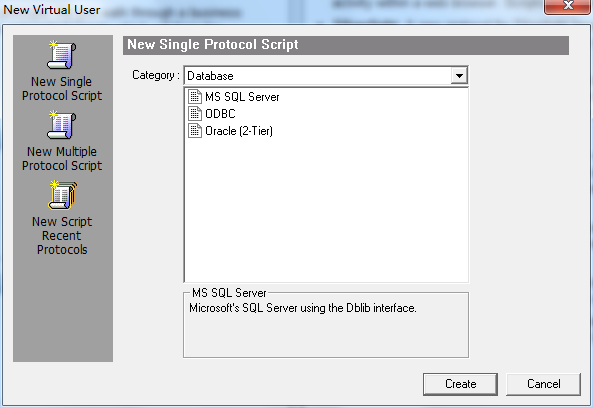
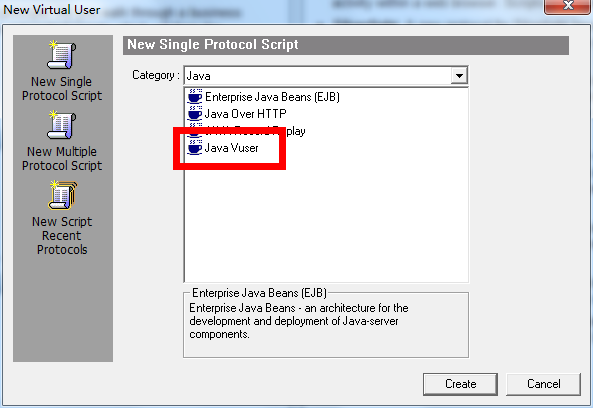
使用Loadrunner测试数据库性能

使用LR对数据库进行性能测试，实际上有多种办法，包括通过现有的数据库协议进行CS模式的先录制后执行的模式，以及通过socket方式向服务器发包方式的测试方式。这些是常规书籍上介绍的比较简单上手的测试方法，但是不具备通用性，受已有协议或socket编程方式的限制，所以需要更为通用的测试方法。

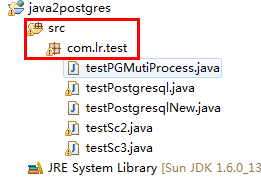


使用Java user的协议进行所有数据库性能的测试工作：

Java user 不需要录制，把所有的操作通过java语言进行实现，通过lr调用java的class进行加压批量操作，这样可以不关心被测系统是哪个数据库，只要能够通过jdbc进行访问，就能实现性能测试。



1. 测试环境准备
   1. 被测服务器准备，根据测试目的，搭建需要的数据库服务器，确保数据库能够正常访问，正常操作；
   2. Java代码的准备，无论使用哪种IDE，只要能够编写访问数据库的class就可以，形式可以是j2se，也可以是j2ee，因为在操作时只使用class的部分方法，所以j2ee就可以了；
   3. LR的脚本调试，把java的class导入到脚本调试模式，根据需要添加事务以及其他操作。
2. 编写数据库访问
   1. 使用myeclipse，创建web project，创建如下图的包目录：



Java文件中包含各种访问数据库的方法。



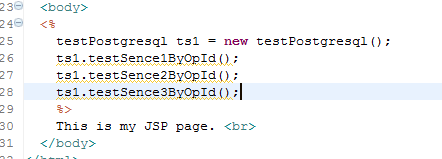
需要注意的是，class中的方法必须是public static，否则LR中无法调用。由于创建的是j2ee程序，所以不用main函数，在web中就可以进行功能验证。

确认class中的方法编写完成，创建一个web.jsp文件，如下：



导入class

声明类，并实例化，直接调用刚才编写的3个方法，因为这3个方法是直接对数据库进行操作，不需要实参，也没有返回值，所以直接实现即可。



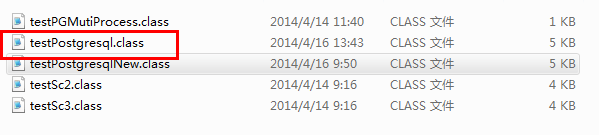
此时启动web服务，在浏览器中输入jsp的地址，直接刷新页面，就可以调用这3个方法，如果正确，就会对相应的表进行操作，如果不正确，则需要修改相应的代码。

* 1. LR脚本准备：

LR脚本实际上就是对访问代码的调用，关键在于需要根据测试场景划分不同的脚本布局。

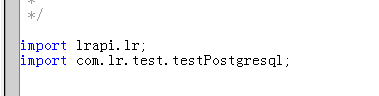
例如：在myEclipse里，我们只编写了一个class，其中包含三个方法，如果在执行性能测试时，这三个方法相互独立，互不干涉，则最简单的划分方法是，创建三个java user，每个java user中包含一个方法，做三份脚本，场景执行时分别进行调用。如果三个方法之间有相互关系，则需要根据实际情况，把有关联的方法放在一起，具体情况可按实际灵活分配。

因为已经将class文件进行编译发布了，所以可以在“java2postgres\WebRoot\WEB-INF\classes\com\lr\test”目录中找到对应的class文件，

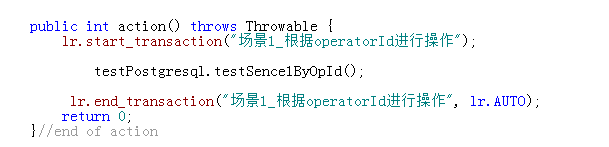


复制这个文件，找到LR的目录：HP\LoadRunner\classes\com\lr\test\ 如果没有文件夹，按相同的内容创建。

在LR脚本中进行引包操作：

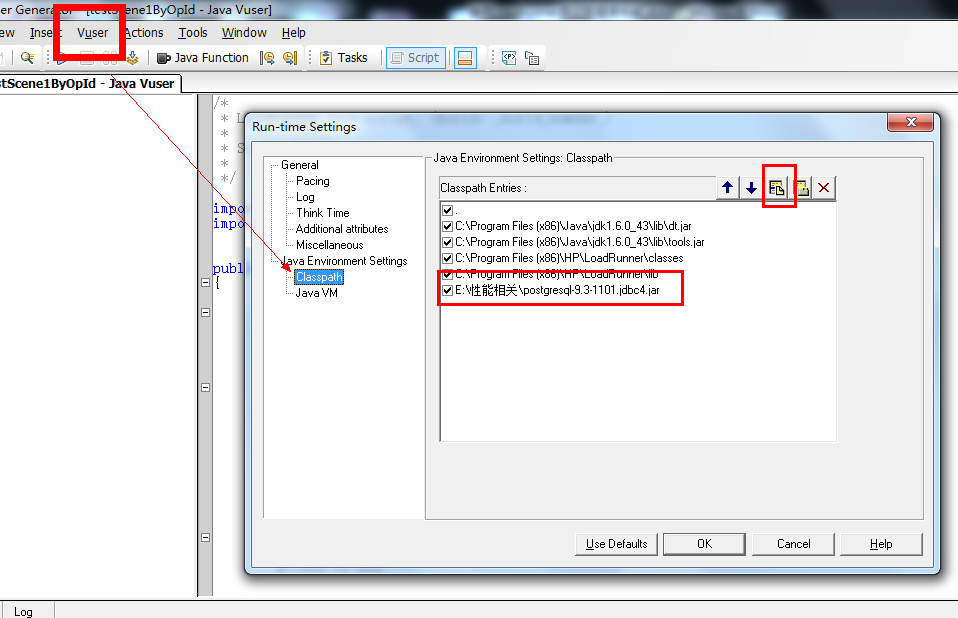


将需要执行的java类以及方法，放在action中，可根据实际测试情况和所需要验证的内容，具体调试代码。



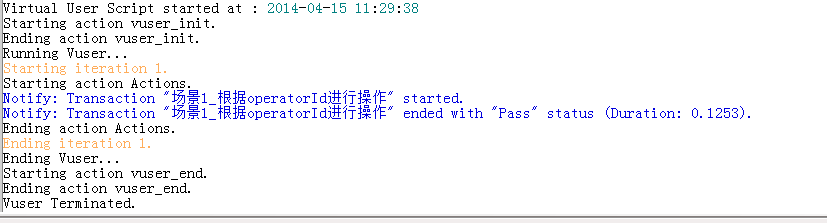
在这里可以像编写普通LR脚本一样，添加事务或集合点等内容。

由于是通过JDBC对数据库进行访问，因此要在java user中加载jdbc驱动。



运行时设置中，增加jdbc驱动，需要注意的是java user使用的本地jdk，需要至多1.6版本，若使用1.7版本，回放会有错误，这是jdk版本的问题。

操作完成之后，按F5或点击运行，进行脚本回放，实际此时也对数据库进行了操作，可以直接查询对应的表，检查功能是否正确。



1. 执行性能测试

已经有了java user的脚本，和普通web性能测试一样，设计场景、执行测试、收集报告、分析性能瓶颈即可。

