Web Sockets

- The problem
- WebSockets
- Servlet for a socket
- Annotations for WebSockets
- Examples

The problem

Http is based on requests.

If user needs something he must make a request.

We can’t notify users from the server.

Solutions

- Request a page every N seconds
- Request a part of a page (ajax) every N seconds
- Long polling
- WebSockets

WebSockets

WebSocket is a computer communications protocol, providing full-duplex communication channels over a single TCP connection.

WebSocket is designed to be implemented in web browsers and web servers, but it can be used by any client or server application.

The WebSocket Protocol is an independent TCP-based protocol.

Its only relationship to HTTP is that its handshake is interpreted by HTTP servers as an Upgrade request.

The WebSocket protocol enables interaction between a browser and a web server with lower overheads, facilitating real-time data transfer from and to the server.

org.eclipse.jetty.websocket.servlet.WebSocketServlet

Abstract Servlet used to bridge the Servlet API to the WebSocket API.

```
/**
 * This class represents a servlet starting a webSocket application
 */
public class WebSocketChatServlet extends WebSocketServlet {
    private final static int LOGOUT_TIME = 10 * 60 * 1000;
    @Override
    public void configure(WebSocketServletFactory factory) {
        factory.getPolicy().setIdleTimeout(LOGOUT_TIME);
        factory.setCreator(new ChatWebSocketCreator());
    }
}
```

```
public interface WebSocketCreator
{
    Object createWebSocket(ServletUpgradeRequest req, ServletUpgradeResponse resp);
}
```

Annotations for WebSockets

- WebSocket
- OnWebSocketConnect
- OnWebSocketMessage
- OnWebSocketError
- OnWebSocketClose