

Déclaration de proxies transactionnels avec [Spring Framework](#)

Ca fait quelques jours que je m'intéresse sur comment réduire le nombre de lignes pour la déclaration d'un service transactionnel avec [spring](#). Faute de temps (tête dans le guidon), j'ai pas regardé la doc et j'ai commencé à inventer des mots clés (dont je ne parlerai pas ici) pour étendre la DTD du fichier de config spring...

5 minutes de parcours de doc ont stoppé toute réflexion.

Voici différentes manières de faire que j'ai retenu.

Solution 1 : Déclaration d'un proxy transactionnel

```
<!-- gestionnaire de transaction -->
<bean id="monTransactionManager"
  class="org.springframework.orm.hibernate.HibernateTransactionManager">
  <property name="sessionFactory">
    <ref bean="mySessionFactory"/>
  </property>
</bean>

<!-- dao -->
<bean name="monDao" class="org.bellard.dao.impl.MonDaoImpl">
</bean>

<!-- service cible -->
<bean name="monServiceTarget" class="org.bellard.service.impl.MonServiceImpl">
  <ref bean="monDao"/>
</bean>

<!-- proxy transactionnel -->
<bean id="monService"
  class="org.springframework.transaction.interceptor.TransactionProxyFactoryBean">
  <property name="transactionManager">
    <ref bean="monTransactionManager"/>
  </property>
  <property name="target">
    <ref bean="monServiceTarget"/>
  </property>
  <property name="transactionAttributes">
    <props>
      <prop key="get*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="find*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="load*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="save*">PROPAGATION_REQUIRED</prop>
    </props>
  </property>
</bean>
```

Problème : Pour chaque service, il faut redéfinir les attributs transactionnels donc très lourd!

Solution 2 : Utilisation d'un modèle de créateur de proxy

```
<!-- modèle de créateur de proxy -->
<bean id="txProxyTemplate" abstract="true"
      class="org.springframework.transaction.interceptor.TransactionProxyFactoryBean">
  <property name="transactionManager">
    <ref bean="monTransactionManager"/>
  </property>
  <property name="transactionAttributes">
    <props>
      <prop key="get*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="find*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="load*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="save*">PROPAGATION_REQUIRED</prop>
    </props>
  </property>
</bean>

<!-- service cible -->
<bean name="monServiceTarget" class="org.bellard.service.impl.MonServiceImpl">
  <ref bean="monDao"/>
</bean>

<!-- proxy transactionnel -->
<bean id="monService"
      parent="txProxyTemplate">
  <property name="target">
    <ref bean="monServiceTarget"/>
  </property>
</bean>
```

=> la déclaration du proxy transactionnel est beaucoup plus réduite!

=> Possibilité d'ajouter des attributs transactionnels propres au service si besoin

Problème : Il faut tout de même déclarer un proxy pour tous les services :-(

Solution 3 : Solution 2 avec la déclaration du service cible dans la déclaration du proxy

```
<!-- service cible dans proxy transactionnel -->
<bean id="monService"
  parent="txProxyTemplate">
  <property name="target">
    <bean name="monServiceTarget" class="org.bellard.service.impl.MonServiceImpl">
      <ref bean="monDao"/>
    </bean>
  </property>
</bean>
```

=> on gagne 1 ligne mais on ne peut plus utiliser le service sans le proxy transactionnel
Le gain est donc négligeable

Solution 4 : Utilisation des Autoproxy

L'autoproxy est un post-processeur de configuration spring: il est activé une fois l'ensemble des beans initialisés.

Il permet la création de proxy sur les beans dont le nom vérifie un pattern : ***Service** dans l'exemple.

```
<!-- transaction interceptor -->
<bean id="monTransactionInterceptor"
  class="org.springframework.transaction.interceptor.TransactionInterceptor">
  <property name="transactionManager">
    <ref bean="monTransactionManager"/>
  </property>
  <property name="transactionAttributes">
    <props>
      <prop key="get*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="find*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="load*">PROPAGATION_REQUIRED,readOnly</prop>
      <prop key="save*">PROPAGATION_REQUIRED</prop>
    </props>
  </property>
</bean>

<!-- autoproxy -->
<bean id="transactionBeanNameProxyCreator"
  class="org.springframework.aop.framework.autoproxy.BeanNameAutoProxyCreator">
  <property name="beanNames"><value>*Service</value></property>
  <property name="interceptorNames">
    <list>
      <value>monTransactionInterceptor</value>
    </list>
  </property>
</bean>
```

```
</property>
</bean>

<!-- service cible -->
<bean name="monService" class="org.bellard.service.impl.MonServiceImpl">
  <ref bean="monDao"/>
</bean>
```

=> Tous les beans ayant le suffixe **Service** sont déclarés comme proxy transactionnel en 1 seule définition!

C'est pour moi de loin la meilleure solution pour la déclaration de proxies transactionnels.