

# WebSockets

**Ariel Strong**

**DesertPy Meetup**

**April 26, 2017**

# Overview of WebSockets

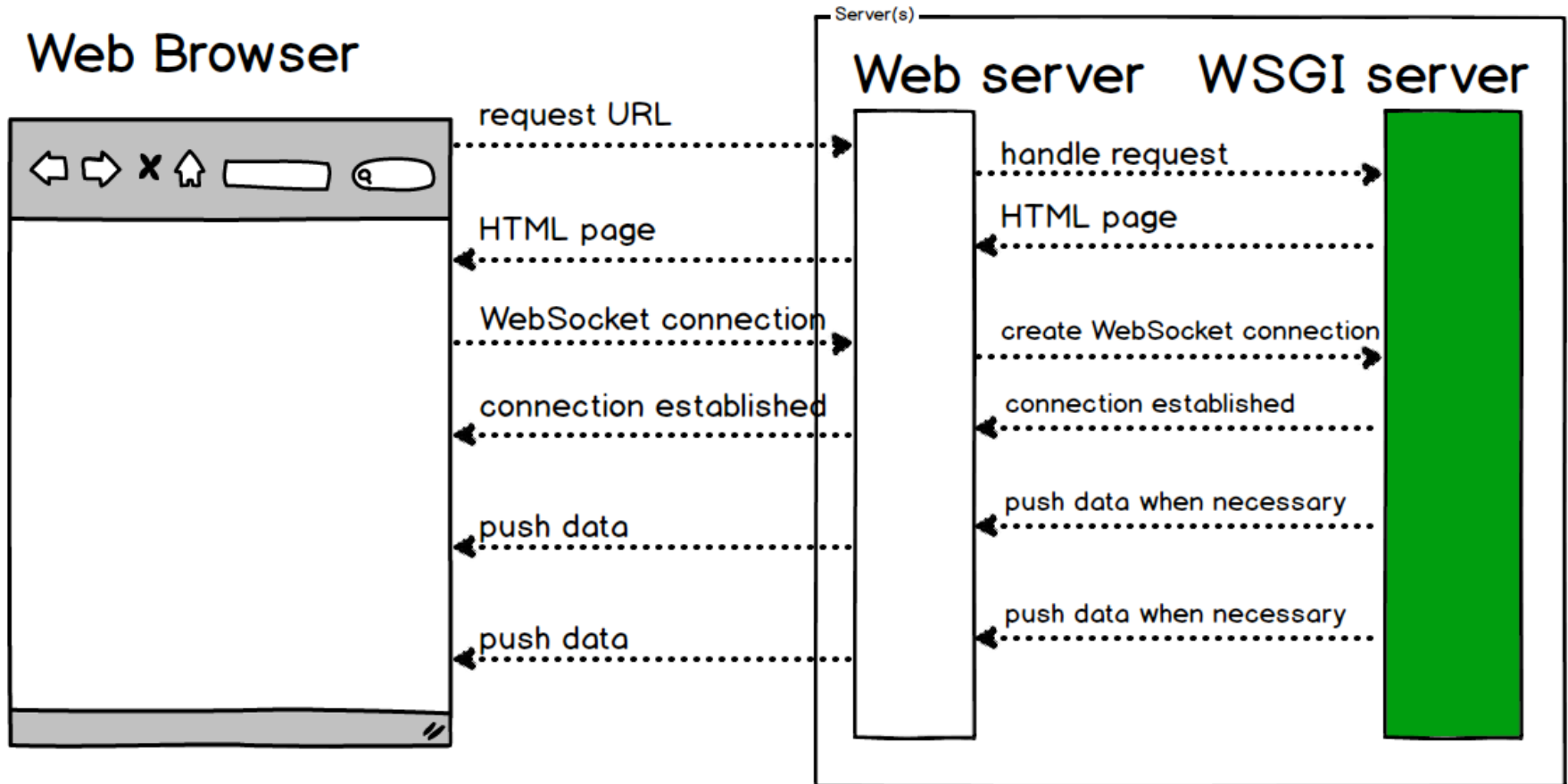
- What are they and why were they developed?
- Where are they commonly used?
- How to use them? (Demo)
- Watchouts when using websockets
- Where to learn more

# What are WebSockets?

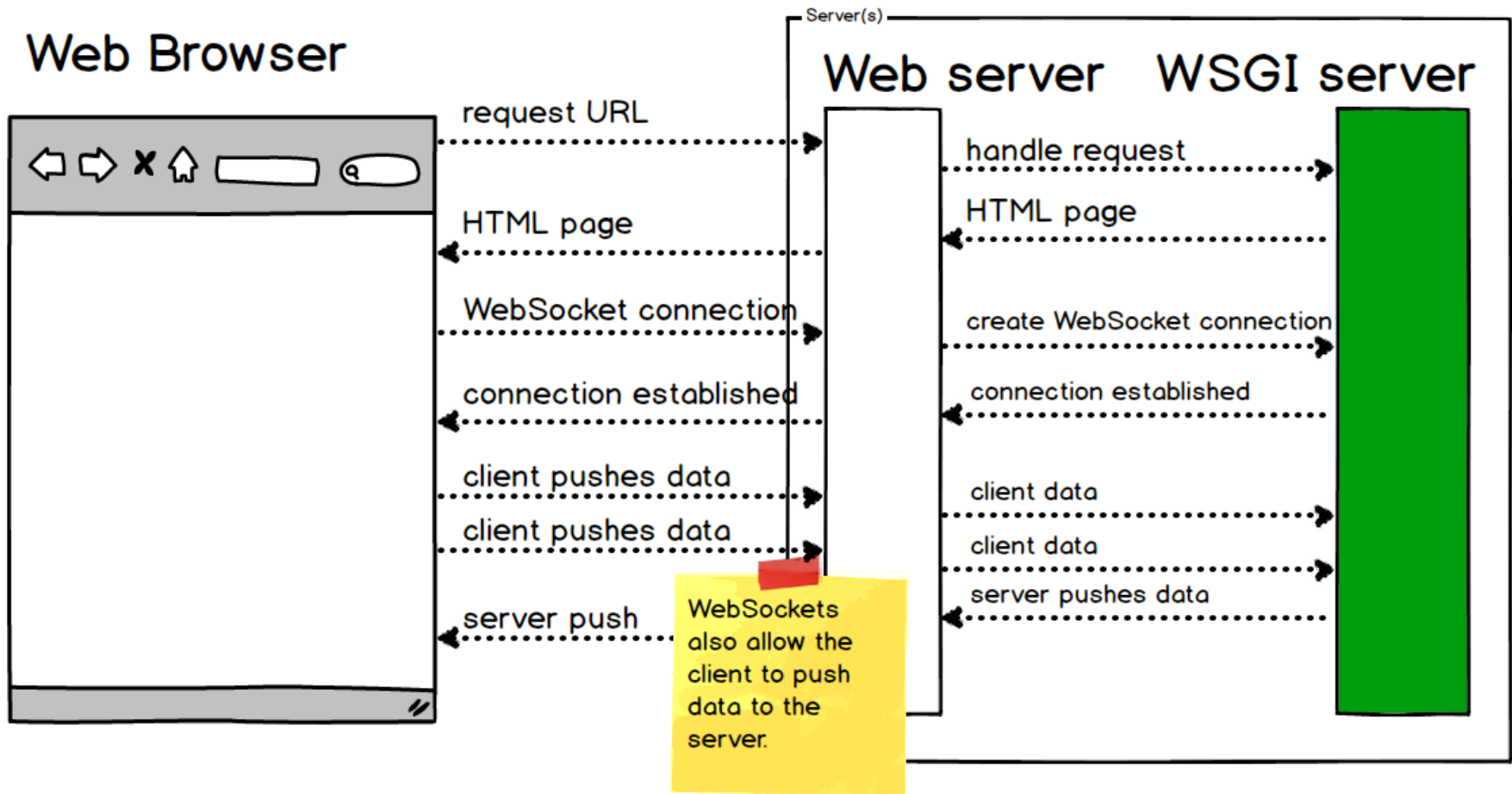
Part of the HTML5 specification that allows for full duplex communication between client and server over a single TCP connection.

- Low level protocol, bypasses "request-response"
- Client initiates via HTTP; server upgrades to TCP
- Sustained connection, both are listening, and both can transmit at any time

# WebSockets



# WebSockets



# Why WebSockets?

With the ongoing evolution of the web, people have come to expect a dynamic experience that is speedy and features realtime interaction.

- WebSockets provide true concurrency
- Reduce network traffic and latency
- Stream through proxies and firewalls with
- Simultaneous upstream and downstream support

# What are WebSockets Used For?

Applications where a sustained connection and/or true two-way communication is needed. Websockets are great for realtime streaming and interaction.

- Multi-player online games
- Dynamic data visualizations
- Chat applications
- Internet of Things
- Whatever you can dream up!

# How to Use WebSockets

Both the web browser and the server must implement the WebSockets protocol to establish and maintain the connection.

This can be done by writing client-side and server code from the ground up or by selecting from a variety of different frameworks and libraries that aid the process. See "[Resources](#)"



# Demo Intro

A super simple chat application utilizing an existing [GitHub repo](#).

Websockets are somewhat complicated and there are quite a few "gotchas," especially for someone new to them and to Python. (Read as "I got down in the weeds researching and didn't finish my data viz...yet.")

# Simple WebSocket Server

- Written in Python
- All latest browsers
- Support for Python 2 and 3

# Web Sockets - LS

Global 93.73% + 0.26% = 93.98%

unprefixed: 93.73% + 0.21% = 93.94%

Bidirectional communication technology for web apps

Current aligned
Usage relative
Date relative
Show all

IE	Edge <sup>*</sup>	Firefox	Chrome	Safari	Opera	iOS Safari <sup>*</sup>	Opera Mini <sup>*</sup>	Android Browser <sup>*</sup>	Chrome for Android
			49						
			55						
		51	56			9.3		4.4	
	14	52	57	10	43	10.2		4.4.4	
11	15	53	58	10.1	44	10.3	all	56	57
		54	59	TP	45				
		55	60		46				
		56	61						

[Notes](#)
[Known issues \(1\)](#)
[Resources \(10\)](#)
[Feedback](#)

Reported to be supported in some Android 4.x browsers, including Sony Xperia S, Sony TX and HTC.

Image from [caniuse.com](http://caniuse.com)

# Simple WebSocket Server

## Install

```
sudo pip install git+https://github.com/dpallot/  
simple-websocket-server.git
```

# Chat Server Example

```
python SimpleExampleServer.py --example chat
```

Open up multiple websocket.html files

# Watchout!

Very different way of thinking compared to HTTP

- HTTP stateless; WebSockets are stateful
- HTTP scales horizontally; WS vertically
- HTTP comes with caching, routing, multiplexing and more builtin
- Those things must be defined on top of the WebSocket protocol

# More Watchouts!

SECURITY! WS are cross origin by design. They allow you to arbitrarily tunnel many things to a browser. Be aware of the implications of that.

- Use `wss://` (WebSockets over SSL/TLS is encrypted)
- Avoid tunneling; develop secured and checked protocols on top of WebSockets
- Validate client input; validate server data
- WS does not handle authorization or authentication, must be added; opening from a page that has auth doesn't protect the WS

# General WebSocket Resources

- [W3C WebSockets API](#)
- [MDN](#)
- [Wikipedia](#)
- [IETF WebSocket Protocol, RFC 6455](#)
- [Full Stack Python Article](#)



# WebSocket Libraries, Frameworks, and Tutorials

- [Python Package Index](#)
- [Awesome Python](#)
- [Libraries.io](#)
- [Realtime Web Technologies Guide](#)

# In Closing

Thank you! It's been a great learning experience and I encourage others who are new to do a talk.

I'm excited to keep learning and finish the data viz that got me started on this. I'll share when done. Also - I've got lots more links if anyone wants more resources.

Contact me:

- Through DesertPy Meetup
- My website: [arielstrong.com](http://arielstrong.com)
- [LinkedIn](#)