

JSF请求处理过程（一）FacesServlet初始化Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/644/2021\\_2022\\_JSFE8\\_AF\\_B7\\_E6\\_B1\\_82\\_E5\\_c104\\_644750.htm](https://www.100test.com/kao_ti2020/644/2021_2022_JSFE8_AF_B7_E6_B1_82_E5_c104_644750.htm) JSF应用中，在web.xml里面配置了一个Servlet，叫做javax.faces.webapp.FacesServlet，于是可以知道，查看、了解一个请求的处理过程可以从这里开始。从官方网站上下载JSF的源代码，项目名比较古怪，叫做“mojarra”，我看的版本是1.2\_12\_b01。里面包含了两个子项目，一个是jsf-api，里面大多是接口以及少量关键类。另外一个项目叫做jsf-ri，对着这个“ri”邪念了半天之后，终于在兄弟提醒之下想明白了是reference implementation的意思。jsf-api是JavaEE标准的一部分，里面的类型包名都是以javax.faces开头的，而jsf-ri项目是sun针对JSF标准的一个参考实现，里面的类型的包名都是以com.sun.faces开头的。FacesServlet初始化（FacesServlet#init）系统启动的时候，会初始化FacesServlet，调用其中的init方法。里面主要做了两件事情，一个是初始化FacesContextFactory，另外一个初始化Lifecycle对象。在jsf-api项目中，FacesServlet类是一个Servlet接口的实现类，而FacesContextFactory和Lifecycle都是接口。在jsf-ri项目中有这两个接口的实现类，分别是com.sun.faces.context.FacesContextFactoryImpl和com.sun.faces.lifecycle.LifecycleImpl类。一个想当然的事实：FacesServlet初始化的时候要根据一些配置来判断具体的FacesContextFactory和Lifecycle实现类是什么，也就是在这里，“JSF标准”和“JSF实现”接轨了。想来MyFaces等等的其他JSF实现应该不外乎两种方式，一种是改变FacesServlet的init

方法中需要用到的配置的值，于是启用自己的FacesContextFactory实现和Lifecycle实现，后面的处理过程就全部走自己的逻辑了。第二种方法笨一点，可能性不大，就是把FacesServlet覆盖替换掉，其中也不需要读什么配置了，直接使用自己的实现类即可不过这种做法估计不符合JSF规范，想来只有我等虾米民众能做的出来。主要代码如下：1

```
facesContextFactory = ( FacesContextFactory
```

```
) FactoryFinder.getFactory
```

```
( FactoryFinder.FACES_CONTEXT_FACTORY ) ; 2
```

```
LifecycleFactory lifecycleFactory = ( LifecycleFactory
```

```
) FactoryFinder.getFactory
```

```
( FactoryFinder.LIFECYCLE_FACTORY ) ; 回头再来看初始
```

化的结果，FacesContextFactory很明显是用来生产FacesContext这么个东西的。而FacesContext可以看做是一

个RequestWrapper（注意这个FaceContext和ServletContext不一样，ServletContext是一个Web应用只有一个的全局对象，对应的是一个Web application，而一个FacesContext对应的是一个request，另外，RequestWrapper这个说法不严格，实际上FacesContext里面也包装了ServletContext、Response等）。

而LifeCycle可以看做是一个过滤器链（类似于servlet规范里面的Filter Chain）。于是，整个JSF请求处理过程，实际上就是包装成为FaceContext的用户请求，通过类似于一个Filter Chain的LifeCycle的过程。100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)