

导入:采用JSI封装、集成第三方类库 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/268/2021_2022__E5_AF_BC_E5_85_A5__E9_87_87_c67_268109.htm

Java的成功，离不开它那个庞大的类库，不单是sun的类库，很多细节的实现都取自第三方（如xml解析采用Apache的实现）。JSI暂时不大算编写丰富的公共API，但是我们可以集成其他成熟的类库，同时隔离他们的依赖，隔离各个脚本的执行上下文，消除命名冲突的危险。这里我们详细介绍一个复杂一点的实例：类似Windows XP文件浏览器左侧的滑动折叠面板(任务菜单)效果。我们先集成Scriptaculous Effect类库，并且在这个基础上按我个人的习惯对一个面板折叠效果做一个简单的封装，展示框架的类库封装功能。

1. 集成Scriptaculous类库：这里我们不做过多介绍，详细情况请参考集成实战；我们发布的版本中已经把Scriptaculous放置于us.aculo.script包中，您可以把这些作为系统内置的类库使用。
2. 编写我们的折叠面板函数（example/effect.js）：`onclick="slidePanel(menu_block1)"`这个事件函数将在我们点击面板标题时触发，能后会调用Scriptaculous Effect的具体实现去实现我们需要的滑动折叠功能。

壁立千仞 无欲则刚——控制依赖 java可以随意的使用第三方类库，可是JavaScript却没那么幸运，随着类库的丰富，烦杂的依赖关系和可能的命名冲突将使得类库的发展越来越困难。程序的易用性也将大打折扣。命名冲突的危险无形的增加你大脑的负担；随着使用的类库的增加，暴露的依赖也将随之增加，这是复杂度陡增的极大祸根，将使得系统越来越复杂，越来越难以控制。潜在的问题越来越多，防不胜

防。JSI的出现，可以解决上述问题，我们建议类库的开发者将自己类库的依赖终结在自己手中，避免依赖扩散，以提高类库的易用性。同样使用上面的例子，假如我们想抛开JSI，实现同样的功能，那我们的页面代码将是（类库代码不用改动）：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