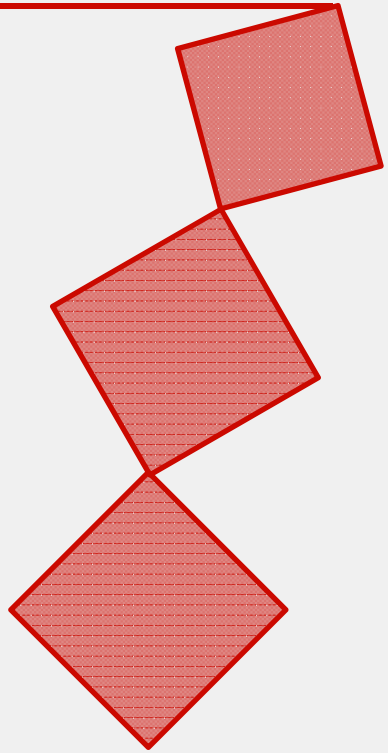
WebSocketを見てみよう

@Kuruma

概要版



whoami

- **Kuruma**
  - しがないOpera Browser使い
  - twitter.com/Kuruma
  - kuruman.org

概要

WebSocketとは

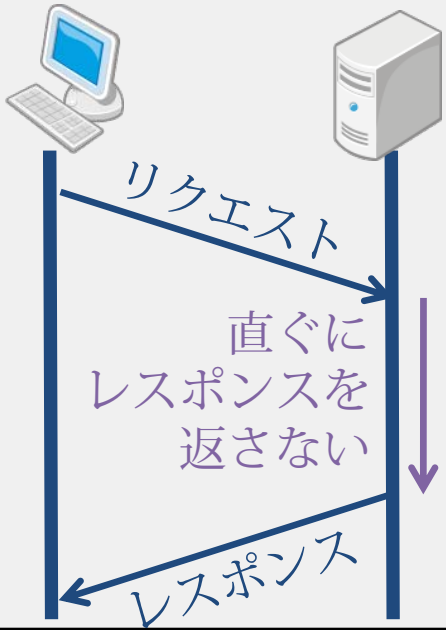
この技術の目的は、サーバとの双方向通信を必要とするブラウザベースのアプリケーションのために、**複数のHTTP接続を開くことなく双方向通信を実現する**ための仕組みを提供すること。

Abstract最後の一文を意訳

HTTPを用いた強引な双方向通信

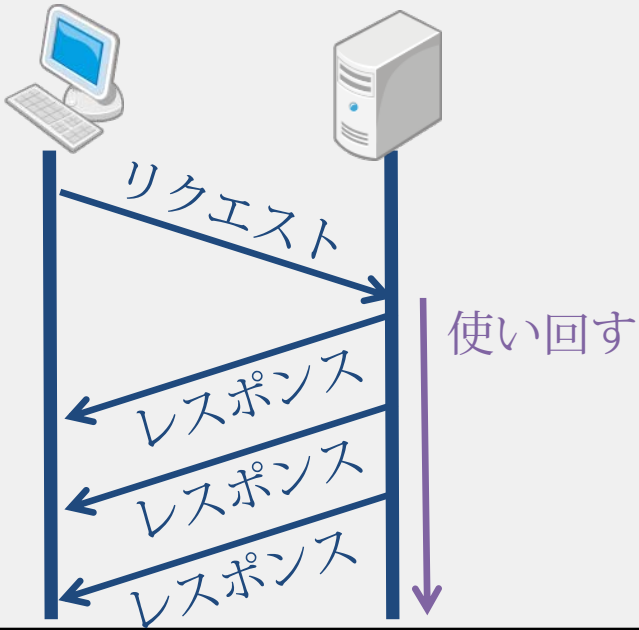
Long Polling

- サーバ契機の情報配信
  - 予めリクエストを送信
  - 即時性が低い
  - ヘッダを何度も送信



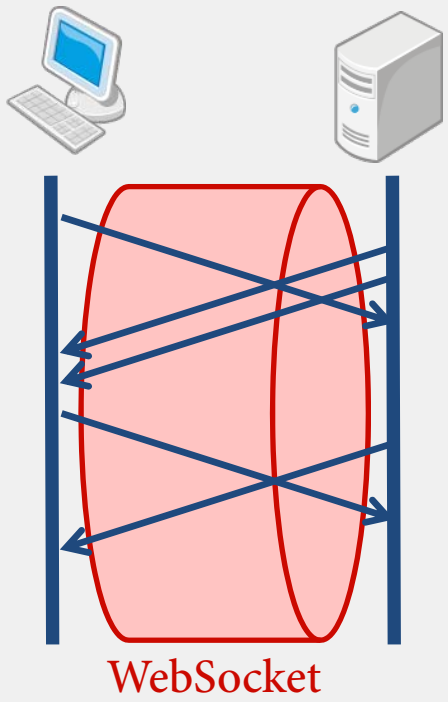
HTTP Streaming

- いつでも即時に情報配信
  - HTTPを終端しない
  - 片方向でセッションを占有



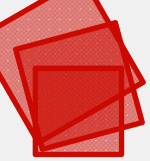
WebSocketによる双方向通信

- 全二重の双方向通信が可能
  - UTF-8の文字列、バイナリデータ
- 生のTCPソケットではない
  - TCPベースの独立したプロトコル



WebSocketの標準化

- WebSocket ∉ HTML5
- 標準化は現在進行形
  - The WebSocket protocol
    - IETF HyBi WG
    - http://tools.ietf.org/wg/hybi/draft-ietf-hybi-thewebsocketprotocol/
  - The WebSocket API
    - W3C WebApps WG
    - http://www.w3.org/TR/websockets/



既存のWebとの親和性

- 既存のWeb-proxyを通過できるように工夫
  - 最初のハンドシェイクをHTTPで行う
  - 80番ポートを使う (TLS使用時は443番)
- もちろんUser Agentの対応は必須

実装状況

WebSocketの変遷概要

	認証	バイナリ	圧縮	拡張	切断時 Status Code	データ フレーム
hixie-75	なし	MIME	不可	不可	なし	0x0 文字列 0xff
hixie-76 (hybi-00)	認証 もどき					
hybi-01		そのまま				バイナリ
hybi-02						
hybi-03			Deflate			
hybi-04	あり			あり		+mask
hybi-05				+negotiate		
hybi-06					あり	
hybi-07						

細かなフレーム形式の変更等は多々有り

hixie-**, hybi-**はそれぞれ
draft-hixie-thewebsocketprotocol-**, draft-ietf-hybi-thewebsocketprotocol-**を示す

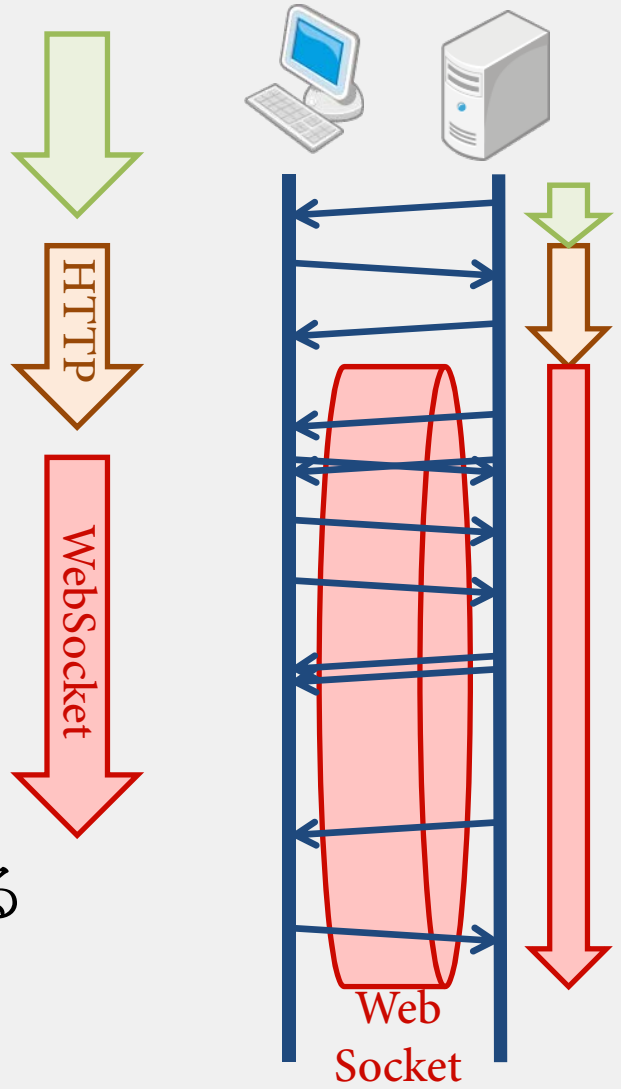
はじめの一歩

おことわり

- 今後大幅に変わる可能性もあります
- 2011-05-25時点でIETF/W3Cの公開しているドラフト文書に基づいた説明です

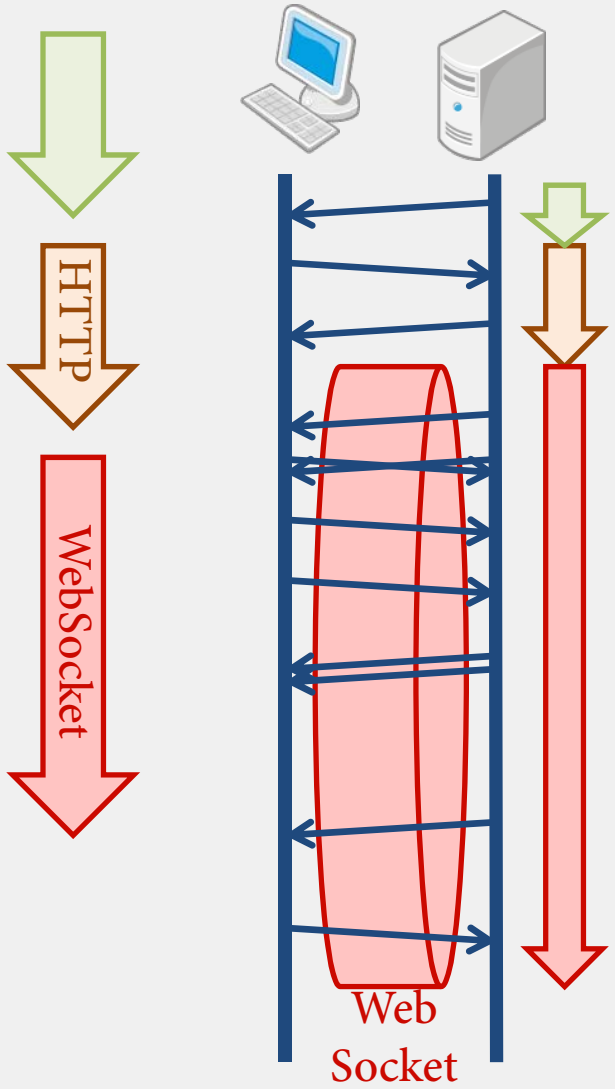
ブラウザから見た流れ

1. サーバAからScriptを取得
2. Scriptを実行
3. サーバAへWebSocket接続を要求
4. サーバAから接続が承認される
5. **サーバAとWebSocket通信**
6. WebSocket接続切断
   - サーバAから切断要求を受ける
   - 接続要求を送信する
   - その他なんらかの理由で切断される



もっと大まかな流れ

1. Scriptを実行
2. HTTPでハンドシェイク
3. WebSocketでデータ転送



1. Scriptを実行 (WebSocket API)

- WebSocket(url [, protocol])
- 4つのイベントハンドラ
  - onopen:　　接続確立時
  - onmessage:　メッセージ受信時
  - onerror:　　エラー発生時
  - onclose:　　接続終了時
- 2つのメソッド
  - boolean send(data)
  - void close()

2. HTTPでハンドシェイク

```
GET /chat HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: dGhlIHNhbXBsZSBub25jZQ==
Sec-WebSocket-Origin: http://example.com
Sec-WebSocket-Protocol: chat, superchat
Sec-WebSocket-Version: 7
```

```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: s3pPLMBiTxaQ9kYGzzhZRbK+xOo=
Sec-WebSocket-Protocol: chat
```

3. WebSocketでデータ転送