



KAAZING

Kaazing Gateway

Open Source HTML 5
Web Socket Server

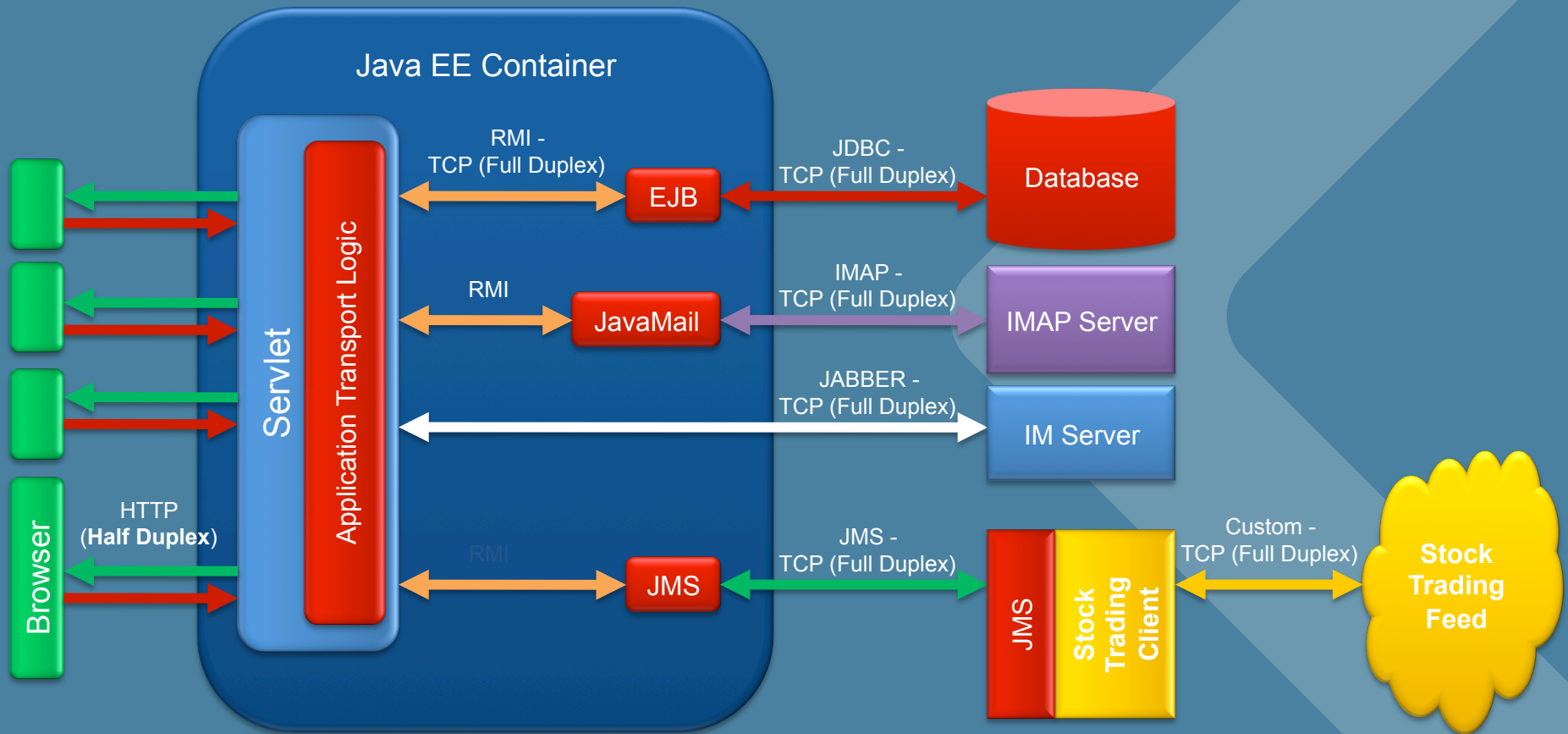
Agenda

- Networking Review
- HTML 5 Communication
- Kaazing Open Gateway
- Kaazing Enterprise Gateway
- Q & A

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

Half-Duplex Architecture



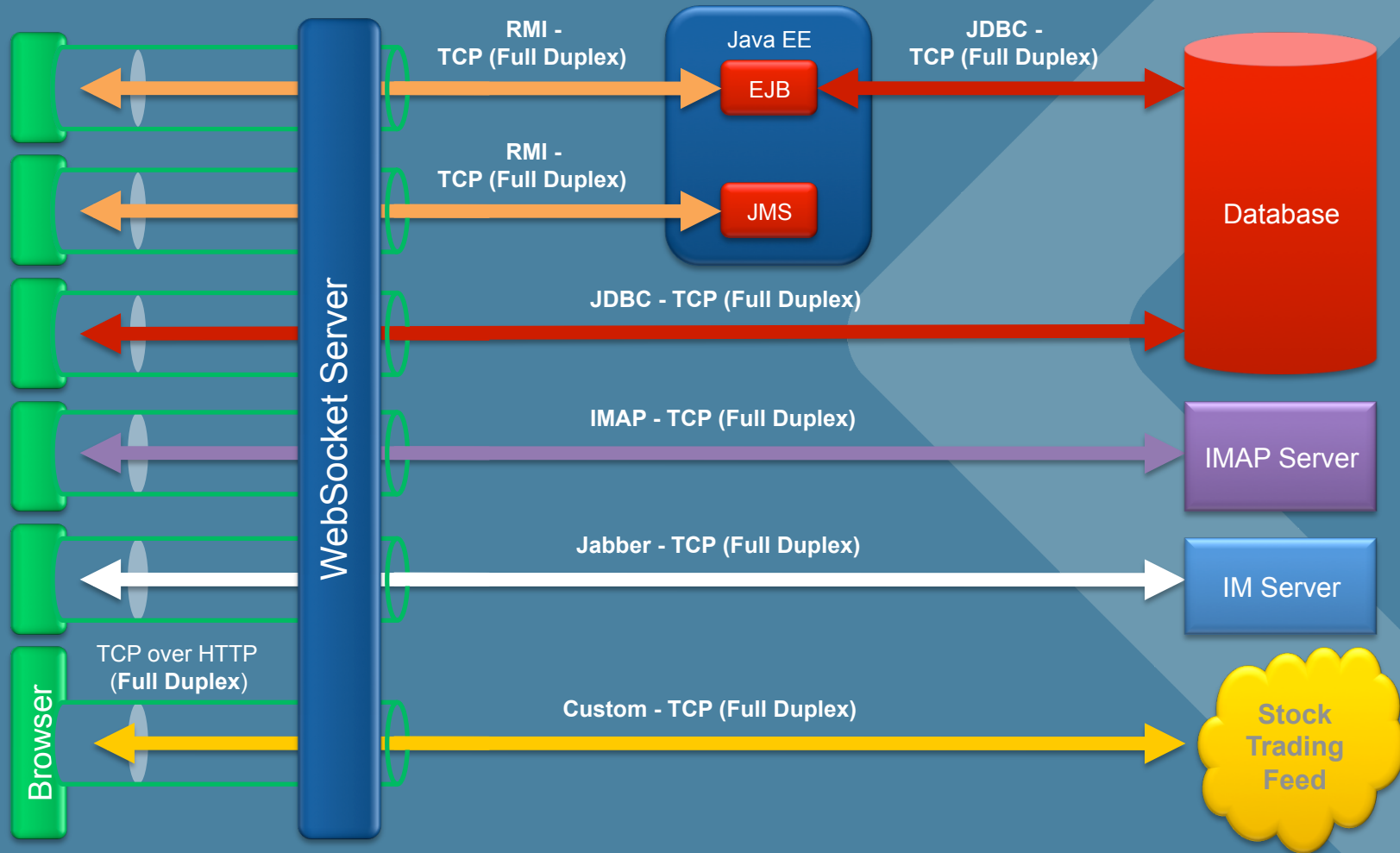
HTML 5 Overview

- Next generation application platform
 - Communication (sockets, cross-site)
 - Graphics (2D)
 - Drag 'n' drop
 - Storage (transient, persistent)
 - Offline
 - Compatibility
 - Scheduled for completion in 2022 (!)

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

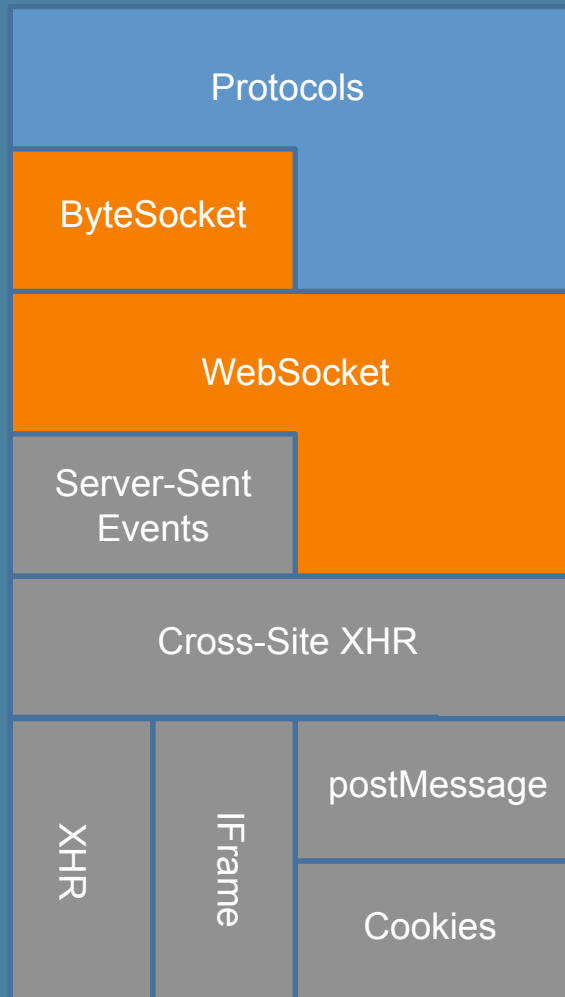
Full-Duplex Architecture



DEMO

Java Messaging Service

Kaazing Protocols



Kaazing ByteSocket

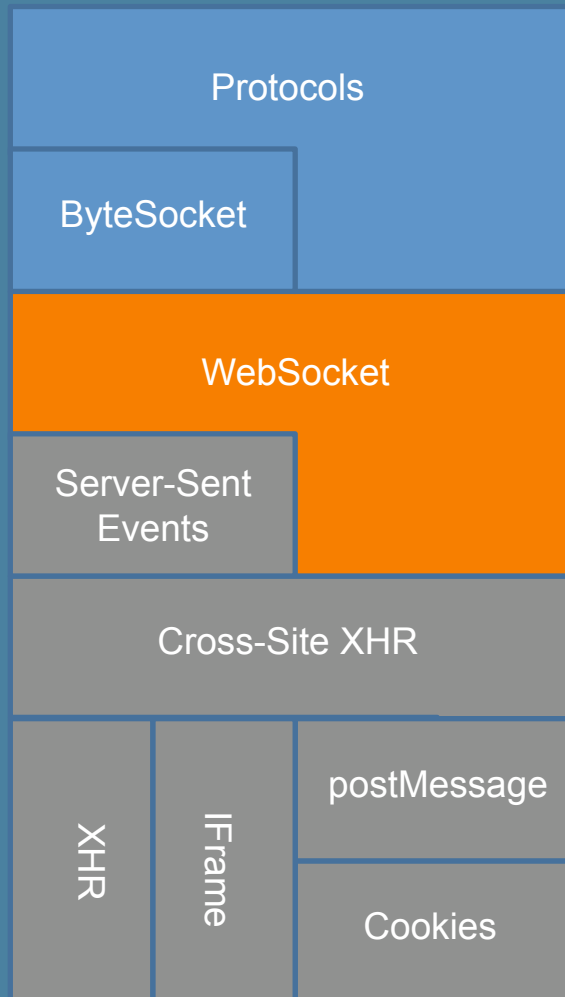
- Provides binary socket abstraction
- Leverages text-based WebSocket
 - Encodes payload using base64
- Send and receive ByteBuffers
 - JavaScript has no byte or ByteArray type (yet)
- Kaazing Gateway converts base64

Kaazing ByteSocket

```
var location = "ws://www.kaazing.org/binary";  
var socket = new ByteSocket(location);  
socket.onmessage =  
    function(event) { alert(event.data.getInt()); }
```

```
var buf = new ByteBuffer();  
buf.putString("Hello, world", Charset.UTF8);  
socket.postMessage(buf.flip());
```

Kaazing ByteSocket



HTML 5 WebSocket

- Provides Full-Duplex Text Socket
- Send and Receive Strings
- Enables Streaming to Server Too
- Browser Support
 - None (yet)

HTML 5 WebSocket Schemes

`ws://www.kaazing.org/text`

`wss://www.kaazing.org/encrypted-text`

HTML 5 WebSocket API

```
var location = "ws://www.kaazing.org/text";  
var socket = new WebSocket(location);  
  
socket.onopen =  
    function(event) {  
        socket.postMessage("Hello, WebSocket"); }  
  
socket.onmessage =  
    function(event) { alert(event.data); }  
  
socket.onclose =  
    function(event) { alert("closed"); }
```


HTML 5 WebSocket Handshake

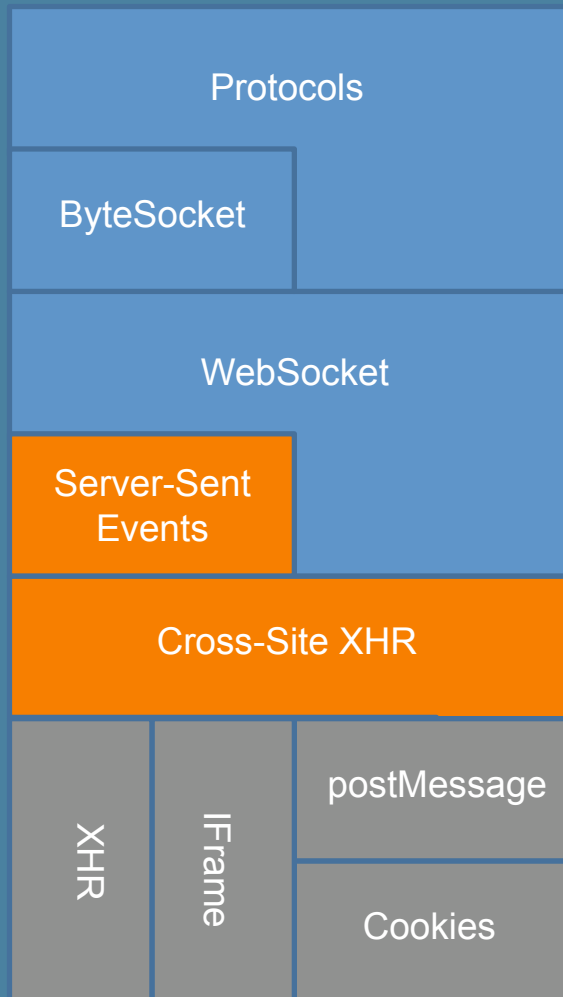
```
GET /text HTTP/1.1\r\nUpgrade: WebSocket\r\nConnection: Upgrade\r\nHost: www.kaazing.org\r\n...\r\n
```

```
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\xff`
- Binary Frames use length prefix
`\x00\x10Hello, WebSocket`
- Text and binary frames on same WebSocket

Kaazing WebSocket



HTML 5 Server-Sent Events

- Standardizes Comet
 - JavaScript API
 - Wire protocol
 - Encourages adoption by servers
- Browser Support
 - Patch under review for Firefox trunk

HTML 5 Server-Sent Events

- 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");
```

HTML 5 Server-Sent Events

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

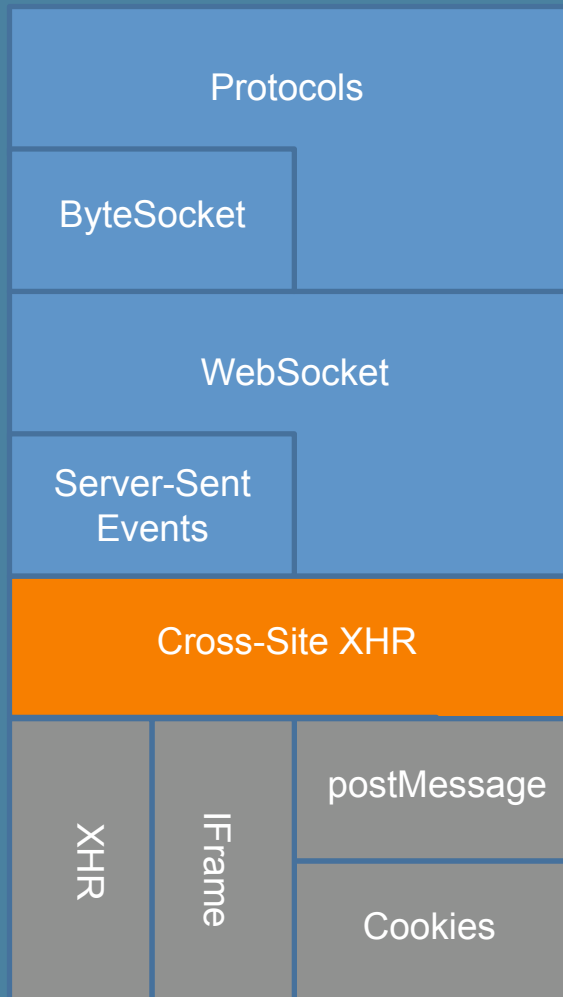
```
200 OK HTTP/1.1\r\n...\r\n
```

```
:comment\r\nid: 10\r\ndata: Hello, Server-Sent Events\r\n\r\n
```

DEMO

HTML 5 Server-Sent Events

Kaazing 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 XMLHttpRequest (similar)
 - Opera, Safari, Chrome coming

Cross-Site XMLHttpRequest

```
GET / HTTP/1.1\r\n
```

```
Host: www.w3.org\r\n
```

```
Origin: http://www.kaazing.org\r\n
```

```
...\r\n
```

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

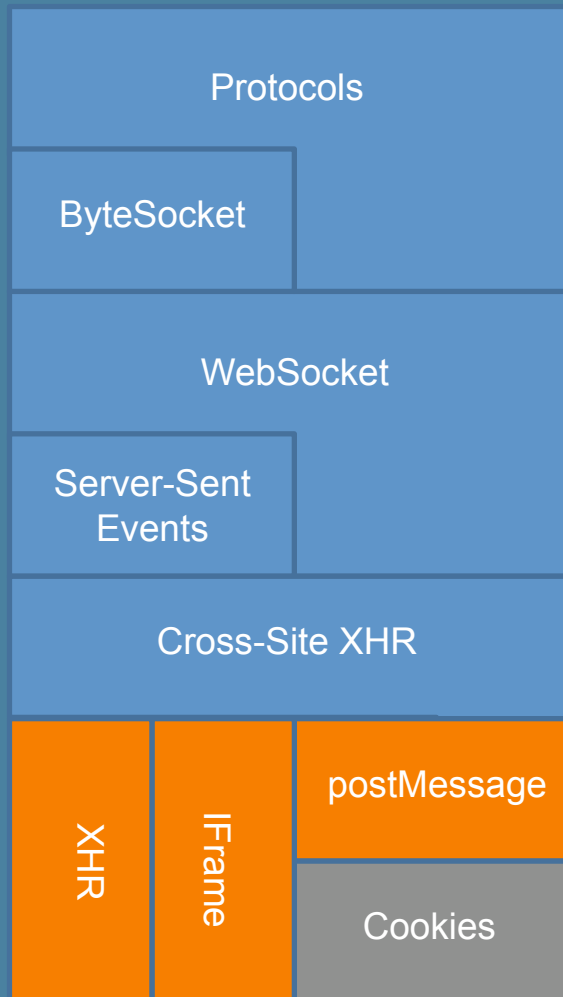
```
Allow-Origin: http://www.kaazing.org\r\n
```

```
...\r\n
```

DEMO

Cross-Site XMLHttpRequest

Kaazing Cross-Site XHR



HTML 5 postMessage

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

- Standard API

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

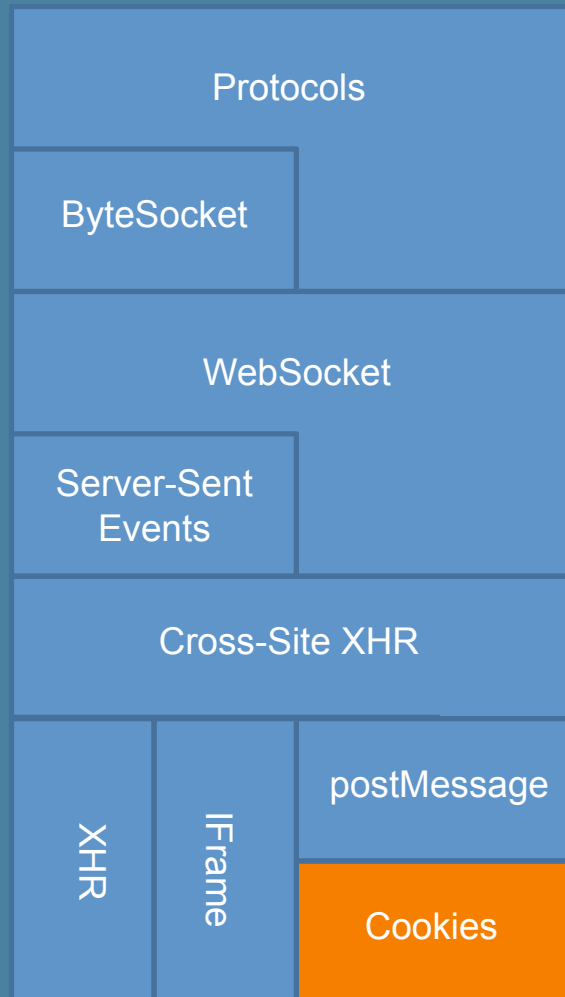
- Browser Support

- IE 8, FF 3, Opera 9, WebKit nightlies

DEMO

HTML 5 postMessage

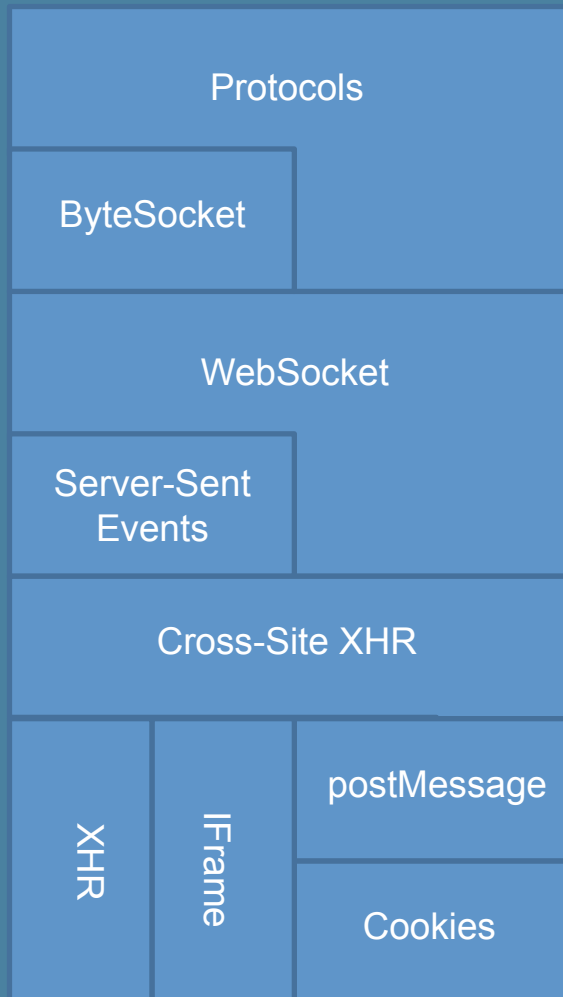
Kaazing postMessage



Kaazing postMessage

- HTML 5
 - `targetWindow.postMessage(message, targetOrigin)`
- Kaazing
 - `postMessage0(targetWindow, message, targetOrigin)`
 - Documents must be served by same or peer domains
 - `http://www.kaazing.org:8000/same-domain`
 - `https://www.kaazing.org:9000/secure-same -domain`
 - or -
 - `http://peer0.kaazing.org:8000/peer-domain`
 - `https://peer1.kaazing.org:9000/secure-peer-domain`

Kaazing Protocols Support



Kaazing Protocols

- Text or Binary
 - Stomp
 - XMPP
 - IRC
 - Telnet
 - IMAP
 - SMTP
 - Custom...

Kaazing Gateway Scalability

- 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
 - MS Silverlight APIs
 - Java APIs
 - Flash/Silverlight/Java runtime detection
 - Encrypted Keyring
 - Single sign-on
 - Protocol Validation
 - Protocol Security Enhancements
 - Management

DEMO

Kaazing Enterprise XMPP Client

Summary

- Kaazing Gateway
 - HTML 5 WebSocket (and more) today
 - Open source community
 - <http://www.kaazing.org>
 - Binary and text protocol support
- Kaazing Enterprise Gateway
 - 60-day free trial
 - <http://www.kaazing.com>
 - 24x7 support available