WebSocket技术在Web树组件的应用研究

1.JIANG Feng-yan，2.DUAN Hui-chuan

1．中国济南山东师范大学，信息科学与工程学院，山东省分布式计算机软件新技术重点实验室。[联系邮箱jiangfengyan1209@126.com](mailto:联系邮箱jiangfengyan1209@126.com)。

2．中国济南山东师范大学基础教育部。[联系邮箱mailsdnu@126.com](mailto:联系邮箱mailsdnu@126.com)。

摘 要—在信息系统的发展中，Web树组件是一种重要的信息展示方式，它提供了方便直观的信息查询，这种信息查询已经成为信息系统的本质特征之一。目前，Web树组件在通信效率、显示和更新率等方面还存在一系列问题。本文采用MyEclipse和MySQL作为开发工具来实现基于WebSocket的Web树组件的设计，这些组件可以显示和动态更新树的结构。理论分析和实验结果表明，改进后的方案提供了更好的用户体验，与传统的基于Ajax技术的长轮询相比，有效降低了数据传输量、通信开销等。

关键词—WebSocket；网站树组件；AJAX；长轮询

1. 引言

树形结构是一种非常流行的显示信息的方式，它被广泛应用于Web应用中，具有直观方便的优点，尤其是在表达目录结构、层次等方面。目前，大多数Web树组件是基于AJAX技术的，Ajax已成功提供无需重新加载页面的动态内容，但是，使用资源密集型的XMLHttp Request定期查询服务器，每次更新时都需要创建一个新的套接字连接和一个新的HTTP消息。Comet 技术实现双向通信是解决上述问题的一种方法。然而，当数据被接收或超时终止连接时，浏览器将等待下一次更新，这可能会导致更多的通信开销，并且Comet技术因为要维护维护两个连接需要消耗服务器的带宽和资源。WebSocket是一个Web技术，它通过一个在Web上的单一的套接字定义了一个全双工的通信通道。它只建立一个TCP套接字连接后的第一个请求连接，这样可以节省服务器的资源，网络带宽，并实现实时通信，这是一个用来解决上述问题的更好的方式。基于WebSocket技术的Web树组件在实时性、通信开销和时间延迟方面取得了巨大的进步。

1. WEBSOCKET

WebSocket采用单插座推拉信息来实现浏览器和服务器之间的全双工通信，这不仅可以避免Comet的连接和可移植性问题，而且能够实现比Ajax长轮询效率更高。目前，HTML5的WebSocket是促进网络的全双工实时通信的主要机制。

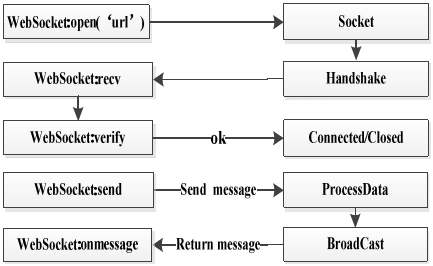


图1.websocket连接建立模型

图1是一个WebSocket连接建立模型，WebSocket协议本质上是一个TCP。为了建立WebSocket连接，客户端和服务器需要在他们初次握手时从HTTP协议升级到WebSocket协议，一旦建立，WebSocket数据帧在客户端和服务器之间的全双工模式中可以来回发送，并且这种连接会一直存在，直到客户端或服务器端主动关闭连接。WebSocket API定义了四个主要的回调方法：onopen, onmessage, onclose和onerror 用于处理在与websoceket连接期间的事件触发。这是非常适合实时，事件驱动的Web应用程序。这是非常适合实时，事件驱动的Web应用程序。

1. 实现树组件的技术比较

网络树的基本操作包括扩展节点，折叠节点，添加节点，修改节点，删除节点，基于Ajax的Web树组件必须在树的基本操作中向服务器发送请求，频繁的HTTP请求会导致不必要的流量浪费，降低网络通信的效率和显示、更新Web树结构的速度。同时，Ajax技术通过在指定的时间间隔向服务器发送请求来更新树结构，当大型客户机基于预定义的时间间隔访问服务器来进行更新时将增加服务器的负载。与基于Ajax技术的解决方案相比，基于Comet技术的Web树组件在一定程度上可以减少网络流量和服务器负载，但是这种方式也存在很大的局限性。首先，如果一个事件发生时，服务器将数据发送到客户端，客户端重新连接，它可以增加套接字开闭的操作数。其次，数据更新可能会有一点延迟，它在客户端完成一个更新后需要时间来重新建立与服务器的连接，在多人合作的情况下，这可能会破坏数据的一致性；而且Comet也会浪费服务器资源来管理连接，从而直接影响用户的响应性。WebSocket技术提供了一个更好的解决办法来显示网络的树结构，大大降低了服务器的负载和传输的数据量，达到网站树组件的有效沟通。

1. 基于WebSocket技术的Web树组件的实现

采用Jetty方案中的WebSocket服务器，相关的WebSocket类引用包命名jetty-websocket-7.4.5.v20110725.jar，Jetty是一个纯java Web容器，本文的组件也可以在Jetty直接部署。使用MyEclipseh和MySQL作为开发工具来实现基于WebSocket技术的Web树组件。

1. 改进类图的整体结构

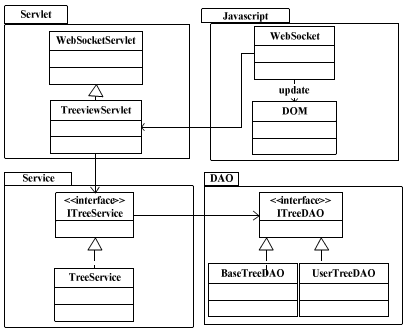


图2.组件类图的整体结构

服务器端构件的设计与实现是基于J2EE体系结构的。服务器端使用树节点对象树视图元素存储树节点信息，其中包含节点号、父节点、节点名、节点描述、节点属性集等，并作为DAO与服务类之间的参数传递。

图2是一个组件类图的UML整体结构，其中由于空间原因省略了类的所有方法和属性。该组件由四个包组成，JavaScript包是在JavaScript语言内部支持的API函数，使用API组件客户端可以实现Ajax读取、写入、修改当前活动文档的DOM节点，并对事件做出响应；在servlet包下的WebSocketServlet类接收所有的客户请求，他们根据不同的请求，调用不同的服务，通过把请求对象和响应对象作为参数，并最终返回处理结果到客户端。Service包定义了实现Web树组件服务逻辑的API，每个服务类通过使用DAO中的依赖类来访问数据库并返回处理结果；DAO是为某些类型的数据库或持久机制提供抽象接口的对象，它在不暴露数据库细节的前提下提供了一些具体操作。当基础数据源更改时，DAO向客户端提供的接口并不会改变，该模式允许DAO在不影响客户端或业务组件的情况下适应不同的存储方案。本质上，DAO是充当组件和数据源之间的适配器。

除了图2中的类，服务器端也使用JSP树标签技术向客户端输出HTML和JavaScript代码。JSP树标签是一个可定制的标签，可以动态生成树结构，标签的参数可以用来设置Web树的类型，包括是否支持右键菜单，以及节点是否有复选框，这样可以简单的配置客户端的JavaScript。在详细的客户端的实现上通过使用Dojo组件库来显示树形结构和操作菜单。Dojo是一个优秀的开源的JavaScript工具的集合，它的目的是提供一个更好的，更容易的和更快的工具集合，以方便Web开发。它提供了丰富的页面显示，可以支持动态网页的开发；Dojo还支持事件处理机制和WebSocket技术等。

1. 操作程序及组件核心代码

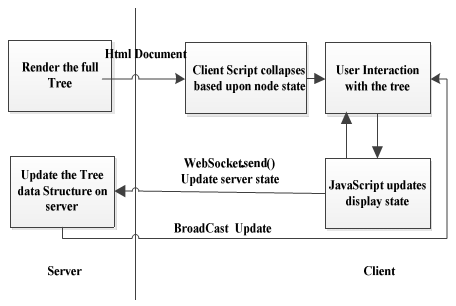


图3.基于WebSocket的操作程序

图3是基于WebSocket的Web树组件的操作程序。Web树组件直接跨越了在服务器端的树结构，并且实现了客户端的显示，如果用户操作一个树节点，客户端会调用WebSocket中“send”的方法来传递消息到服务器，然后将其推送到所有连接里的用户。

1. 结论

基于WebSocket的Web树组件利用WebSocket接口实现服务器和客户机之间的实时数据的传输。相比于Ajax轮询和Comet的解决方案，通过WebSocket有效地降低了服务器的负载，减少了节点的数据传输，降低了通信成本，简化了实现的复杂度等等。它极大地提高了Web服务的能力，并为用户带来了更好的操作体验。