotation.CommonAnnotationBeanPostProcessor.getResource(CommonAnnotationBeanPostProcessor.java:416)

at org.springframework.context.annotation.CommonAnnotationBeanPostProcessor$ResourceElement.getResourceToInject(CommonAnnotationBeanPostProcessor.java:550)

at org.springframework.beans.factory.annotation.InjectionMetadata$InjectedElement.inject(InjectionMetadata.java:159)

at org.springframework.beans.factory.annotation.InjectionMetadata.inject(InjectionMetadata.java:87)

at org.springframework.context.annotation.CommonAnnotationBeanPostProcessor.postProcessPropertyValues(CommonAnnotationBeanPostProcessor.java:303)

... 31 more

 spring 的配置文件（beans.xml）如下：

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:context="http://www.springframework.org/schema/context"

xmlns:aop="http://www.springframework.org/schema/aop" xmlns:tx="http://www.springframework.org/schema/tx"

xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans-3.1.xsd

http://www.springframework.org/schema/context

http://www.springframework.org/schema/context/spring-context-3.1.xsd

http://www.springframework.org/schema/aop

http://www.springframework.org/schema/aop/spring-aop-3.1.xsd

http://www.springframework.org/schema/tx

http://www.springframework.org/schema/tx/spring-tx-3.1.xsd" default-lazy-init="true" >

<context:annotation-config />

<context:component-scan base-package="com" />

<bean id="dataSource" destroy-method="close"

class="org.apache.commons.dbcp.BasicDataSource">

<property name="driverClassName" value="${jdbc.driverClassName}" />

<property name="url" value="${jdbc.url}" />

<property name="username" value="${jdbc.username}" />

<property name="password" value="${jdbc.password}" />

</bean>

<bean

class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">

<property name="locations">

<value>classpath:jdbc.properties</value>

</property>

</bean>

<bean id="sessionFactory"

class="org.springframework.orm.hibernate3.annotation.AnnotationSessionFactoryBean">

<property name="dataSource" ref="dataSource" />

<property name="packagesToScan">

<list>

<value>com.kingbase.domain</value>

</list>

</property>

<property name="hibernateProperties">

<props>

<prop key="hibernate.dialect">

org.hibernate.dialect.MySQLDialect

</prop>

<!-- <prop key="hibernate.dialect">

org.hibernate.dialect.PostgreSQLDialect

</prop> -->

<prop key="hibernate.show\_sql">true</prop>

<prop key="hibernate.format\_sql">true</prop>

<prop key="hibernate.hbm2ddl.auto">update</prop>

<prop key="current\_session\_context\_class">thread</prop>

</props>

</property>

</bean>

<bean id="txManager"

class="org.springframework.orm.hibernate3.HibernateTransactionManager">

<property name="sessionFactory" ref="sessionFactory"></property>

</bean>

<!-- 事务的注解，如 @Transactional(readOnly=true) -->

<!--<tx:annotation-driven transaction-manager="txManager" />

-->

<aop:config>

<aop:pointcut id="bussinessService"

expression="execution(public \* com.kingbase.dao..\*.\*(..))" />

<aop:advisor pointcut-ref="bussinessService" advice-ref="txAdvice" />

</aop:config>

<tx:advice id="txAdvice" transaction-manager="txManager">

<tx:attributes>

<tx:method name="check\*" read-only="true" />

<tx:method name="count\*" read-only="true" />

<tx:method name="search\*" read-only="true" />

<tx:method name="get\*" read-only="true" />

<tx:method name="test\*" read-only="true" />

<tx:method name="is\*" read-only="true" />

<tx:method name="show\*" read-only="true" />

<tx:method name="delete\*" propagation="REQUIRED" />

<tx:method name="update\*" propagation="REQUIRED" />

<tx:method name="save\*" propagation="REQUIRED" />

<tx:method name="add\*" propagation="REQUIRED" />

<tx:method name="set\*" propagation="REQUIRED" />

<tx:method name="verify\*" propagation="REQUIRED" />

<tx:method name="create\*" propagation="REQUIRED" />

<tx:method name="judge\*" propagation="REQUIRED" />

</tx:attributes>

</tx:advice>

</beans>

我使用hibernate的版本是： **4.2.2.Final**

spring的版本是：3.2.3.RELEASE

原因：hibernate的版本升为 **4** 了，但是配置文件中仍使用的是hibernate3

解决办法：

把spring配置文件中的

org.springframework.orm.hibernate3.annotation.AnnotationSessionFactoryBean

改为：

org.springframework.orm. **hibernate4.LocalSessionFactoryBean**

就OK 了

详情请参阅： <http://stackoverflow.com/questions/7528862/exception-noclassdeffounderror-for-cacheprovider>

修改后的beans.xml 见附件。