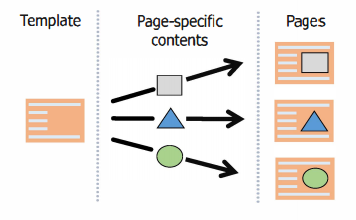
Mvc架构驱动的重构来实现客户端网页构成

摘要：本文呈现了一种将java web应用的页面构成从服务器重定位到浏览器的方式。目的是减少过多重复的操作和传输页面共享的代码或数据。减少量是动过执行重构算法产生的，从mvc架构的视角来看，有效地保证了源和目标应用的一致性，因为当前的问题是要求目标应用要保存源应用上的可观察的行为。研究案例表明，我们的重构工具能有效的支持重构的程序。

关键字：重构，mvc，单页应用，基于模板的网页应用，软件引擎，软件引擎。

1. **介绍**

在一个网页应用中，网页页面通常分享着一些内容，比如页头，页脚和菜单。在一块地方用一个模板去维护这些公共内容分隔页面上具体展示的内容。一个基于模板的web应用(TWA)使用这样一种模板去动态地生成页面，并将页面上具体的内容与模板进行绑定。举个例子，在图1中，这三个页面通过使用一个模板和每个页面具体的内容生成。

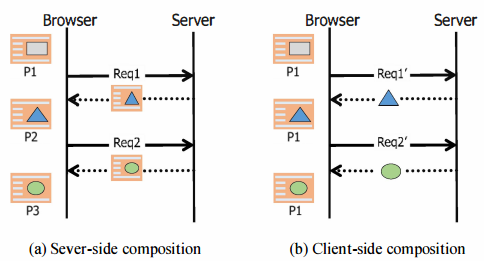


图一.基于模板的web应用

在java web应用中，<jsp:include>标准动作常用于将页面具体内容填充进模板的占位符中

当用户发出一个请求之后，一个网页浏览器下载和展现一个新的页面，TWAs 主要同过多页面应用模型与用户进行交互。如图2a展示的一样，一件最重要的记住的事情就是TWAs是对于每个请求，服务器都会结合对应的模板和页面具体内容并发送。因此对于么每个请求，模板数据被重复进行传输并用整个页面刷新进行渲染。

对于这种重复问题，其中一种解决方案是将页面构成由服务器转交给浏览器。无论用户何时点击一个超链接或是提交一个表单，网页浏览器接收到当前页面和新页面之前的不同处，这时局部地更新当前页面不同之处而不同过整个页面的刷新。如图2b所示。每个请求，请求1和请求2造成一次关于有联系的差异部分的部分页面刷新。页面1是物理上的一个单独页面但不能有不同的统一资源定位地址（URLs）,取决于他的状态（即包含的内容）。这种网页应用称它为单页应用（SPAs）。



图二.web页面组件

MVC架构被普遍用于网页应用开发中。因此，当重构或重建多页面应用成一个单页面应用时，它能很自然地考虑到三个组件和他们之间的交互作为一个基本的研究和改动的单元。然而，这些架构组件还没有在大多数之前的工程中充分地描述关于转化的问题。

这个转化是主要要求在它执行前前提条件能够满足。特别是，关于重复代码和数据的减少量有时候会导致在源和目标应用上的不一致：目标应用上的可观察行为能与他的源应用行为有所不同。然而，大多数之前的关于转换问题的调查确实没有在具体的当转换手段无法应用时的环境进行过。