

jBPM4的运行期环境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_ti2020/641/2021_2022_jBPM4_E7_9

A_84_E8_BF_c104_641959.htm 万物生长靠太阳，儿童的生长离不开土壤、空气和水，当然，也离不开绿坝娘的调教。应用程序也是如此，离不开数据库连接、事务、日志、消息等，这些，共同构成了应用程序的运行期环境。理想中的环境是什么样子的哩。好吧，一句话，召之即来，挥之即去，当需要某个服务时，ok，打个响指，该服务就准备好被调用了，调用完毕后也不用费心费力地擦屁股，不必老是提心吊胆。有好事者追问：你擦了吗，确定擦了？真的确定擦了？直接丢弃给环境降解处理，自然又环保，还有个好名声叫专注领域逻辑。

一、运行期环境就是一个餐馆

- 1、提供必要的服务
作为一个餐馆，必须有厨师做饭我吃，必须有桌子和椅子。作为运行期环境同样如此，我要发消息，你得提供我发消息的Service，我要获取节点任务，你得扔给我TaskService.
- 2、提供获取这些服务的统一方式
好吧，我不会亲自到厨房告诉厨师我想吃什么（因为我担心这样一来我会吃不下去），我也不会亲自到收银台给钱。这些服务有一个统一的获取方式：服务员。我想吃什么和结账，告诉服务员即可。关键是这一方式要统一，要足够简单。Spring最懒，把服务给你全部注入了，当然你也可以握住BeanFactory的纤纤细手，一个一个的get.
- 3、提供特定于我线程不安全的服务
我点了一盘鱼香肉丝，隔壁也点了一盘鱼香肉丝，结果服务员让我们吃同一盘鱼香肉丝。我立刻跳起来：靠，你们的服务不是线程安全的吗？！Hibernate的Session正是属于这么一种情况，需要环境

进行隔离，我的唯一职责就是吃饭！我的领域逻辑是如何优美的进餐！为此还要不断重构我吃饭的姿势哩。好不容易吃完饭，付完款，正准备离场。服务员风度翩翩地走到我的身旁，我以为还有打折券供应，结果是：服务员小姐轻启朱唇：先生，麻烦您把吃剩的盘子清洗完毕。崩溃！像数据库连接的打开，关闭、事务的打开、提交等都属于运行期环境应该做的事情。4、其他的七七八八杂事不少，例如统一的事件机制、权限拦截等等。二、jBPM4的运行期环境 好吧，先来看看如何建立jBPM4的运行期环境：EnvironmentFactory environmentFactory = new DefaultEnvironmentFactory().

```
Environment environment = environmentFactory.openEnvironment(). try { everything available in this block } finally { environment.close(). }
```

 两个关键的类

：EnvironmentFactory和Environment. EnvironmentFactory是全局的，在整个应用程序中保持一个实例即可。Environment则是每次方法调用则要new一个。看看Environment的主要方法

```
public abstract Object get(String name). public abstract T get(Class<T> type).
```

 是的，environment为我们的代码提供所需要的服务类实例。那么，如何获得environment？继续看：

```
public static Environment getCurrent();
```

 static，我喜欢也。方便、快捷，不管是在地上、车上还是房顶上，随处都可调用。

那么，为什么Environment每次调用要new呢？好吧，当你需要获取数据库Session的时候，是不是每次都要new呢

。Environment提供的服务里包括了非线程安全的数据库操作服务。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