

The Changing Face of Communications

IoT, REST, & Reactive

Todd L. Montgomery
@toddlmontgomery



*“Everything changes and
nothing stands still”*

— Heraclitus of Ephesus

IoT

There is *Hype*, but...

Large numbers are large...



Either ingest or streaming.
2x for Request/Response

Updates / Sec

=

Devices * Frequency * Market Share

Updates / Sec

=

Devices * Frequency * Market Share

9 Billion (Today)

50 Billion by 2020 (Cisco)

26 Billion by 2020 (Smartphone/Tablet - Gartner)

75 Billion by 2020 (Morgan Stanley)

Updates/Sec

=

50 Billion * 6/min * 1%

=

50 Million/sec

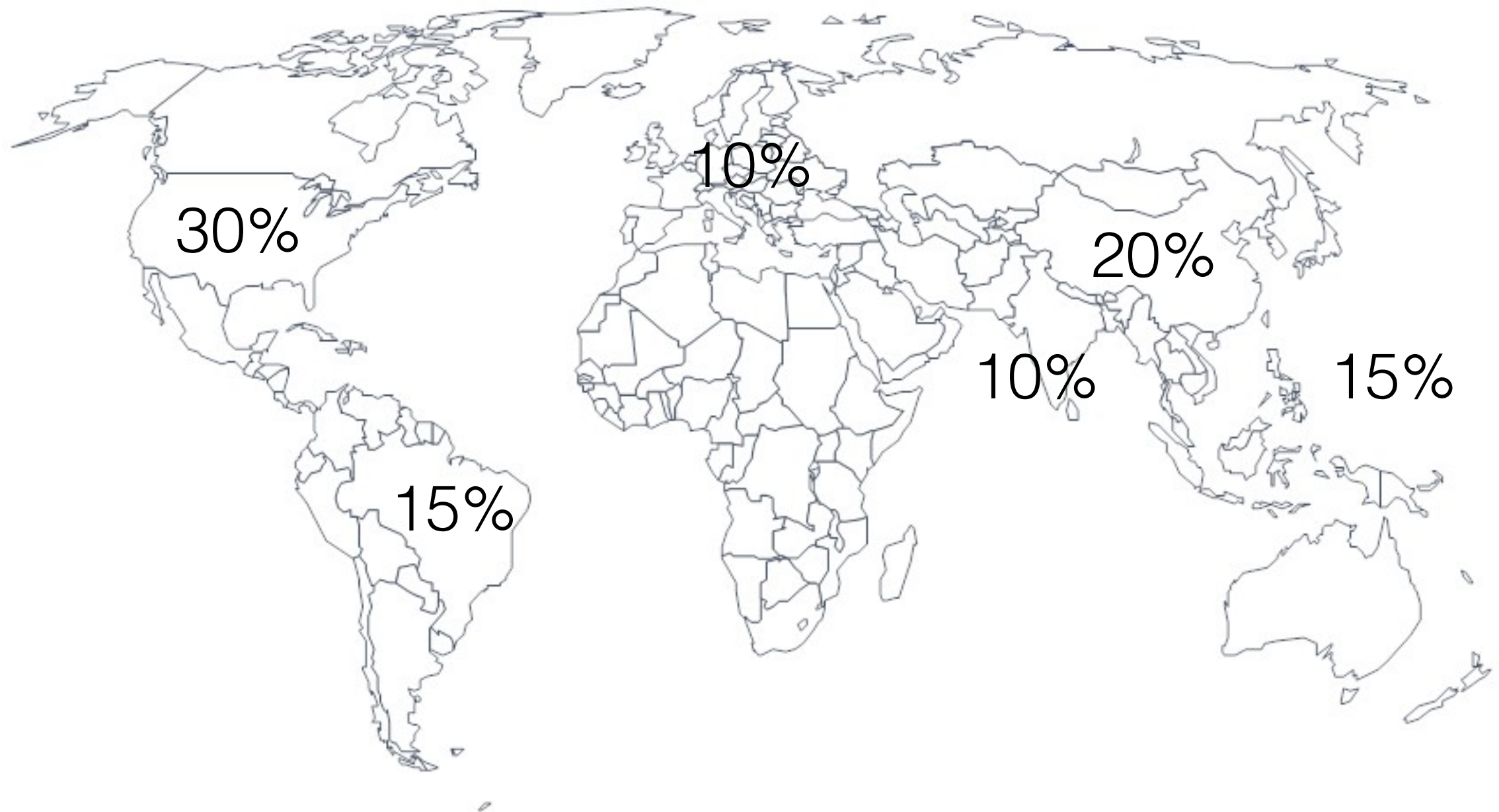
Bandwidth

=

50 Billion * 6/min * 1% * 200 bytes

=

9.3 GB/s (74.5 Gb/s)



And...
Geographic Distribution





OPEN INTERCONNECT
CONSORTIUM





VS

Google

VS



OPEN INTERCONNECT
CONSORTIUM

VS



ALLSEEN
ALLIANCE

courtesy of



What History Has Taught Us

~~*THE* IoT Protocol!~~

~~THE IoT _____!~~

Internet of Things

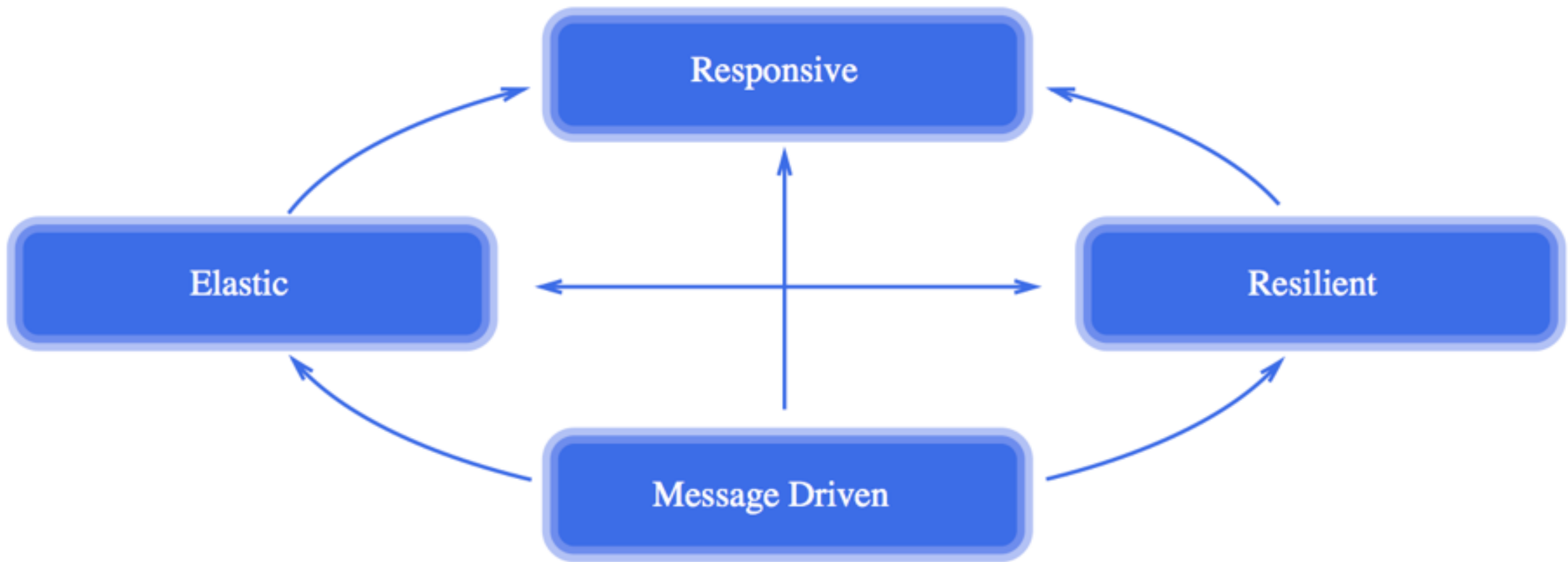
- ✓ *Multi-Protocol*
- ✓ *Multi-Language (Polyglot)*
- ✓ *Multi-Platform*
- ✓ *Huge Scale*
- ✓ *Highly Integrated*

It's not just the protocol

Architecture, Deployment, etc.

- ✓ *Micro-Services*
- ✓ *Reactive Extensions*
- ✓ *Functional Composition*
- ✓ *Immutability*
- ✓ *Containerization*
- ✓ *CI/CD*
- ✓ *...*

*The most scalable, resilient, &
fast systems seem to follow
some common principles...*



*In an emerging era of rapid change,
protocols of interaction matter*

*Like Data Structures
Like Schemas*

pro·to·col *noun* \ˈprō-tə-,kəl, -kōl, -,käl, -kəl\

...

3 b : a set of conventions governing the treatment and especially the formatting of data in an electronic communications system <network *protocols*>

...

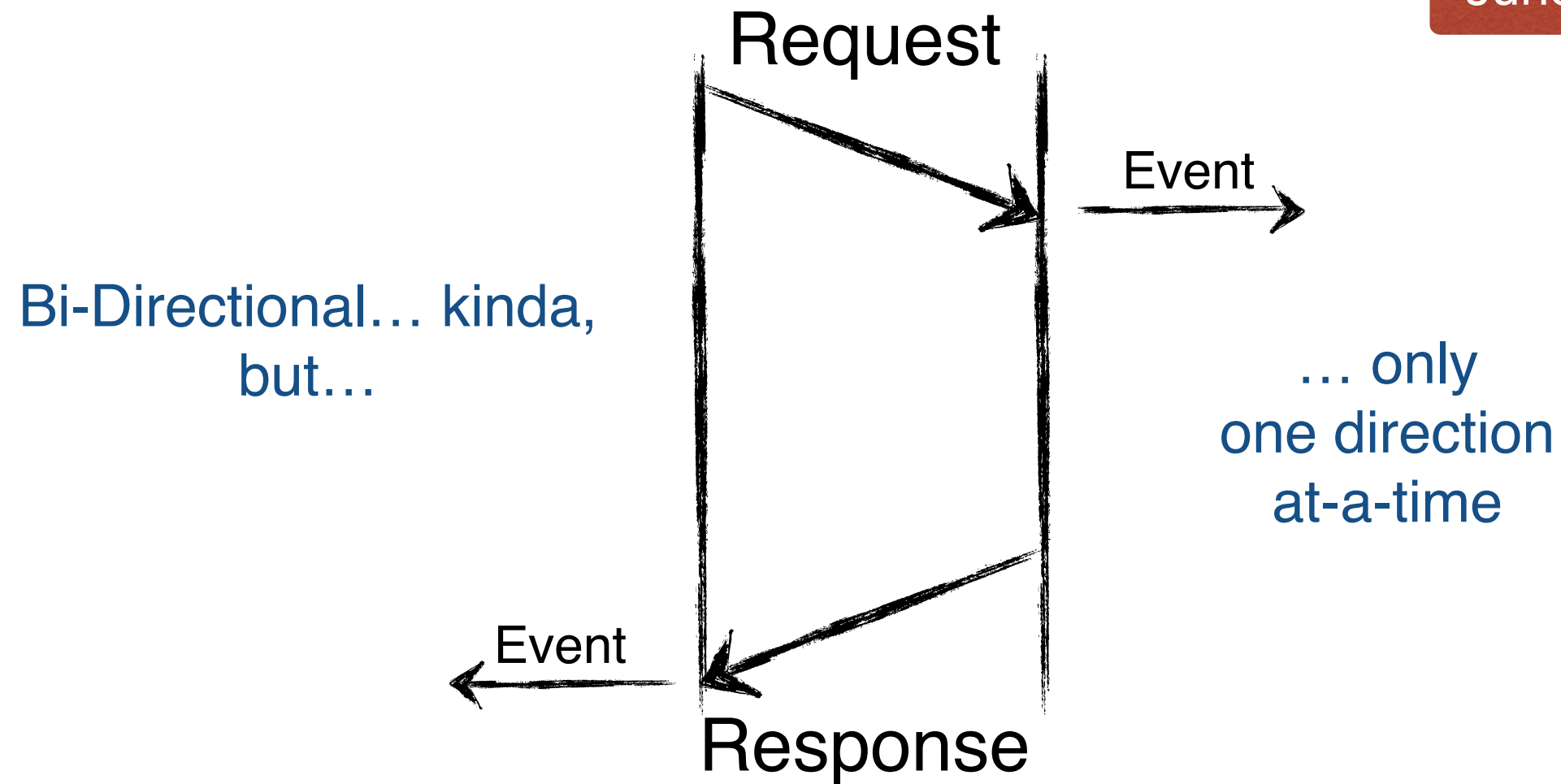
3 a : a code prescribing strict adherence to correct etiquette and precedence (as in diplomatic exchange and in the military services) <a breach of *protocol*>

Where we are coming from...

HTTP

RFC 2068, 2616, ..., 7230-7240

June 2014



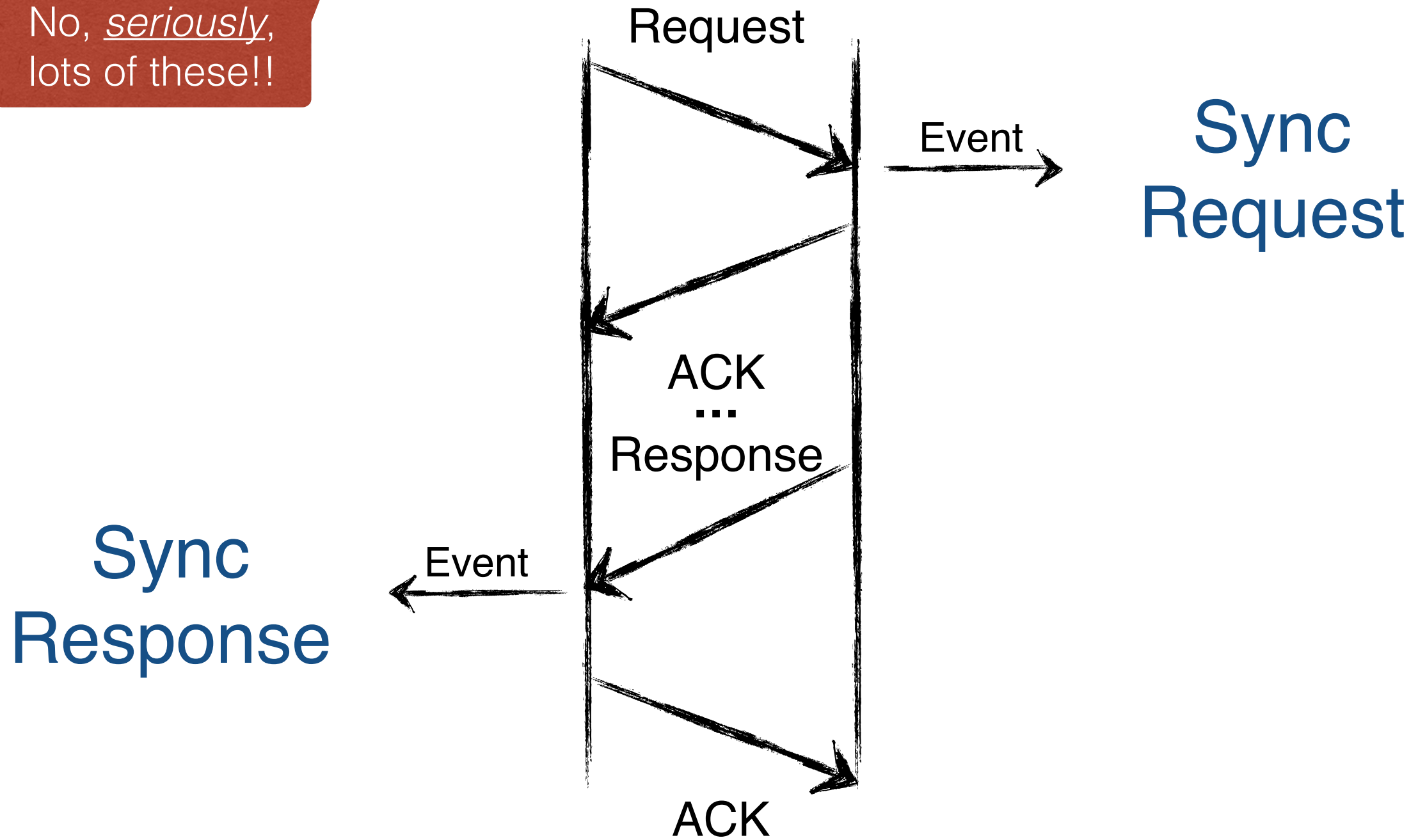
Synchronous
Request/Response

- ✓ *ASCII Encoded*
- ✓ *Very Synchronous*
- ✓ *Many TCP Connections*

Web Services

http://en.wikipedia.org/wiki/List_of_web_service_specifications

No, *seriously*,
lots of these!!



But... Async Request/Response... kinda

But... HTTP is
THE Most Successful Protocol
EVAR!

*“Yeah, yeah, but your scientists were so preoccupied with whether or not they **could** that they didn't stop to think if they **should**.”*
— *Jurassic Park*

*Just because you **could** use HTTP,
doesn't mean you **should**...*

HTCPCP

RFC 2324, Extended by RFC 7168



"there is a strong, dark, rich requirement for a protocol designed espressoly [sic] for the brewing of coffee"

[[Docs](#)] [[txt](#)|[pdf](#)] [[Errata](#)]

Updated by: [7168](#)

Network Working Group
Request for Comments: 2324
Category: Informational

INFORMATIONAL
Errata Exist
L. Masinter
1 April 1998

Hyper Text Coffee Pot Control Protocol (HTCPCP/1.0)

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (1998). All Rights Reserved.

Abstract

This document describes HTCPCP, a protocol for controlling, monitoring, and diagnosing coffee pots.

1. Rationale and Scope



BREW

WHEN

418 I'm a teapot

"This has a serious purpose – it identifies many of the ways in which HTTP has been extended inappropriately."
— Larry Masinter, author
<http://larry.masinter.net/>

*Extending Appropriately**

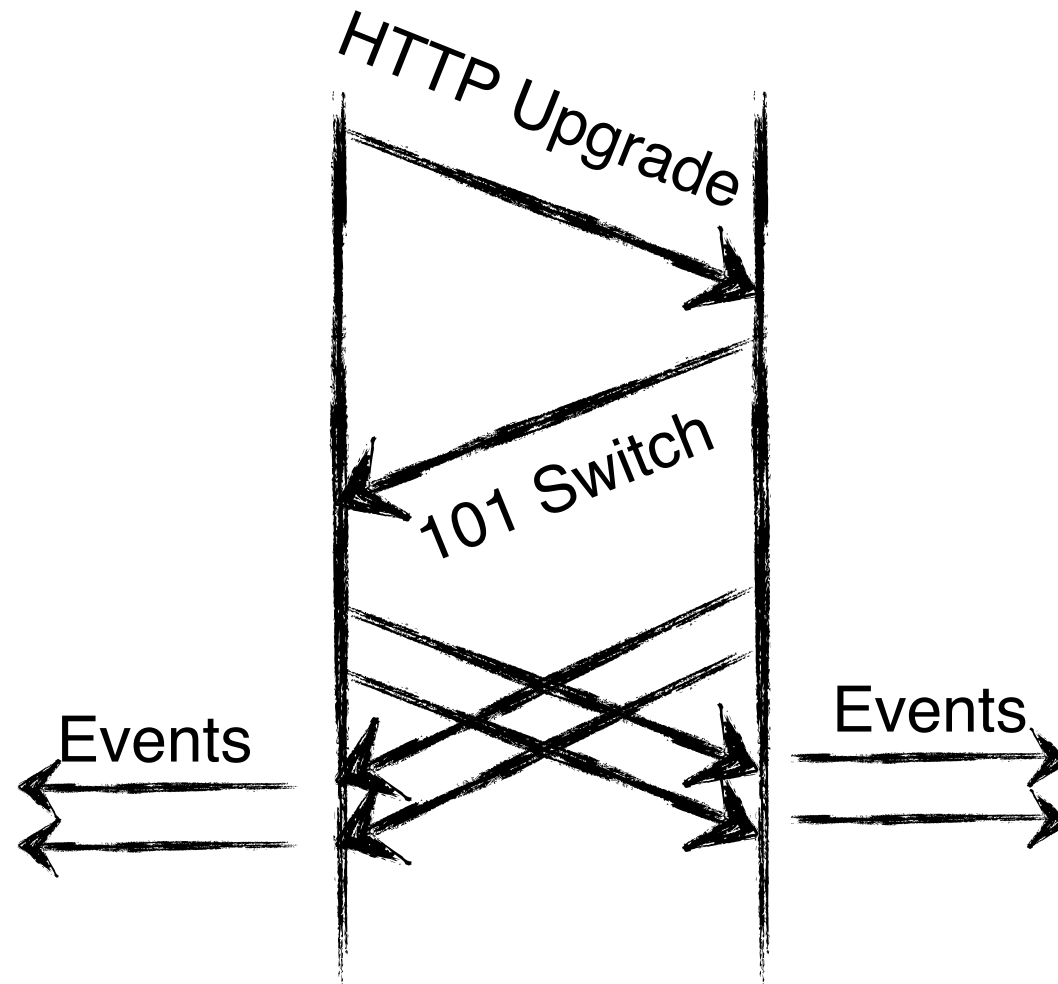
* - some would disagree...

WebSocket RFC 6455

Really a
Transport
Protocol

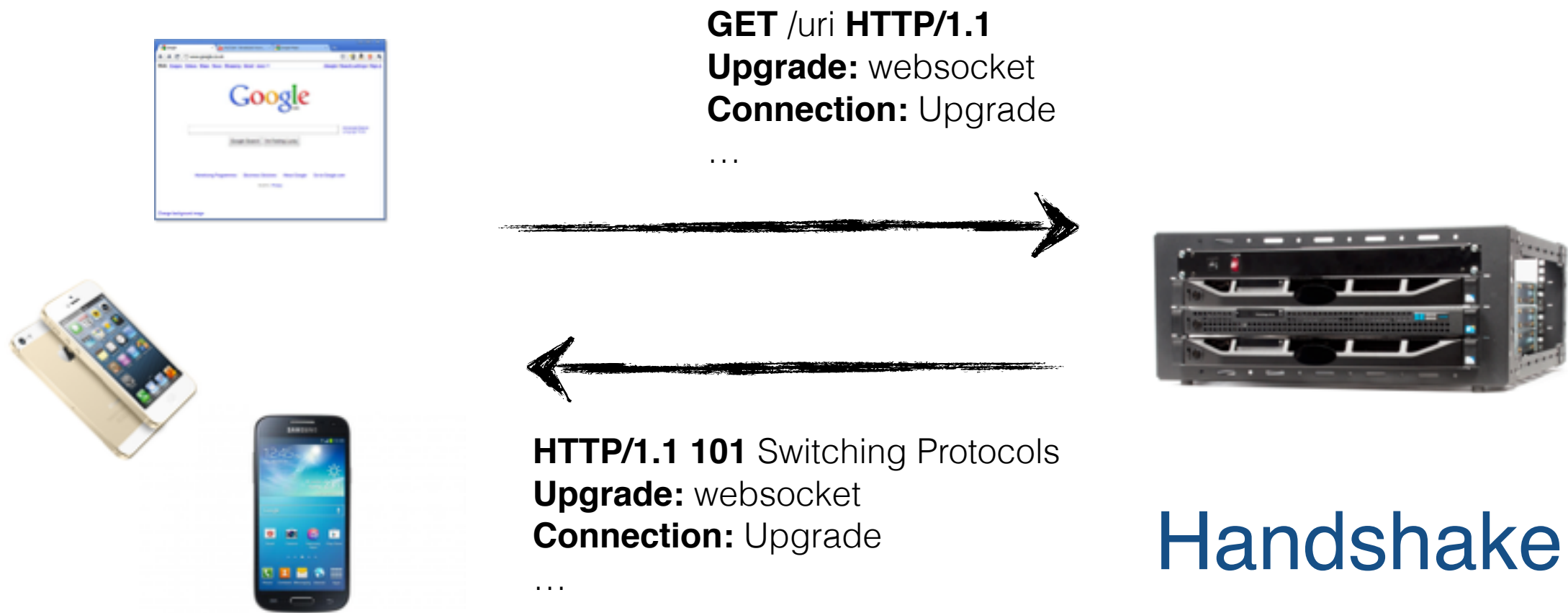
Async
Request/
Response

Streaming

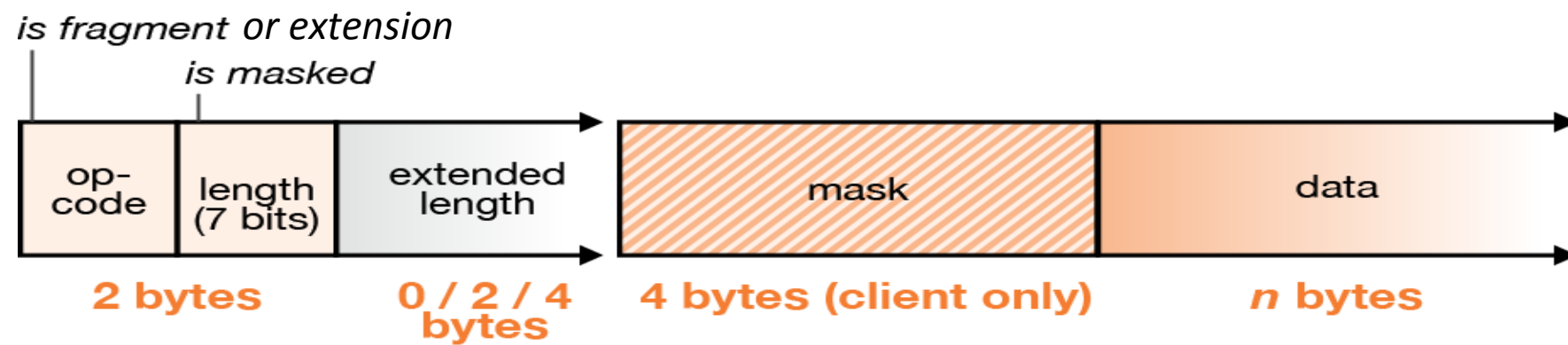


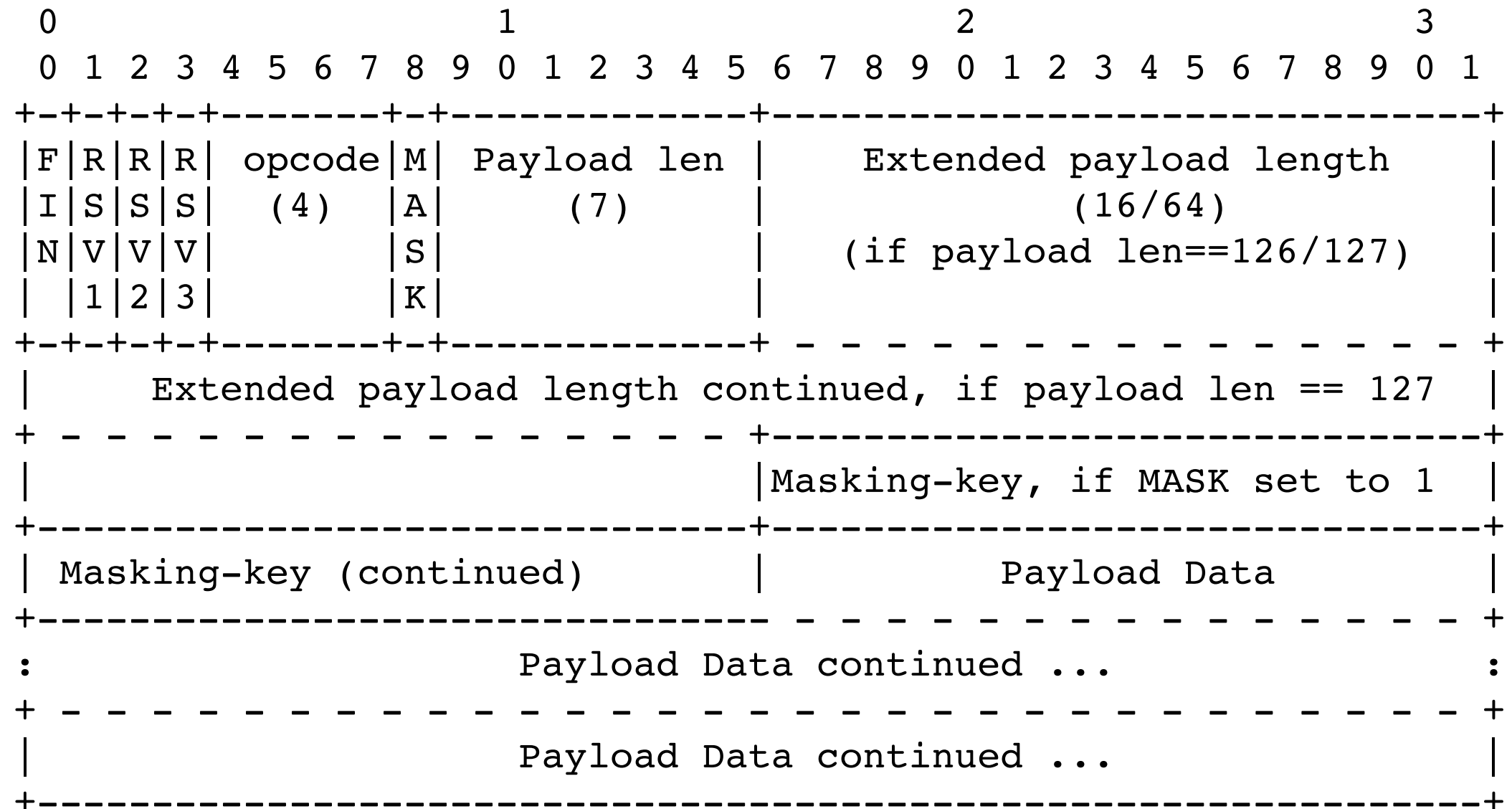
Ingest

Full Duplex, Asynchronous
“TCP over the Web”



Simple Framing

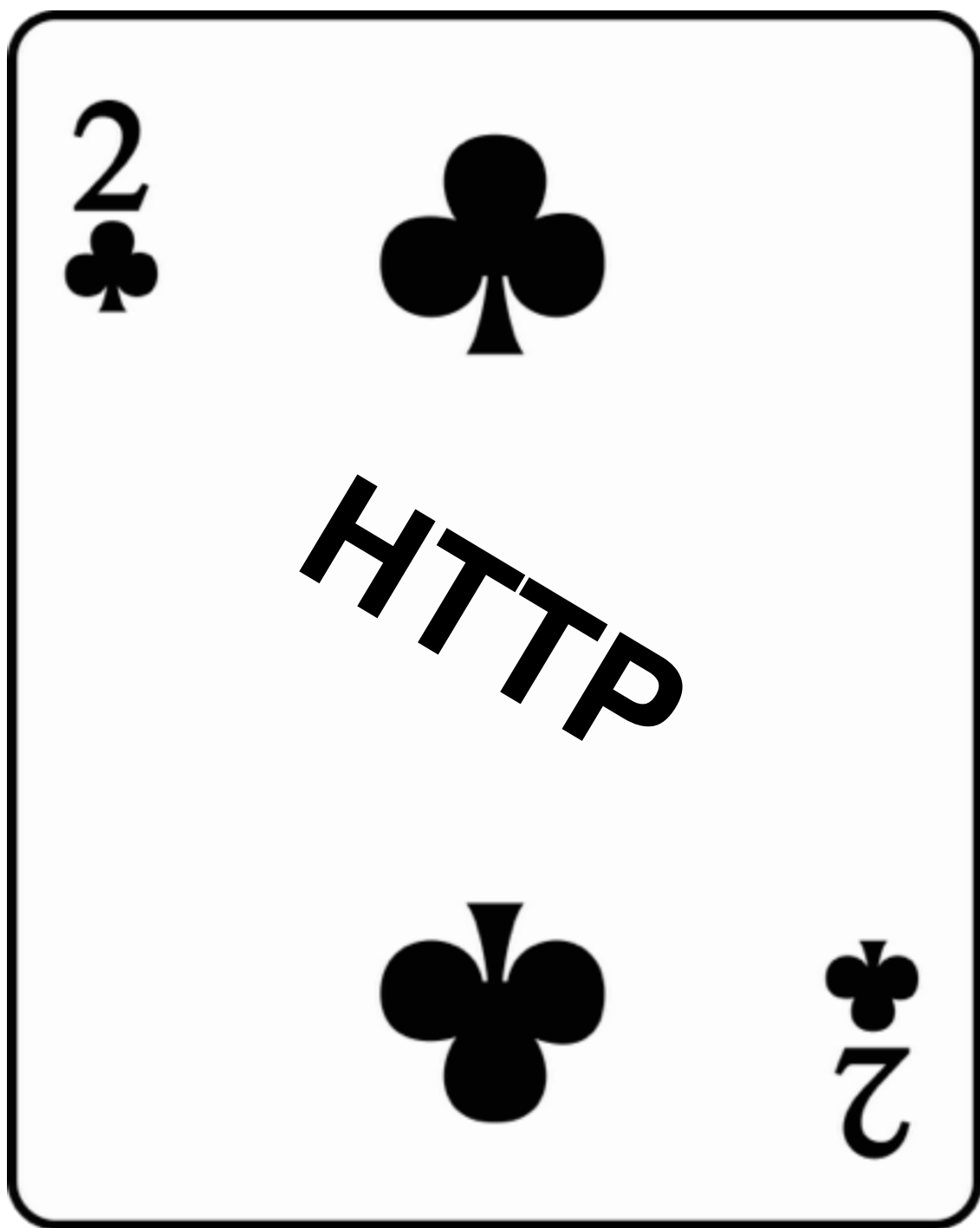




WebSocket was a scout...

- ✓ *Persistent Connections*
- ✓ *“Different” Framing*
- ✓ *Binary Encoding*

scouting for...



There is emerging implementation experience and interest in a protocol that retains the semantics of HTTP without the legacy of HTTP/1.x message framing and syntax, which have been identified as hampering performance and encouraging misuse of the underlying transport.

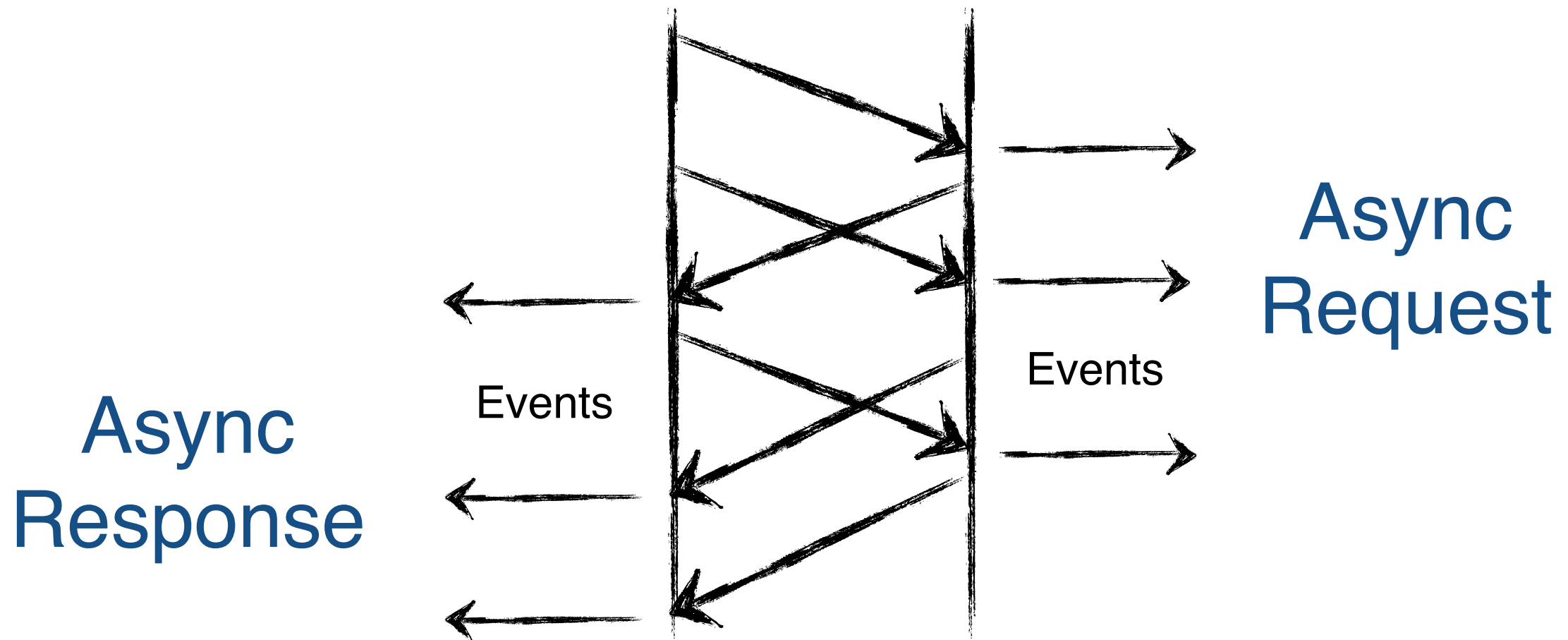
— IETF httpbis Charter

<http://datatracker.ietf.org/wg/httpbis/charter/>

Model on SPDY

SPDY & HTTP/2

IETF Drafts



Async Request/Response
Streaming (Server Push)

- ✓ *Persistent Connection*
- ✓ *Binary Encoding*
- ✓ *Multiple Streams*
- ✓ *Efficient Headers (HPACK)*
- ✓ *Server Push*

HPACK

- ✓ *Binary Encoding*
- ✓ *Huffman Encoding*
- ✓ *Dynamic Table*
- ✓ *Static Table*
- ✓ *HTTP Method included*

name	language	role(s)	negotiation(s)	protocol id(s)
Akamai Ghost	C++	intermediary	ALPN, NPN	h2-14
Apache Traffic Server v5.3.0	C++	intermediary	ALPN, NPN	h2, h2-14
Chromium	C++	client	ALPN, NPN	h2, h2-14
cl-http2-protocol	Common Lisp	client, server	NPN, direct	
curl and libcurl	C	client	ALPN, NPN, Upgrade	
Deuterium	C	client, server	ALPN, direct	h2, h2-14, h2c, h2c-14
Ericsson MSP		proxy	NPN, Upgrade, direct	
F5	C	server, proxy	ALPN, NPN	
H2O	C	Server, proxy	ALPN, NPN, Upgrade, direct	h2, h2-14, h2-16
Haskell http2 lib	Haskell	HPACK, framing		
http-2	Ruby	server, client	ALPN, NPN, Upgrade, direct	h2, h2c, h2-17
http2	Go	client, server	NPN (+ ALPN w/ Go 1.4)	
hyper	Python	client	NPN	h2, h2-16, h2-15, h2-14
Jetty	Java	client, intermediary, server	ALPN, Upgrade, Direct	h2, h2-17, h2-14, h2c, h2c-17

Lucid	Erlang	Server	NPN, direct	h2, h2-16, h2-14
Microsoft	C/C++	Client, Server	ALPN	h2
mod_h2	C	Server	ALPN, NPN, Upgrade	h2c?(-1[46])?
Mozilla Firefox	C++	client	ALPN, NPN	h2-15, h2-14, h2
Netty	Java	client, server	ALPN, NPN	
nhttp2	C	client, server, intermediary	ALPN, NPN, Upgrade, direct	h2, h2-16, h2-14, h2c-14
node-http2	NodeJS	server, client	ALPN, NPN, direct	h2
OkHttp	Android, Java	mock server, client	ALPN, NPN	
OpenLiteSpeed	C++	Server	ALPN, NPN, Upgrade	
Protocol::HTTP2	Perl	server, client	ALPN, NPN, Upgrade, direct	h2-17, h2-14, h2c-17, h2c-14
Riverbed SteelApp ADC	C++	Server	ALPN, NPN, Upgrade, direct	h2-14, h2c-14
Sasazka	NodeJS	server	NPN	
Test GFE	C++	intermediary	ALPN, NPN	
Trusterd	C/mruby	client, server	ALPN, NPN, direct	
Twitter	C++	server, client	ALPN, NPN	
Undertow	Java	Server, Intermediary	ALPN, Upgrade	
Warp	Haskell	Server	ALPN, direct	
Wireshark	C	other	ALPN, NPN, Upgrade, direct	

<https://github.com/http2/http2-spec/wiki/Implementations>

IETF

*Constrained RESTful Environments
(CoRE) Working Group*

“Constrained” isn’t what you might think...

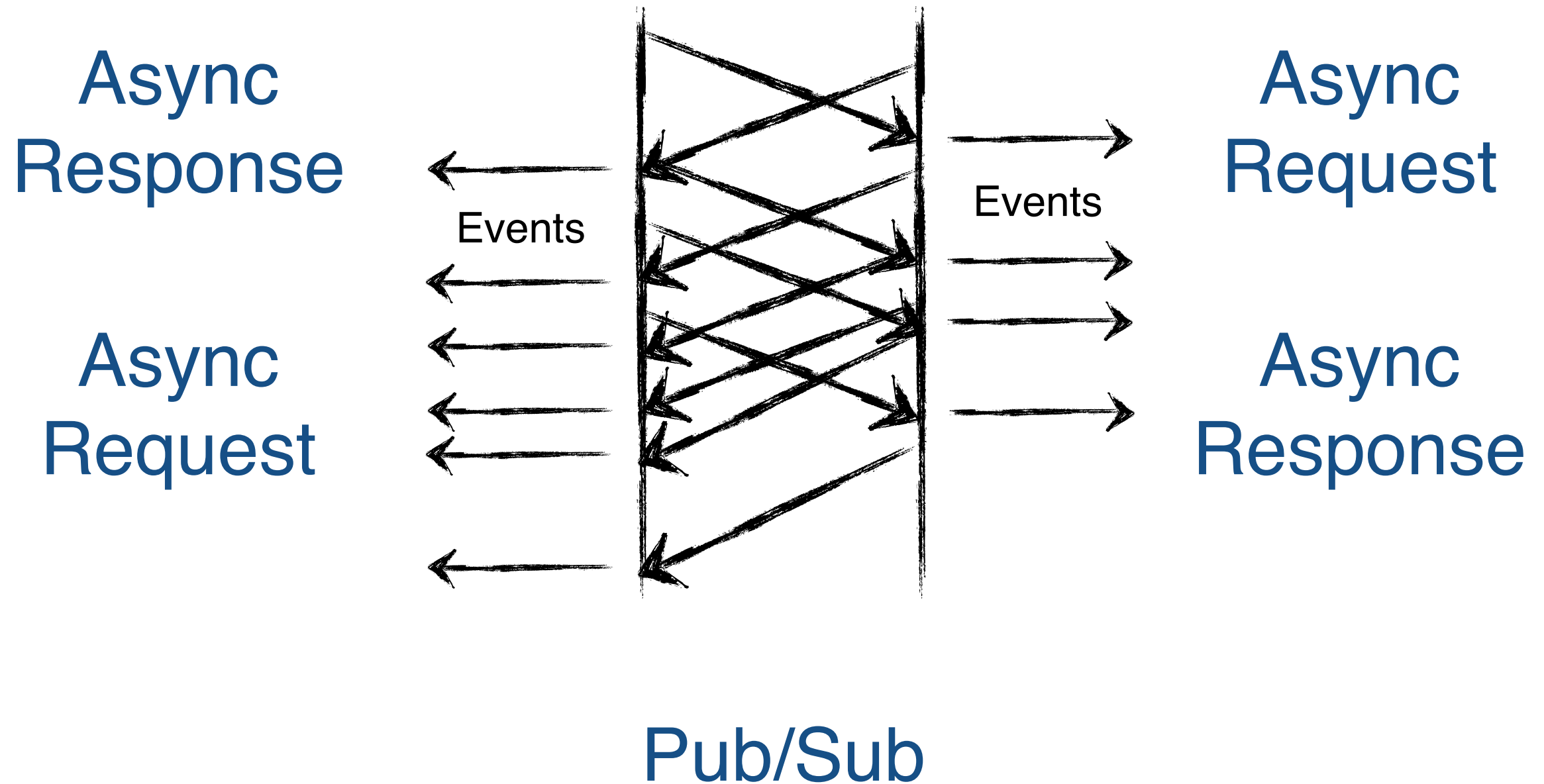
CoRE is providing a framework for resource-oriented applications intended to run on constrained IP networks. A constrained IP network has limited packet sizes, may exhibit a high degree of packet loss, and may have a substantial number of devices that may be powered off at any point in time but periodically "wake up" for brief periods of time.

— IETF CoRE Charter

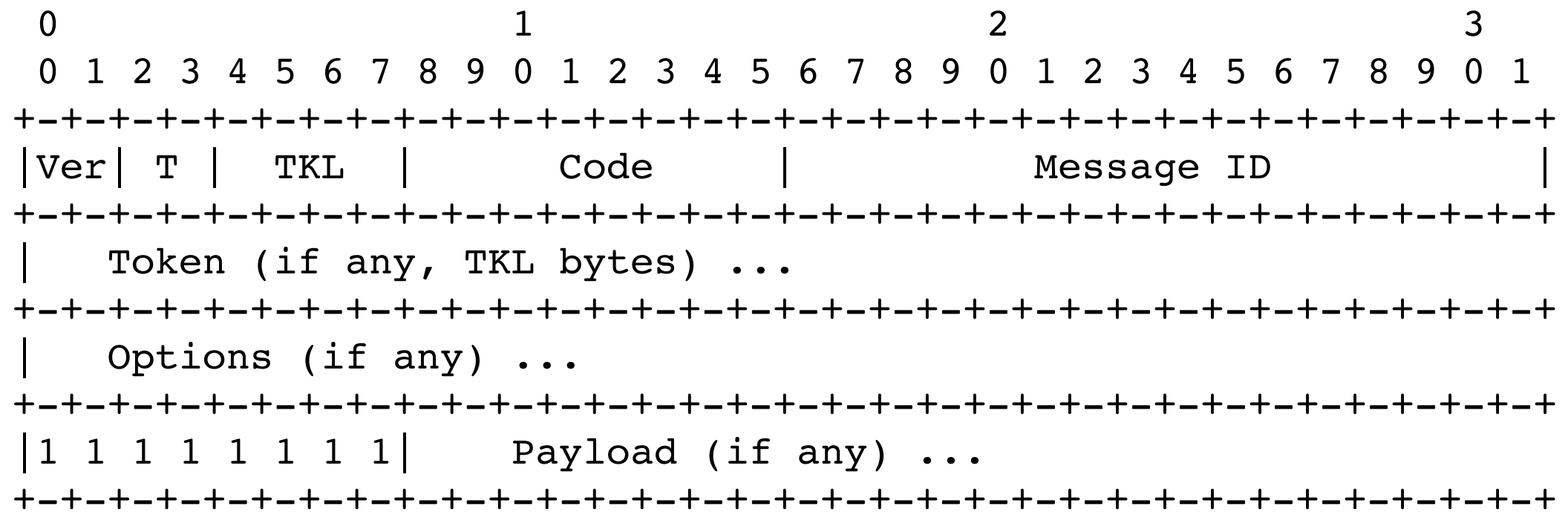
<http://datatracker.ietf.org/wg/core/charter/>

Constrained Application Protocol (CoAP)

RFC 7252



- ✓ *HTTP Mapping*
- ✓ *UDP, DTLS, & WebSocket*
- ✓ *Resource Discovery*
- ✓ *Linking*



Name	Programming Language	Implemented CoAP version	Client/Server	Implemented CoAP features	License	Link
Californium	Java	RFC 7252	Client + Server	Observe, Blockwise Transfers, DTLS	EPL+EDL	https://www.eclipse.org/californium
cantcoap	C++/C	RFC 7252	Client + Server		BSD	https://github.com/staropram/cantcoap
CoAP implementation for Go	Go	RFC 7252	Client + Server	Core + Draft Subscribe	MIT	https://github.com/dustin/go-coap
CoAP.NET	C#	RFC 7252 , coap-13, coap-08, coap-03	Client + Server	Core, Observe, Blockwise Transfers	3-clause BSD	https://github.com/smeshlink/CoAP.NET
CoAPSharp	C#, .NET	RFC 7252	Client + Server	Core, Observe, Block, RD	LGPL	http://www.coapsharp.com
CoAPthon	Python	RFC 7252	Client + Server + Forward Proxy + Reverse Proxy	Observe, Multicast server discovery, CoRE Link Format parsing	BSD	https://github.com/Tanganelli/CoAPthon
Copper	JavaScript (Browser Plugin)	RFC 7252	Client	Observe, Blockwise Transfers	3-clause BSD	https://github.com/mkovatsc/Copper https://addons.mozilla.org/de/firefox/addon/copper-270430/
Erbium for Contiki	C	RFC 7252	Client + Server	Observe, Blockwise Transfers	3-clause BSD	http://www.contiki-os.org/ (er-rest-example)
ETRI CoAP	C	RFC 7252	Client + Server	Core, Observe, Block	Commercial	http://coap.or.kr/index_en.html
iCoAP	Objective-C	RFC 7252	Client	Observe, Blockwise Transfers	MIT	https://github.com/stuffrabbit/iCoAP
jCoAP	Java	RFC 7252	Client + Server	Observe, Blockwise Transfers	Apache License 2.0	https://code.google.com/p/jcoap/
libcoap	C	RFC 7252	Client + Server	Observe, Blockwise Transfers	BSD/GPL	http://sourceforge.net/projects/libcoap/develop
microcoap	C	RFC 7252	Client + Server		MIT	https://github.com/1248/microcoap
nCoap	Java	RFC 7252	Client + Server	Observe	BSD	https://github.com/okleine/nCoAP
node-coap	Javascript	RFC 7252	Client + Server	Core, Observe, Block	MIT	https://github.com/mcollina/node-coap
Ruby coap	Ruby	RFC 7252	Client + Server (david)	Core, Observe, Block, RD	MIT, GPL	https://github.com/nning/coap https://github.com/nning/david
Sensinode C Device Library	C	RFC 7252	Client + Server	Core, Observe, Block, RD	Commercial	https://silver.arm.com/browse/SEN00
Sensinode Java Device Library	Java SE	RFC 7252	Client + Server	Core, Observe, Block, RD	Commercial	https://silver.arm.com/browse/SEN00
Sensinode NanoService Platform	Java SE	RFC 7252	Cloud Server	Core, Observe, Block, RD	Commercial	https://silver.arm.com/browse/SEN00
SMCP	C	RFC 7252	Client + Server	Core, Observe, Block	MIT	https://github.com/darconeous/smcp
TinyOS CoapBlip	nesC/C	coap-13	Client + Server	Observe, Blockwise Transfers	BSD	http://docs.tinyos.net/tinywiki/index.php/CoAP
txThings	Python (Twisted)	RFC 7252	Client + Server	Blockwise Transfers, Observe (partial)	MIT	https://github.com/siskin/txThings/

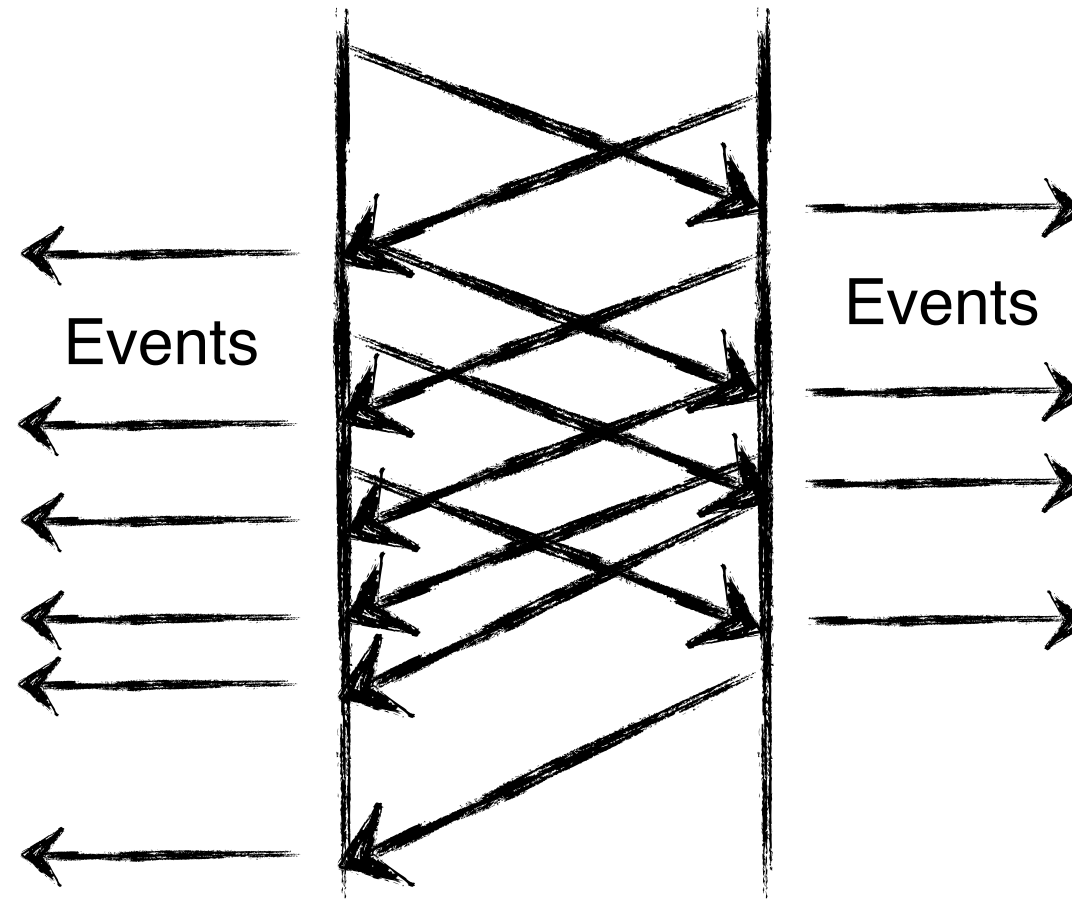
MQ Telemetry Transport (MQTT) OASIS Standard

Async
Response

Async
Request

Streaming

Ingest

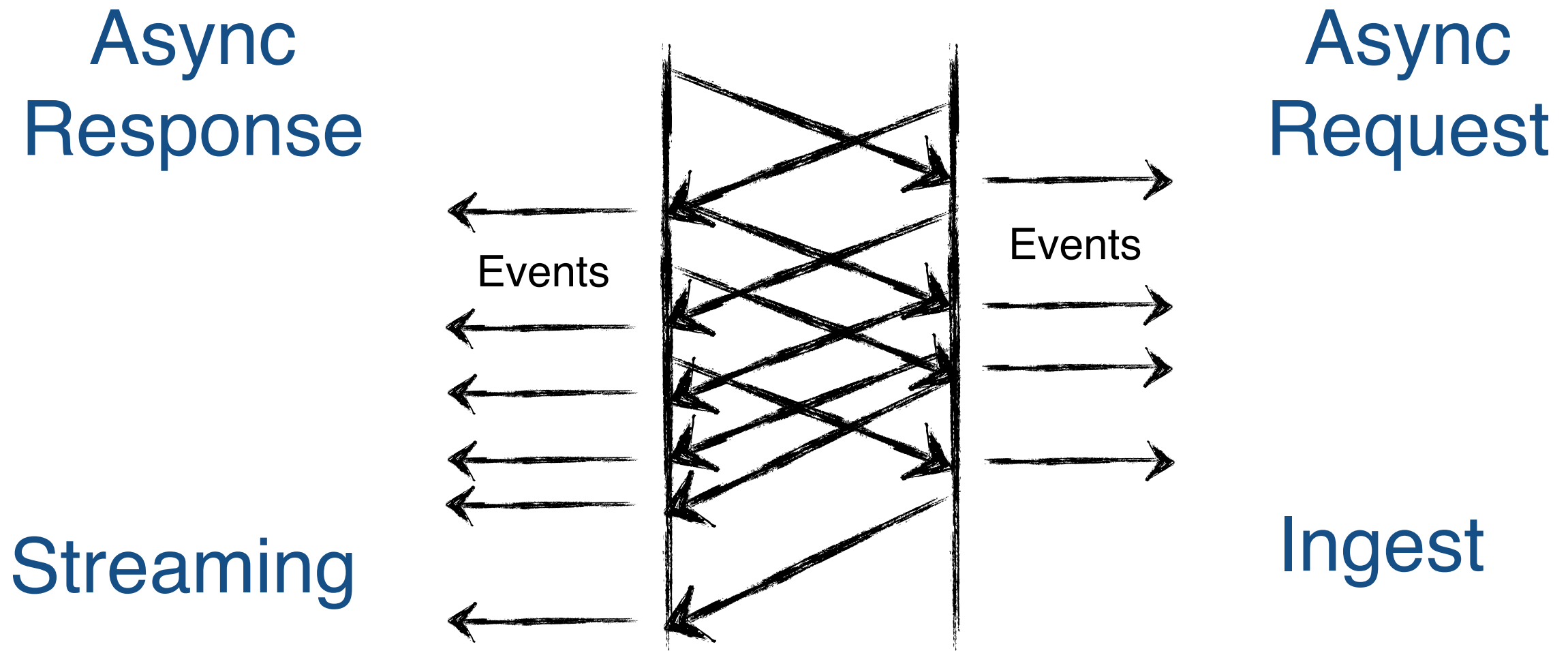


Lightweight Pub/Sub

- ✓ *TCP or WebSocket (3.1.1)*
- ✓ *MQTT-SN for non-TCP*

Extensible Messaging & Presence Protocol (XMPP)

RFC 6120 (Obsoletes 3920)



Presence & Persistent Streams

<http://xmpp.org/>

<https://tools.ietf.org/html/rfc6120>

- ✓ *XMPP Standards Foundation*
- ✓ *<http://xmpp.org/>*
- ✓ *TCP + TLS + SASL*
- ✓ *(Semi) Structured XML + Binary*

Only scratched the surface...

- ✓ *DDS*
- ✓ *WebRTC*
- ✓ *Aeron*
- ✓ *...*

Takeaways!

Protocols of Interaction Matter

*Look for the
right tool (protocol) for the job*

Like a Data Structure...

Like a Schema...

Questions?

- Reactive Manifesto <http://www.reactivemanifesto.com>
- Aeron <https://github.com/real-logic/Aeron>
- SlideShare <http://www.slideshare.com/toddleemontgomery>
- Twitter @toddlmontgomery

Thank You!