Real Time Programming with Arduinos using WebSockets

Justin Mclean
Class Software

Email: justin@classsoftware.com
Twitter: @justinmclean
Blog: http://blog.classsoftware.com

Who am I?

- Director of Class Software for almost 15 years
- Developing and creating web applications for 15 years
- Programming for 25 years
- Adobe certified developer and trainer in Flex and ColdFusion
- Adobe Community Professional
- Based in Sydney Australia

WebSockets

WebSockets

- Bidirectional real time communication over a single TCP/IP socket
- For browser/server communication
- Fast protocol has low overhead and connections kept open
- Binary and text support

Browser Support

- Multiple versions of the protocol (14!)
- Safari 5+ and Chrome 6+
- Disabled due to security concerns in current versions of Firefox 4 and Opera 11
- Firefox 6+ enabled but different
- Internet Explorer 9+ support via a plugin
- iOS (4.2+) but not Android (yet)
- Standards are such wonderful things

WebSockets API

- To create:
  ws =new WebSocket("ws://10.0.0.20/");
- Events:
  ws.onopen
  ws.onclose
  ws.onmessage
  Ws.onerror
- Send
  ws.send(“message”);

Fallbacks

- HTTP long polling (slow)
- Flash (oh the irony!)
- web_socket.js
- Socket.IO

Protocol

- HTTP
- Handshake
- Upgrade to WebSocket
- Data transfer and message framing

Request

- GET index.htm HTTP/1.1
  Upgrade: WebSocket
  Connection: Upgrade
  Host: example.com
  Origin: http://example.com
  Sec-WebSocket-Key1: 4 @1 46546xW%0l 1 5
  Sec-WebSocket-Key2: 12998 5 Y3 1 .P0

  ^n:ds[4U

Handshake

- Discard any none numbers in key1
- Divide by number of spaces in key1
- Discard any none numbers in key2
- Divide by number of spaces in key2
- Change to big endian
- Concatenate key1, key2 and key3 together
- Take an MD5 hash of the result

Response

- HTTP/1.1 101 WebSocket Protocol Handshake
  Upgrade: WebSocket
  Connection: Upgrade
  Sec-WebSocket-Origin: http://example.com
  Sec-WebSocket-Location: ws://example.com/demo
  Sec-WebSocket-Protocol: sample

  8jKS'y:G*Co,Wxa-

Data Framing

- Start 0x00
- Data
- End 0xFF

New Protocol Request

- GET /ws HTTP/1.1
  Host: pmx
  Upgrade: websocket
  Connection: Upgrade
  Sec-WebSocket-Version: 6
  Sec-WebSocket-Origin: http://pmx
  Sec-WebSocket-Extensions: deflate-stream
  Sec-WebSocket-Key:
  x3JJHMbDL1EzLkh9GBhXDw==

New Protocol Response

- HTTP/1.1 101 Switching Protocols
  Upgrade: websocket
  Connection: Upgrade
  Sec-WebSocket-Accept:
  HSmrc0sMlYUkAGmm5OPpG2HaGWk=

New Handshake

- Magic string added to Sec-WebSocketKey
- SHA1 hashed
- Base 64 encoded

New Data Framing

- Control + Length + Mask (optional) + Data
- More secure
- Less issues with proxies
- Greater overhead (2-7 bytes vs 2 bytes)

vs Other Technologies

- AJAX uses polling which is not real time
- Far less overhead than JSON
- Far less trouble and overhead than Comet
- But a way to go before it matches Flash/Flex

Issues

- Poor browser support require fallbacks
- Multiple versions of the protocol in play
- Protocol versions incompatible and are likely to change again
- Possible security issues
- Not suitable for all applications (but better than AJAX!)
- Resource considerations

Arduino and WebSockets

- Complex handshake
- Expensive key generation
- Limited connections
- Limited protocol - need to create your own message structure