John and Jonas are founders of Kaazing, provider of the open source HTML 5 Websocket gateway.

Both dudes are frequent writers for magazines such as JDJ, JavaMagazine, and AjaxWorld.

John and Jonas speak frequently at conferences Worldwide on HTML 5 Communication.

Authors of Pro JSF and Ajax: Building Rich Internet Components.
Overall Presentation Goal

Learn how to architect and build full-duplex Web applications using HTML 5 communication
If we were not restricted by the traditional limitations of HTTP, what type of Web applications would we build?
Agenda

- HTML 5 Overview
- Networking Review
- Today’s Architecture
- HTML 5 Server-Sent Events
- HTML 5 WebSockets
- Kaazing Gateway
HTML 5 Overview

- Next generation application platform
  - Communication (sockets, cross-site)
  - Graphics (2D)
  - Drag ‘n’ drop
  - Storage (transient, persistent)
  - Offline
  - Compatibility
Networking Review

- Desktop Networking
  - Full-duplex bidirectional TCP sockets
  - Access any server on the network

- Browser Networking
  - Half-duplex HTTP request-response
  - HTTP polling, long polling, streaming
  - Same-origin HTTP requests
Today’s Architecture

- **Browser** (Half Duplex)
- **Servlet**
- **Java EE Container**
  - **RMI**
  - **EJB**
  - **JavaMail**
  - **JMS**
- **Application Transport Logic**
- **Database**
- **IMAP Server**
- **IM Server**
- **Stock Trading Client**
- **Stock Trading Feed**

Transport Protocols:
- **HTTP** (Half Duplex)
- **RMI**
- **JDBC** (TCP Full Duplex)
- **IMAP** (TCP Full Duplex)
- **JABBER** (TCP Full Duplex)
- **JMS** (TCP Full Duplex)
- **Custom** (TCP Full Duplex)
What’s Missing?

- No true bi-directional communication
  - Comet/Reverse Ajax are non-standard one-way push
- No guaranteed message delivery
- Complex middle-tier architecture
  - Adds unnecessary latency
HTML 5 Communication

- WebSocket
  - Proxy-friendly text socket for your browser
- Server-Sent Events
  - Standardized HTTP streaming (downstream)
- Cross-Site XMLHttpRequest
  - Secure cross-site remote communication
- postMessage
  - Secure inter-iframe communication
HTML 5 WebSockets

- Provides Full-Duplex Text Socket
  - Operates over a single socket
- Traverses firewalls and routers seamlessly
- Leverages cross-site access control
- Integrates with:
  - Cookie-based authentication
  - Existing HTTP load balancers
Full-duplex Architecture

- TCP over HTTP (Full Duplex)
- RMI - TCP (Full Duplex)
- Java EE - EJB - TCP (Full Duplex)
- JDBC - TCP (Full Duplex)
- IMAP - TCP (Full Duplex)
- Jabber - TCP (Full Duplex)
- RMI - TCP (Full Duplex)
- JMS
- Database
- IMAP Server
- IM Server
- Custom - TCP (Full Duplex)
- Stock Trading Feed
ws://www.kaazing.org/text

wss://www.kaazing.org/encrypted-text
Creating a WebSocket instance:

```javascript
var myWebSocket = new WebSocket
("ws://www.websocket.org");
```
Associating listeners:

```javascript
myWebSocket.onopen = function(evt) {
    alert(“Connection open ...”);
};

myWebSocket.onmessage = function(evt) {
    alert( “Received Message: ” + evt.data);
};

myWebSocket.onclose = function(evt) {
    alert(“Connection closed.”);
};
```
Sending messages:

```javascript
myWebSocket.postMessage("Hello WebSocket!");
myWebSocket.postMessage("Goodbye Comet!");
myWebSocket.disconnect();
```
HTML 5 WebSocket Handshake

GET /text HTTP/1.1\r\nUpgrade: WebSocket\r\nConnection: Upgrade\r\nHost: www.kaazing.org\r\n...\r\n
HTTP/1.1 101 WebSocket Protocol Handshake\r\nUpgrade: WebSocket\r\nConnection: Upgrade\r\n...\r\n
HTML 5 WebSocket Frames

- Frames can be sent full-duplex
  - Either direction at any time

- Text Frames use terminator
  - \x80Hello, WebSocket\x0ff

- Binary Frames use length prefix
  - \x00\0x10Hello, WebSocket

- Text and binary frames on same WebSocket
DEMO

Java Messaging Service
Kaazing WebSocket Support

- Protocols
  - ByteSocket
  - WebSocket
  - Server-Sent Events
  - Cross-Site XHR
  - postMessage
  - XHR
  - IFrame
HTML 5 Server-Sent Events

- A continuous stream of data sent from server to browser
  - Standardizes Comet
  - JavaScript API
  - Wire protocol

Browser Support
- Patch under review for Firefox trunk – Bug 338583
- Opera already done
HTML 5 Server-Sent Events

- Introduced a new HTML DOM Element

  `<eventsource src="http://www.kaazing.org/sse"
  onmessage="alert(event.data)" />

- HTML DOM API

  `var es =
  document.createElement("eventsource");
  es.addEventListener("message",
    function(event) { alert(event.data); },
    false);
  es.addEventSource("http://www.kaazing.org/sse");`
Guaranteed message delivery

GET /sse HTTP/1.1\r\nHost: www.kaazing.org\r\nLast-Event-ID: 9\r\n...

200 OK HTTP/1.1\r\n...

:comment\n**id**: 10\n**data**: Hello, Server-Sent Events

DEMO

HTML 5 Server-Sent Events
Cross-Site XMLHttpRequest

- W3C Technical Report
  - Access Control for Cross-Site Requests
  - Published Sept 12, 2008
  - http://www.w3.org/TR/access-control/

- Browser Support
  - Firefox 3.1-beta
  - IE8 XDomainRequest (similar)
  - Opera, Safari, Chrome coming
Cross-Site XMLHttpRequest

GET / HTTP/1.1
Host: www.w3.org
Origin: http://www.kaazing.org
...

200 OK HTTP/1.1
Allow-Origin: http://www.kaazing.org
...

HTML 5 postMessage

- Send Strings Between HTML Documents
  - Documents may be served by different sites

- Standard API

  ```javascript
  targetWindow.postMessage(message, targetOrigin)
  window.onmessage = function(event) {
    alert(event.data);
  }
  ```

- Browser Support
  - IE 8, FF 3, Opera 9, WebKit nightlies
Kaazing WebSocket Support

Protocols

WebSocket

Server-Sent Events

Cross-Site XHR

postMessage

XHR

IFrame
Kaazing ByteSocket

- Provides binary socket abstraction
- Leverages text-based WebSocket
  - Encodes payload using base64
- Send and receive ByteBuffers
  - JavaScript has no byte or ByteArray type (yet)
- Kaazing Gateway converts base64
Kaazing Protocols

- Text or Binary
  - Stomp
  - XMPP
  - IRC
  - Telnet
  - IMAP
  - SMTP
  - Custom…
Kaazing Gateway Scalability

- Based on SEDA (Staged Event-Driven Architecture)
  - Leverages Java New I/O (NIO)
- Concurrency
  - Proportional to bandwidth not connections
- Latency
  - Socket integration, bytes-in, bytes-out
- Stateless
  - Minimal memory usage, balancing, failover
Kaazing Enterprise Gateway

Features

- Adobe Flex APIs
- Flash runtime detection
- EncryptedKeyring
- Single sign-on
- Protocol Validation
- Protocol Security Enhancements
- Management
Kaazing Stomp client and XMPP Client
Summary

- HTML 5 Communication is here
- WebSockets and SSE standardize Comet
- Avoid vendor lock-in, the standards shall set you free!
- Kaazing Community – [www.kaazing.org](http://www.kaazing.org)
- Kaazing Enterprise – [www.kaazing.com](http://www.kaazing.com)
Concluding statement

Next generation Web architecture has arrived!
Q&A
Thanks for your attention!

http://www.kaazing.com

http://www.kaazing.org

http://www.w3c.org