

WebSocket-Workshop - WS²

LUIGI LO IACONO , HOAI VIET NGUYEN

Fachhochschule Köln

luigi.lo_iacono@fh-koeln.de , viet.nguyen@fh-koeln.de

May 23, 2013

Abstract

Webanwendungen drängen mehr und mehr dazu echtzeitfähig zu werden. Das aktuelle HTTP schränkt jedoch diese Fähigkeit ein. Echtzeitkommunikationen sind nur mit Umwegen und mit einem Effizienzverlust zu realisieren. Dieser Workshop soll eine Einführung in WebSockets anbieten, worauf Entwickler aufbauen können und ihre eigenen Echtzeitanwendungen einfacher und effizienter realisieren können. Dabei soll nicht nur auf die theoretischen Grundlagen eingegangen werden, sondern auch auf Praxis bezogene Anwendungsgebiete von WebSockets. Der Workshop soll den Teilnehmern die vielfältigen Einsatzmöglichkeiten dieser neuen Technologie, anhand von ausgesuchten Beispielanwendungen, zeigen. Es wird nicht nur auf konventionelle Anwendungen, wie Live-Ticker oder Chats, eingegangen, sondern auch auf Anwendungsgebiete, die durch eine Echtzeit-Technologie, wie WebSockets, profitieren und sogar bereichert werden können. Die Teilnehmer können lernen, wie man z.B. eine Fernsteuerung von Webanwendung mittels WebSockets realisiert. Oder auch Usability-Tests durch WebSockets vereinfacht und visuell ansprechender gestalten kann. Ausserdem soll auf die Sicherheit von WebSockets eingegangen werden. Den Teilnehmern des Workshop sollen Möglichkeiten präsentiert werden, wie Sie ihre WebSocket-Verbindungen vor unerlaubten Zugriffen schützen können.

I. EINLEITUNG

DAS HTTP wurde ursprünglich nach dem Request/Response-Modell entwickelt. Daten konnten erst nach einer HTTP-Anfrage erhalten werden. Diese Eigenschaft des HTTP-Protokolls schränkt Anwendungsfelder ein, wo Server in der Lage sein müssen, unabhängig vom Client, Daten zu schicken. Diese Anwendungsfelder können Chats, Live-Ticker, Video- oder Audio-Streams sein. Um solche Anwendung zu realisieren, mussten bisher Plugins, wie Flash oder Java-Applets, in Betracht gezogen werden. Will man nicht auf solche Plugins zurückgreifen, musste man sich anders behelfen und auf Lösungsansätze der bisherigen Webstandards ausweichen. Diese Lösungsansätze beruhen auf stetigen und regelmäßigen Anfragen des Clients. Mit Polling und Long Polling war es möglich eine Bidirektionalität nachzuahmen. Beim Polling-

Verfahren schickt der Browser in einem bestimmten Intervall (z.B. alle 2 Sekunden) Anfragen an den Server. Der Server antwortet entsprechend in diesem Intervall, ob neue Daten vorhanden sind. Das Long Polling-Verfahren ist etwas geschickter, indem es die Intervalle länger wählt. Hier antwortet der Server nicht direkt auf jede Anfrage, sondern erst wenn Informationen vorhanden sind. Der Browser stellt somit eine Anfrage und wartet eine bestimmte Zeit (z.B. 20 Sekunden) bis der Server antwortet. Antwortet der Server nicht, schliesst der Client die Verbindung und baut diese nach dem gleichen Muster wieder auf. Antwort der Server aber innerhalb dieser bestimmten Wartezeit, nimmt der Client die neuen Daten an und schickt direkt danach eine erneute Anfrage und warten wieder. Polling und Long Polling verbrauchen sehr viele Ressourcen, da für jede Anfrage ein sehr großer Overhead entsteht. Ausserdem muss

die TCP-Verbindung immer erneut aufgebaut werden. Dies kann bei einer Anwendung mit sehr vielen Benutzern, den Server in die Knie zwingen. WebSockets können diese Probleme beheben, indem Sie eine echte bidirektionale Verbindung vom Client zum Server ermöglichen [1]. Dadurch entstehen neue Aussichten, Anwendungen zu entwickeln die sich aus dem Request/Response-Gefängnis befreien können. Dieser Workshop soll den Teilnehmern diese Aussichten und Möglichkeiten zeigen, wie Sie mit WebSockets innovative Echtzeitanwendungen entwickeln können.

II. ZIELGRUPPE UND VORRAUSSETZUNG

Der Workshop richtet sich an alle Entwickler, die sich für das Thema WebSockets und Echtzeitanwendungen interessieren. Die Teilnehmer sollten möglichst ihren eigenen Rechner zum Workshop mitnehmen. Grundlegende Kenntnisse in JavaScript und Java wären von großen Vorteil. Es reichen aber auch gute Kenntnisse in anderen Programmiersprachen. Der Kurs fokussiert sich hauptsächlich auf die Sprachen JavaScript und Java. Die verwendeten Frameworks sind Node.js, vert.x und das Play Framework. Kenntnisse in diesen Frameworks wären von Vorteil, sind aber keine Voraussetzung. Frontend- und Backend-Entwickler aus folgenden Bereichen können z.B. an diesem Workshop teilnehmen:

- Mobile Devices
- Webentwicklung in HTML5
- Webservices
- IT-Sicherheit

III. INHALT DES WORKSHOPS

Der Workshop startet mit einer kleinen Einführung in WebSockets. Der Workshop soll möglichst interaktiv gestaltet werden. D.h. es gibt möglichst wenig Folien. Es wird z.B. direkt im Browser ein WebSocket geöffnet und anhand der Chrome Developer Tools und Wireshark, das Protokoll und der Verbindungsaufbau analysiert. Anschliessend soll den Teilnehmern die Implementierung

eines einfachen WebSocket-Echo-Servers per Live-Coding gezeigt werden. Durch diesen Echo-Server sollen sowohl Binär- als auch Textnachrichten versendet werden können. Nach dem Live-Coding sollen die Teilnehmer auf Grundlage des Echo-Server, die Anwendung weiter zu einem Chat oder Liveticker ausbauen. Die Workshop-Leiter werden dann für Fragen und Hilfestellungen zu Verfügung stehen. Danach sollen die Anwendungsgebiete durch WebSockets, anhand von Demos bzw. Beispielanwendungen, vorgestellt werden. Daraufhin wird die Implementierung eines dieser Anwendung von Grund auf präsentiert. Welche dieser Demos implementiert wird, können die Teilnehmer selber bestimmen. Die verwendeten Programmiersprachen für die Implementierung sollen, wie oben schon erwähnt, JavaScript und Java sein. Die Teilnehmer können dann selbst entscheiden, welcher dieser beiden Sprachen sie lieber bevorzugen. Die Implementierung der Beispielanwendungen soll auch möglichst interaktiv sein. Es wird je nachdem nur ein paar Folien gezeigt, die die Architektur der ausgewählten Anwendung darstellt. Anschliessend soll direkt an einem Editor live programmiert werden. Im Anschluss sollen die Teilnehmer einer der anderen Beispielanwendungen selber programmieren. Die Workshop-Leiter stehen wieder für Fragen und Hilfestellungen den Teilnehmern zur Seite. Zusätzlich wird auf das Thema Sicherheit in Bezug auf WebSockets eingegangen. Es soll den Teilnehmern gezeigt werden, wie Sie ihre WebSockets vor unerlaubten Zugriffen schützen können. Dabei soll auf die Sicherheitsmöglichkeiten, der verschiedenen WebSocket-Frameworks eingegangen werden. Welche Frameworks ermöglichen den Entwicklern eine zuverlässige Zugriffskontrolle, welche nicht? In welchen Frameworks gibt es noch Schwächen. Wie kann man trotz diese Schwachpunkte eine Zugriffskontrolle realisieren. Dies soll wieder anhand einer Beispielimplementierung gezeigt werden. Der Fokus liegt hier allerdings auf JavaScript bzw. Node.js, da in dieser Technologie, Frameworks

vorliegen, die eine Zugriffskontrolle mit WebSockets, am einfachsten ermöglichen.

Der Workshop soll mit einer Diskussionsrunde ausklingen. Es können Fragen und Anregungen gestellt werden, sowie Verbesserungswünsche der Beispielanwendungen. Unklarheiten können hier auch geklärt werden. WebSocket-erfahrende Teilnehmer und die Workshop-Leiter können ihre Erfahrungen über WebSockets austauschen. Der Workshop gliedert sich in folgenden Themen:

- Einführung in WebSockets
- Analyse von WebSockets mit Wireshark, Fiddler, Chrome Developer Tools
- Implementierung von verschiedenen

Echo-Servern (binär und Text)

- Anwendungsgebiete
- Implementierung von Beispielanwendungen
- Sicherheit von WebSockets
- Diskussionsrunde

REFERENCES

- [1] **Luigi Lo Iacono, Hoai Viet Nguyen**
Gegenverkehr - Birektionale Client-Server-Verbindungen mit WebSockets. IX - Magazin für professionelle Informationstechnik *Ausgabe November 2012*