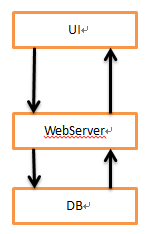
## 基于comet服务器推技术思路的Pushlet技术实现

目前许多有关管控的平台都牵涉自信息推送，即系统能够将后台发生数据变化等实时传送到客户端且无须客户端不停的刷新和发送请求。

### 传统推送

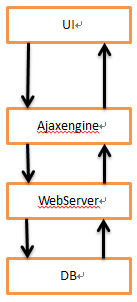
（1）基于webServer推送

传统推送方式主要是发送请求、解析服务器返回的消息并以不同的风格显示。



（2）基于Ajax推送

Ajax主要是基于浏览器发送异步请求，提高用户操作的响应性。



以上两种推送方式都是让用户在过时的信息下进行操作，所以实时的效果还是需要其他的技术来做实现。

这时候基于HTTP长连接、无须在浏览器安装插件的服务器推技术就得到了发展，这种技术成为Comet。

### Comet技术

Comet技术的实现方式有两种：基于 AJAX 的长轮询（long-polling）方式(long-polling)和流方式(streaming)。

（1）基于 AJAX 的长轮询方式

长轮询:HTTP的连接保持，服务器端会阻塞请求,直到服务器端有一个事件触发或者到达超时。客户端在收到响应后再次发出请求,重新建立连接。通过这种方式,服务器可以在数据可用的任何时候将数据“推”到客户端。因为这种方案基于AJAX,请求异步发出，无须安装插件，IE、Mozilla FireFox都支持。

使用 AJAX 实现“服务器推”与传统的 AJAX 应用不同之处在于：

1. 服务器端会阻塞请求直到有数据传递或超时才返回。

2.客户端 JavaScript 响应处理函数会在处理完服务器返回的信息后，再次发出请求，重新建立连接。

3. 当客户端处理接收的数据、重新建立连接时，服务器端可能有新的数据到达；这些信息会被服务器端保存直到客户端重新建立连接，客户端会一次把当前服务器端所有的信息取回。

（2）流方式

流方式：在流方式中，服务器推数据返回客户端，但不关闭连接，连接始终保持，直到超时，超时后通知客户端重新建立连接，并关闭原来的连接。

##### Pushlet技术

Pushlet就是comet思路的一种技术实现。并且它支持了两种comet的实现方式。在pushlet里面，它提供了基于 AJAX 的 JavaScript 库文件用于实现长轮询方式的“服务器推”；还提供了基于 iframe 的 JavaScript 库文件用于实现流方式的“服务器推”。这两种方式正好是comet的两种实现方式。

从设计模式的角度来看，pushlet使用的是观察者模式。客户端发送请求，订阅感兴趣的事件；服务器端为每个客户端分配一个会话 ID 作为标记，事件源会把新产生的事件以多播的方式发送到订阅者的事件队列里。

**备注：**长轮询不是一个真正的push技术，传统的轮询是前端ajax轮询，每隔一段时间发一个请求，服务器响应后马上关掉连接，但是这种方式明显有很大的开销，所以才有了长轮询，就是响应时间变长了，浏览器（客户端）发送一个请求，服务器hold住连接（就是循环加睡觉，可以到网上找找简单的实现代码），等有消息的时候才返回，当然浏览器的这个连接在这个过程中可以阻塞也可以异步非阻塞，ajax是异步的，等等，你说了这么多我还是不知道为什么长轮询可以当push技术用，和传统的有什么区别，这里说一下，长轮询技术要求服务器一旦发送了响应，客户端必须马上再发一个请求，这就变成了服务器是主动方，所以才说是一种伪push技术。