

How we scaled push messaging for millions of Netflix devices

Susheel Aroskar
Cloud Gateway

NETFLIX

**Why do we need
push?**



Crime TV Comedies



Cerebral Crime TV Shows

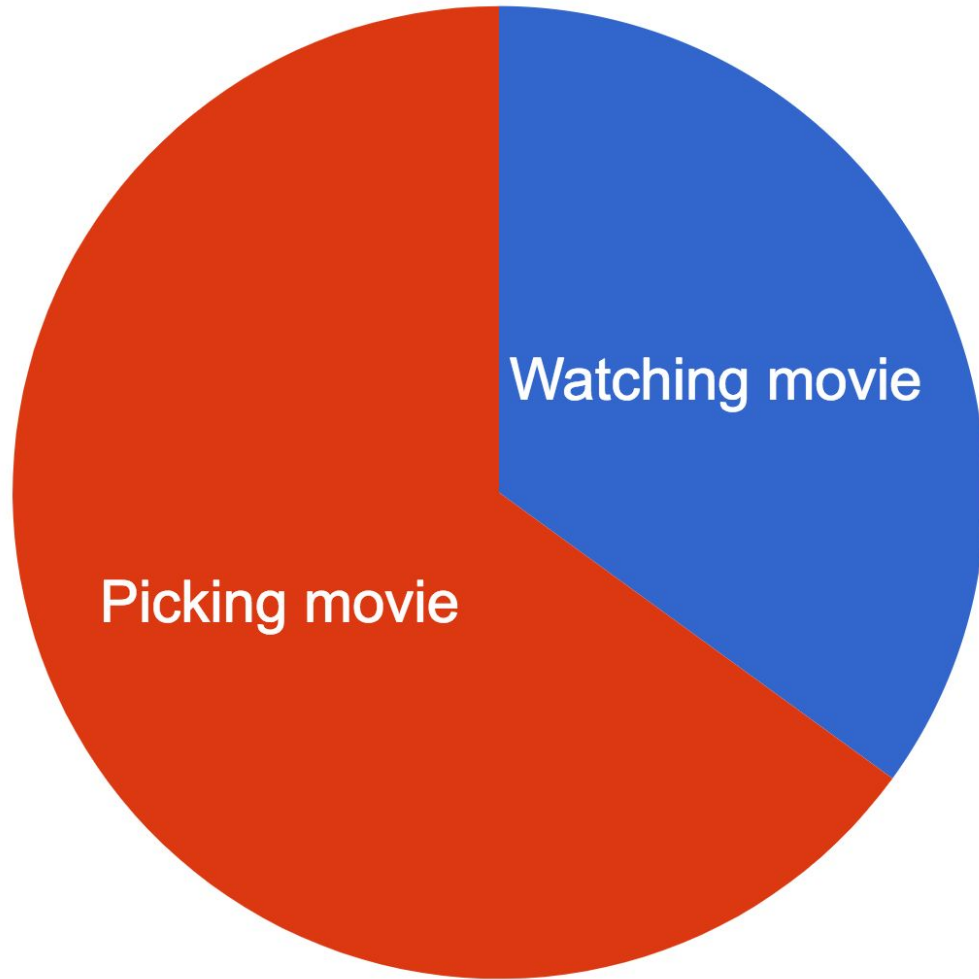


Because you watched Collateral



British TV Shows





How I spend my
time in Netflix
application...

- What is push?

- What is push?
- How you can build it

- What is push?
- How you can build it
- How you can operate it

- What is push?
- How you can build it
- How you can operate it
- What can you do with it



Susheel Aroskar

Senior Software Engineer
Cloud Gateway
saroskar@netflix.com

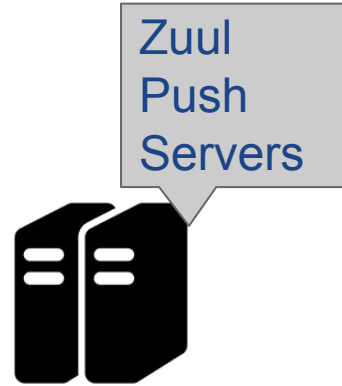
 github.com/raksoras
 [@susheelaroskar](https://twitter.com/susheelaroskar)

PERSIST
UNTIL
SSOMETHING
HHAPPENS

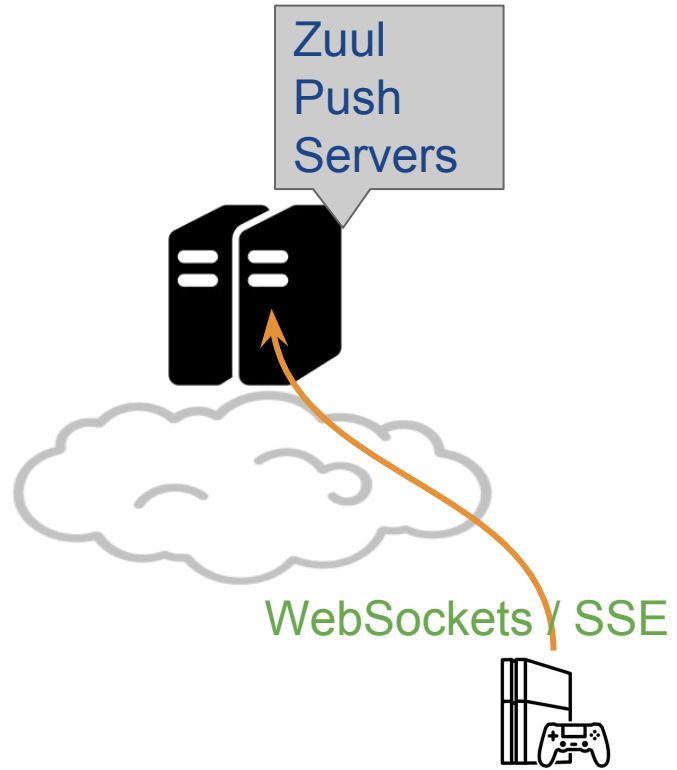
PERSIST
UNTIL
SSOMETHING
HHAPPENS

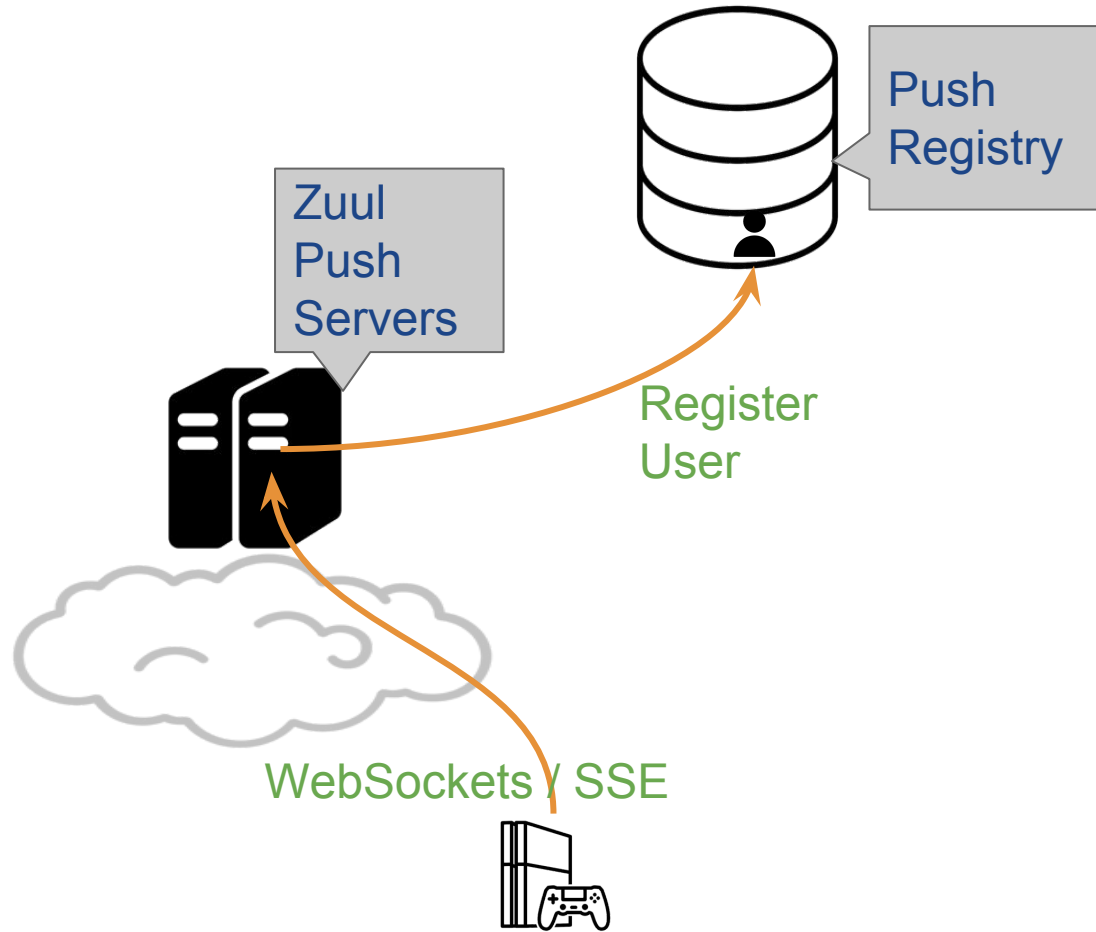
Zuul Push Architecture

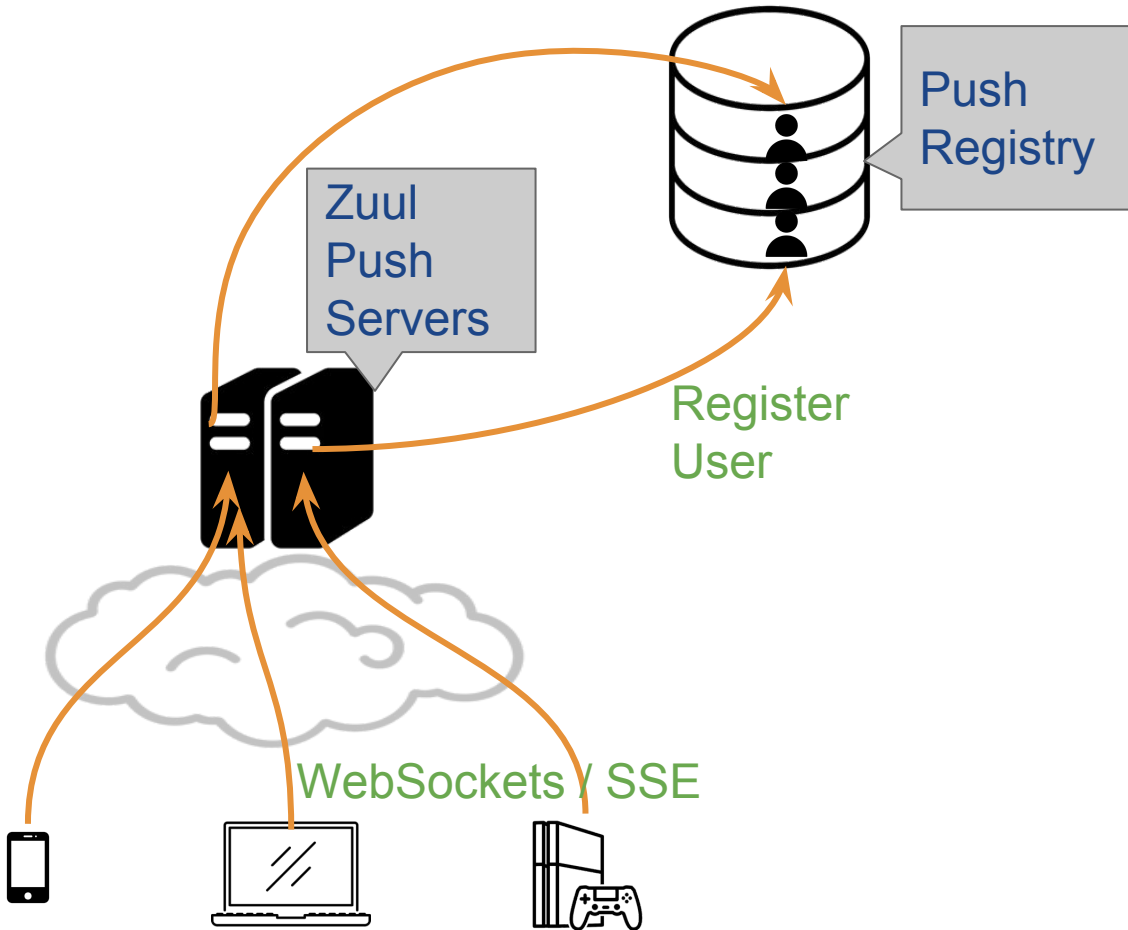


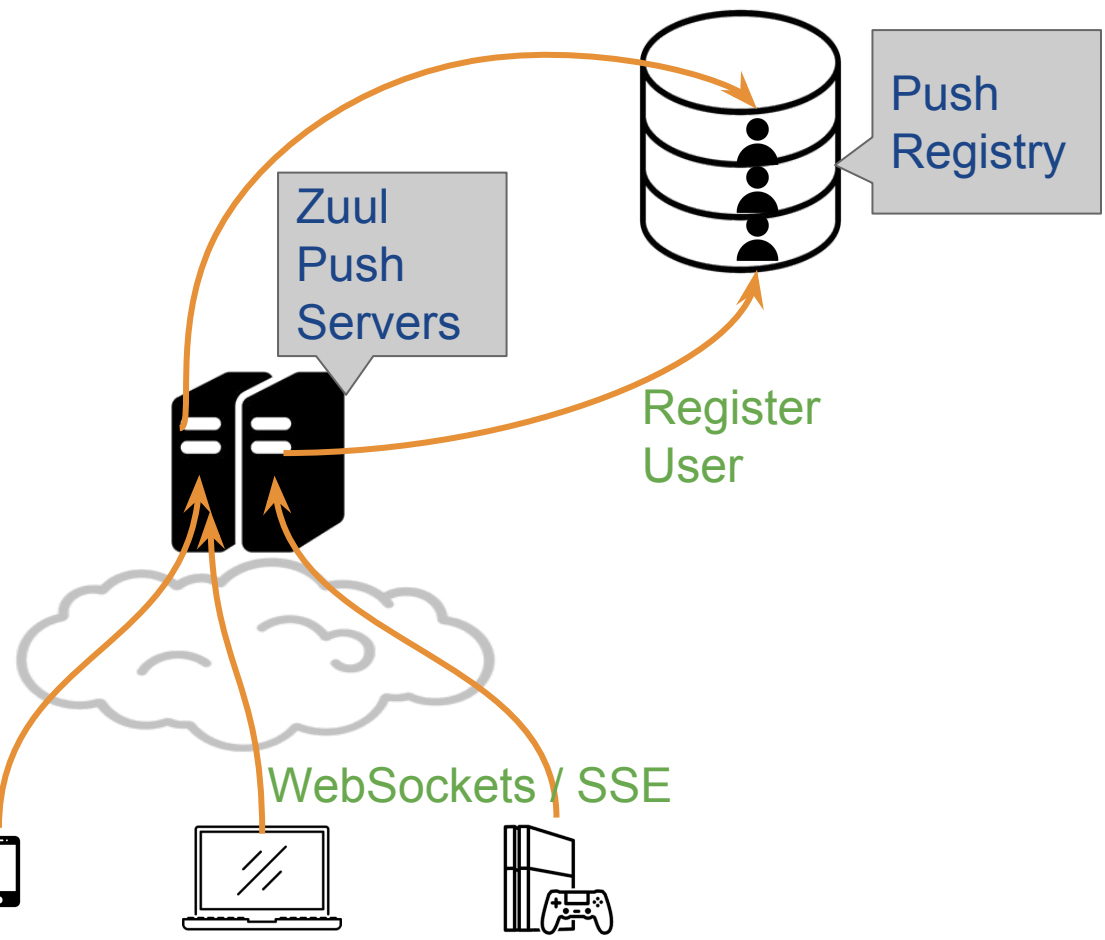
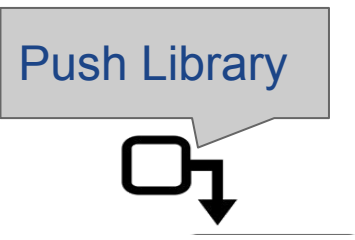


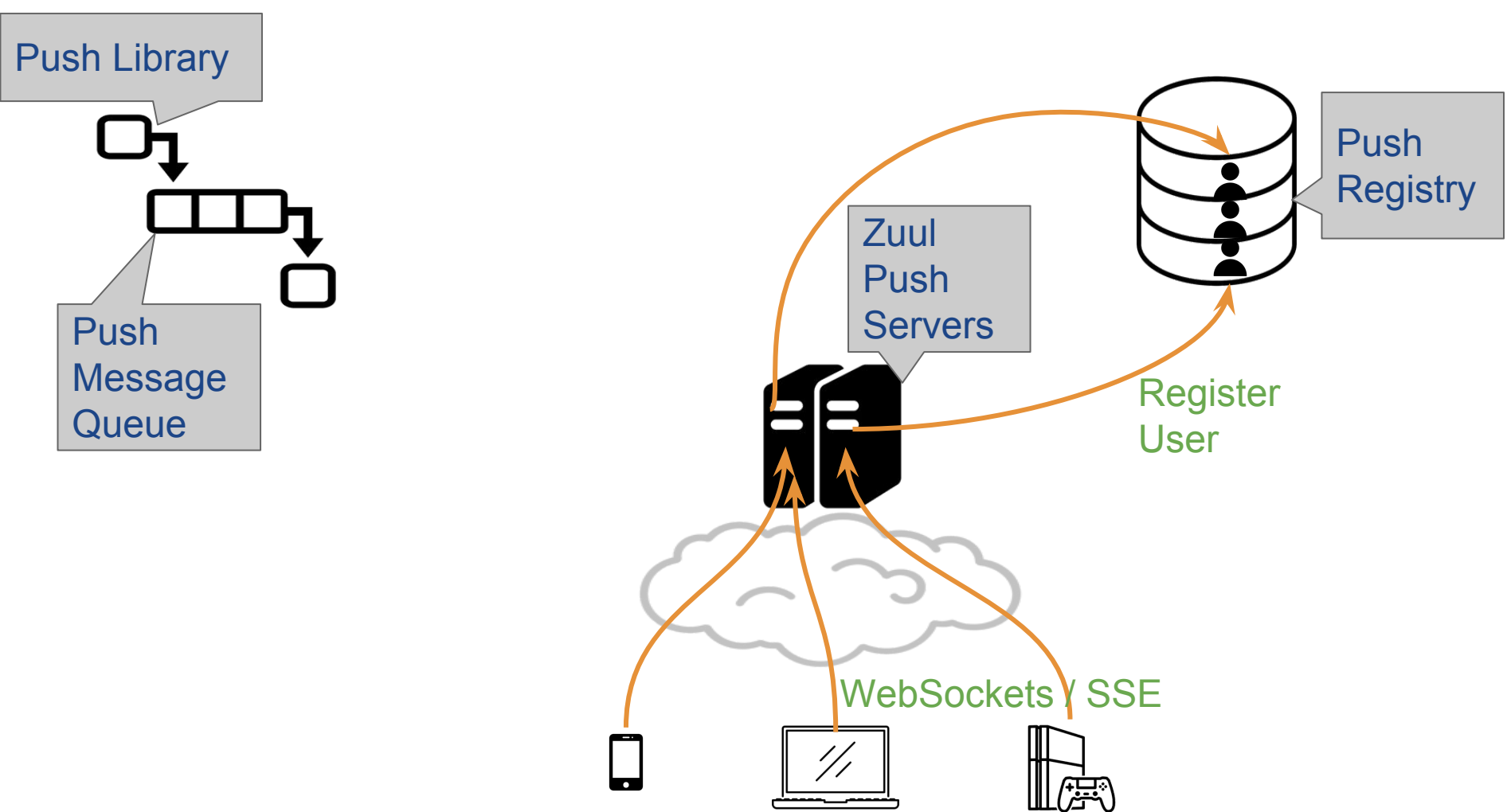
Zuul
Push
Servers

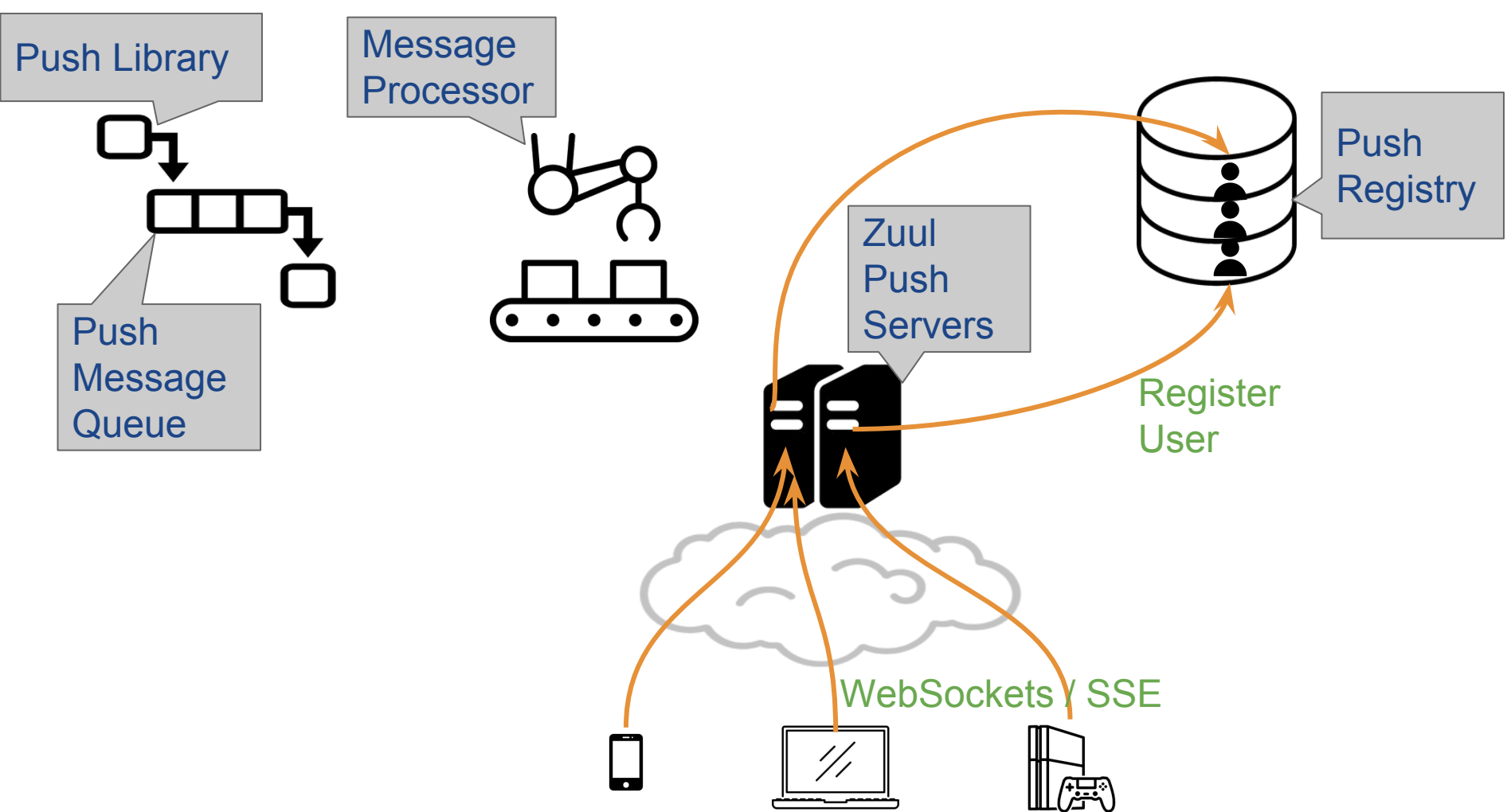


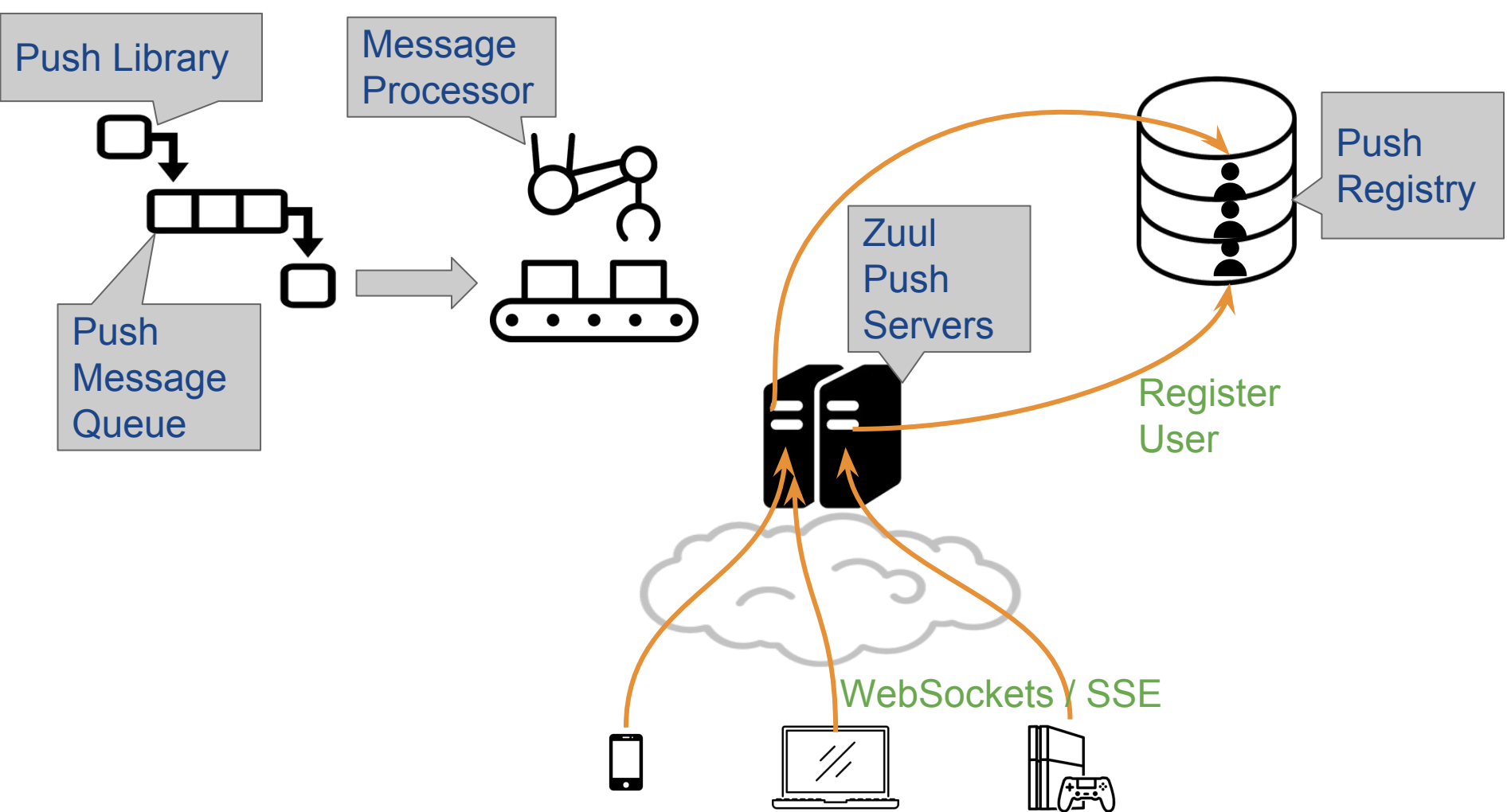


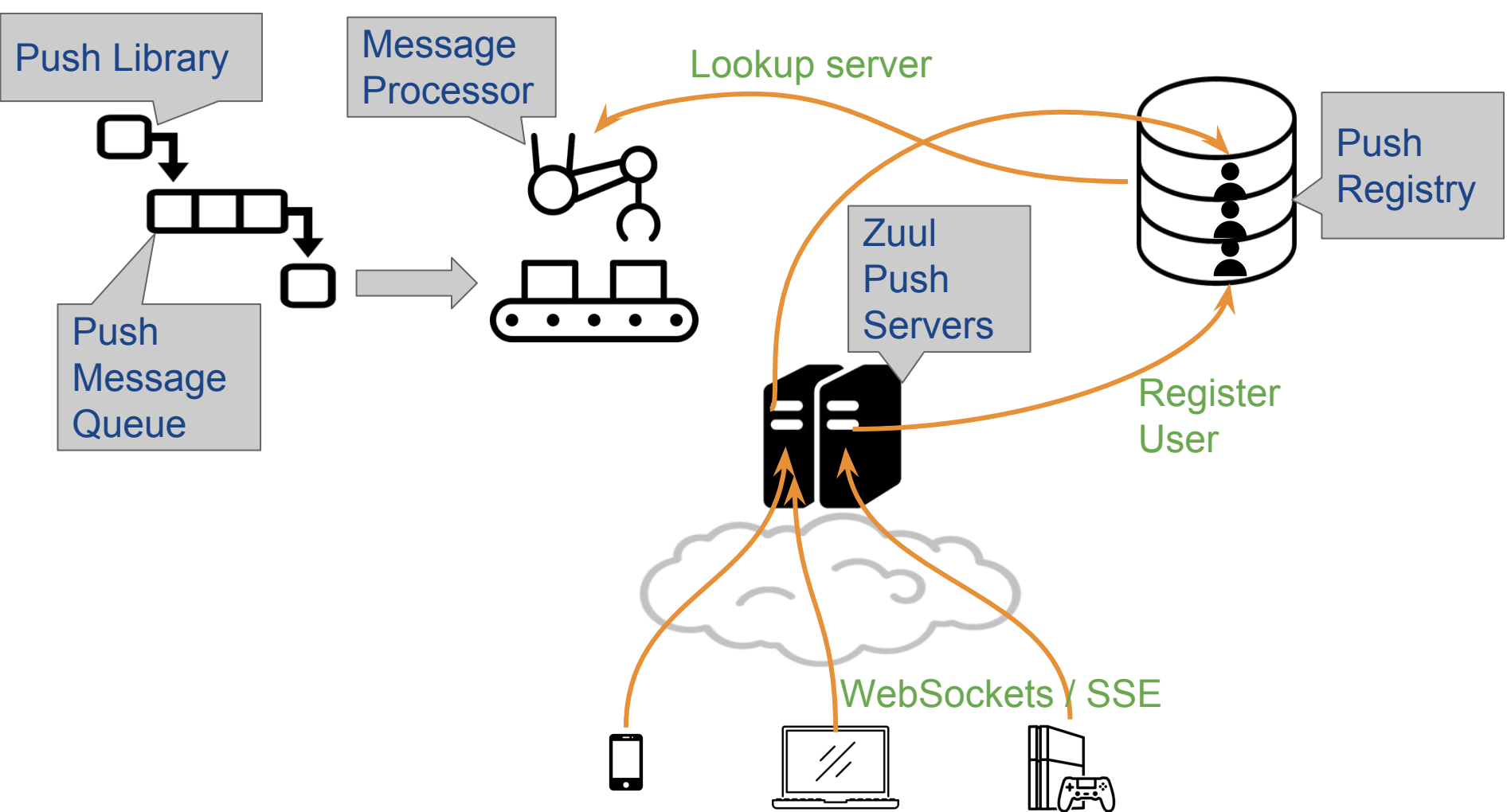


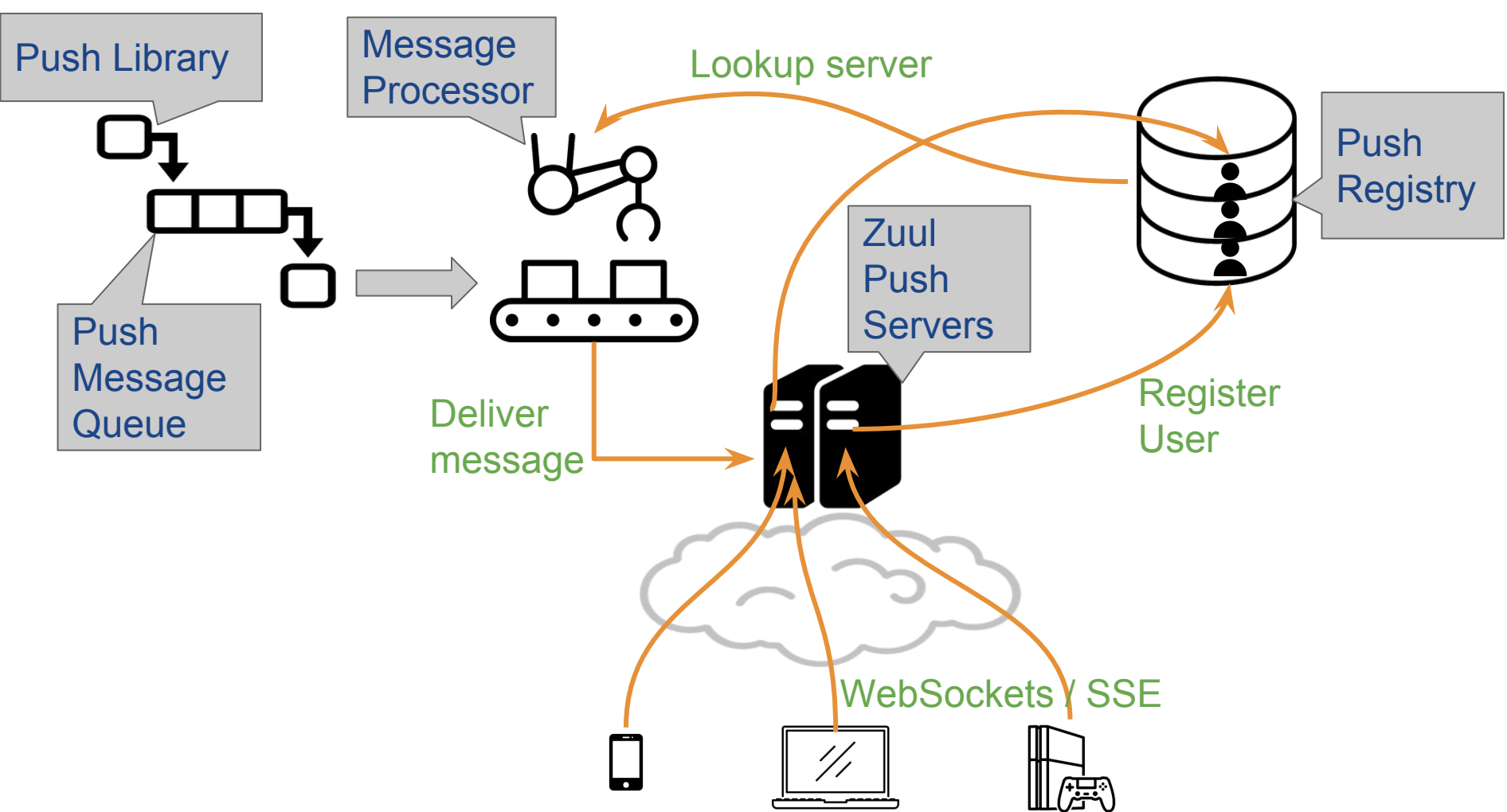










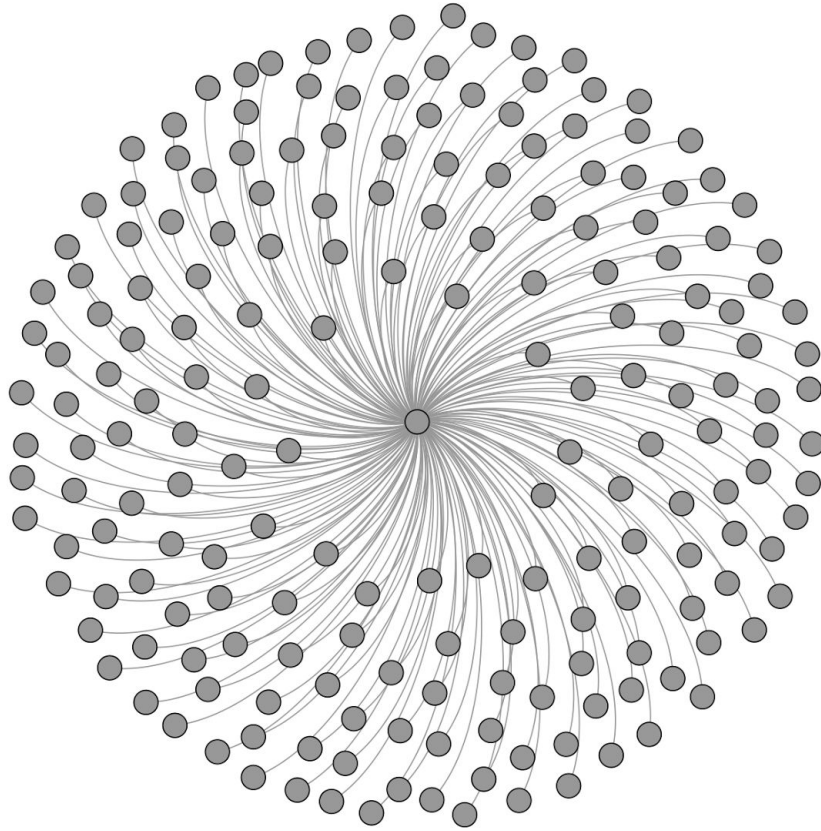


Zuul Push server



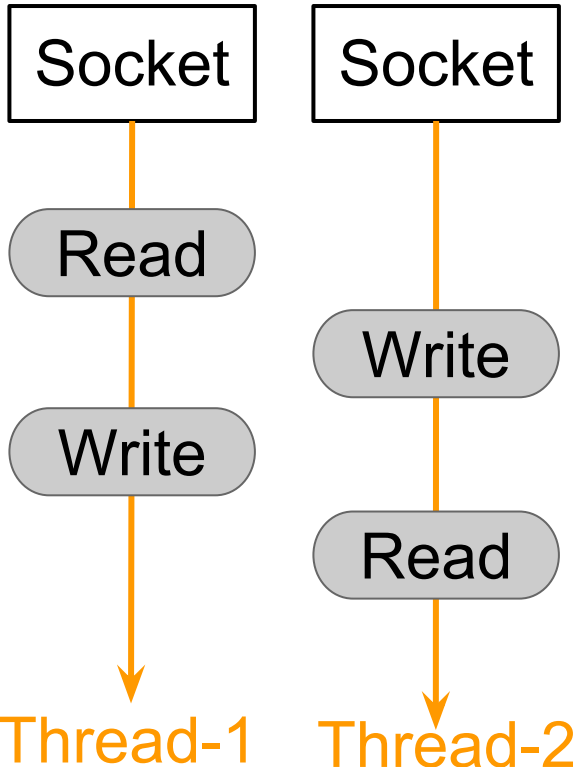
| Handling millions of
persistent connections

NETFLIX

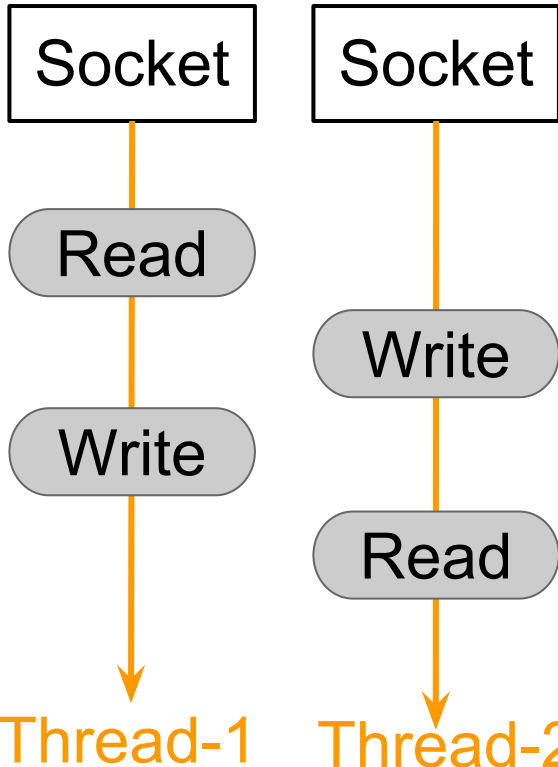


C10K challenge

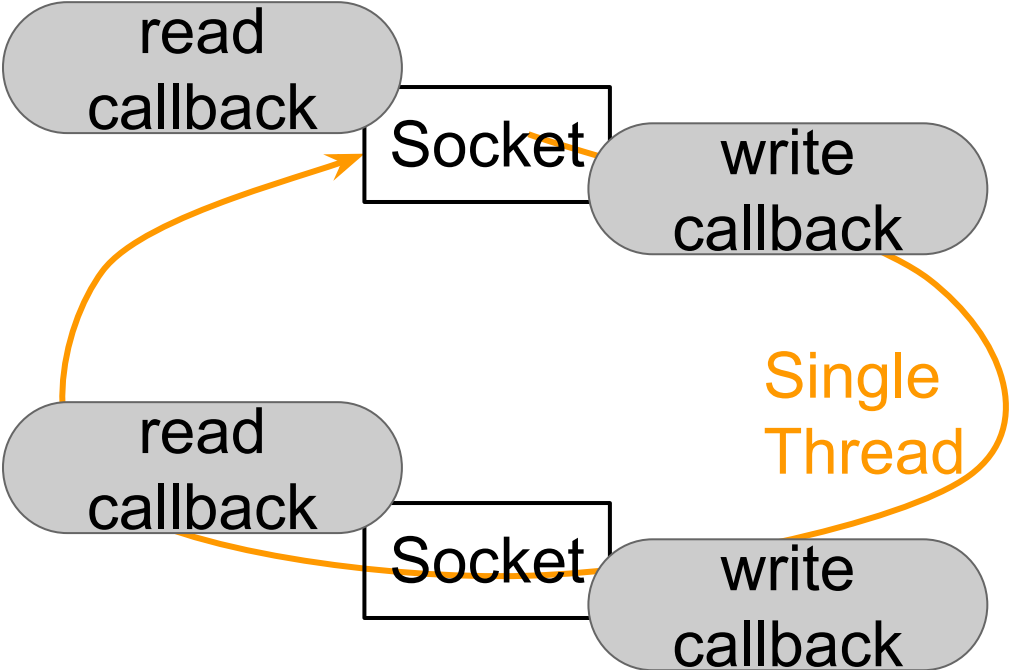
Thread per Connection



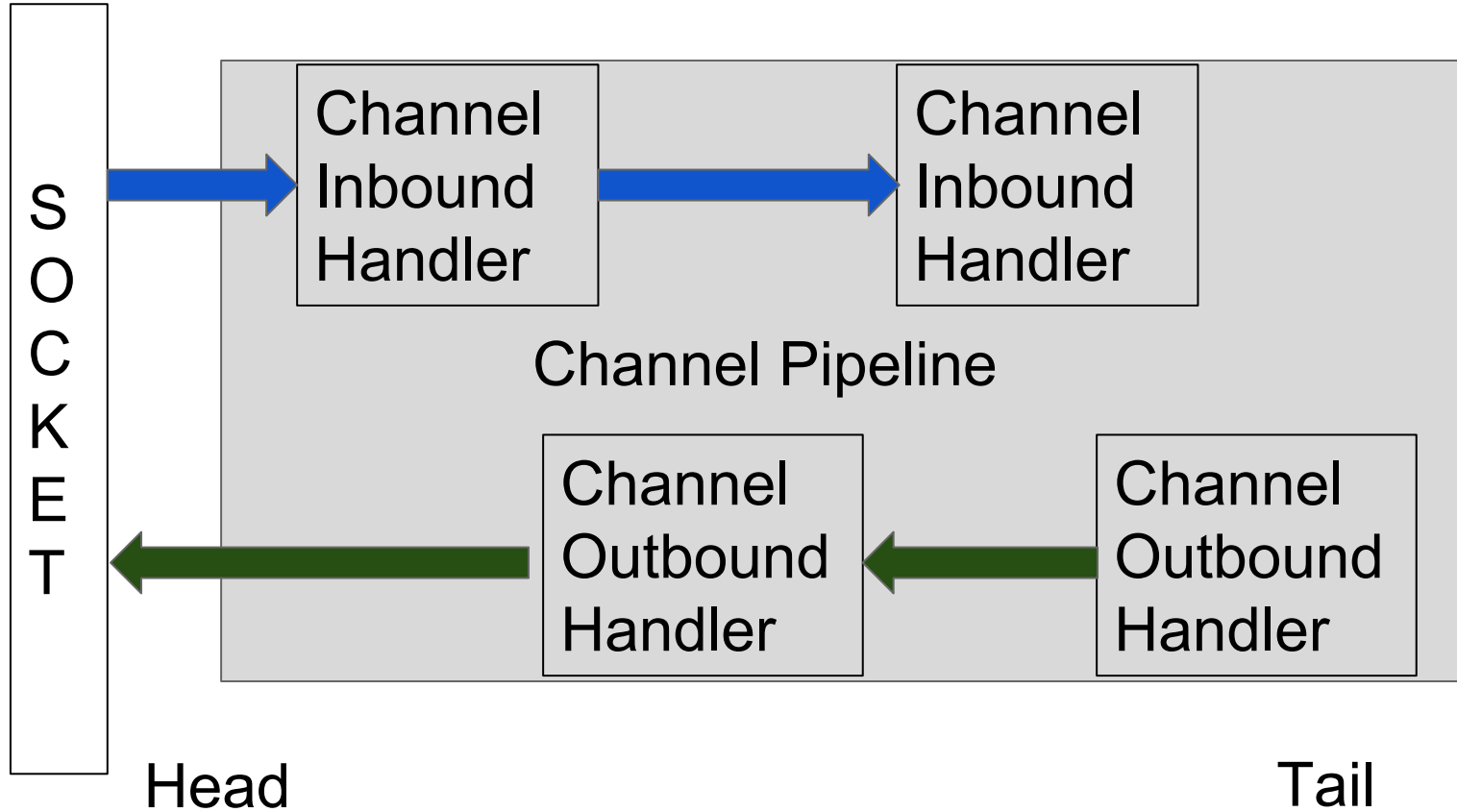
Thread per Connection



Async I/O

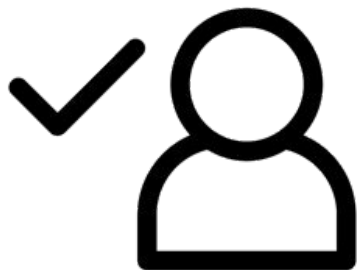


Netty



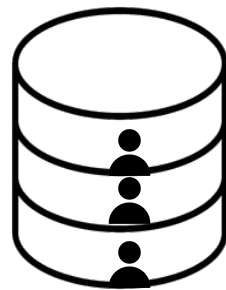
```
protected void addPushHandlers(ChannelPipeline pl) {
    pl.addLast(new HttpServerCodec());
    pl.addLast(new HttpObjectAggregator());
    pl.addLast(getPushAuthHandler());
    pl.addLast(new WebSocketServerCompressionHandler());
    pl.addLast(new WebSocketServerProtocolHandler());
    pl.addLast(getPushRegistrationHandler());
}
```

Plug in your custom authentication policy



Authenticate by Cookies, JWT
or any other custom scheme

Push Registry



Tracking clients' connection
Metadata in real-time

NETFLIX

```
public class MyRegistration extends PushRegistrationHandler {
    @Override
    protected void registerClient(
        ChannelHandlerContext ctx,
        PushUserAuth auth,
        PushConnection conn,
        PushConnectionRegistry registry) {

        super.registerClient(ctx, authEvent, conn, registry);
        ctx.executor().submit(() -> storeInRedis(auth));
    }
}
```

Push registry features checklist



Push registry features checklist



- Low read latency

Push registry features checklist



- Low read latency
- Record expiry

Push registry features checklist



- Low read latency
- Record expiry
- Sharding

Push registry features checklist



- Low read latency
- Record expiry
- Sharding
- Replication



What we use



<https://github.com/Netflix/dynomite>

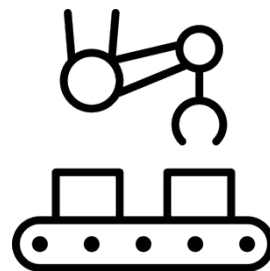
Redis

- + Auto-sharding
- + Read/Write quorum
- + Cross-region replication

Dynomite

Message Processing

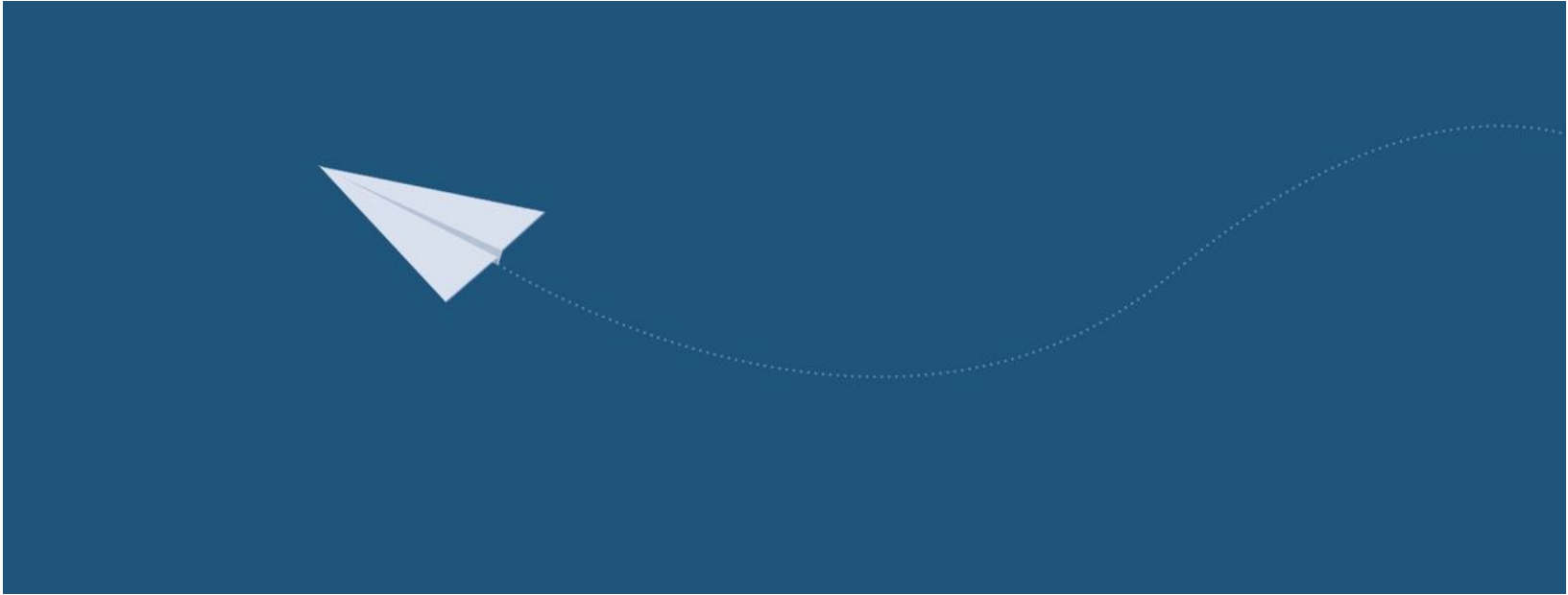
| Queue, Route
| Deliver



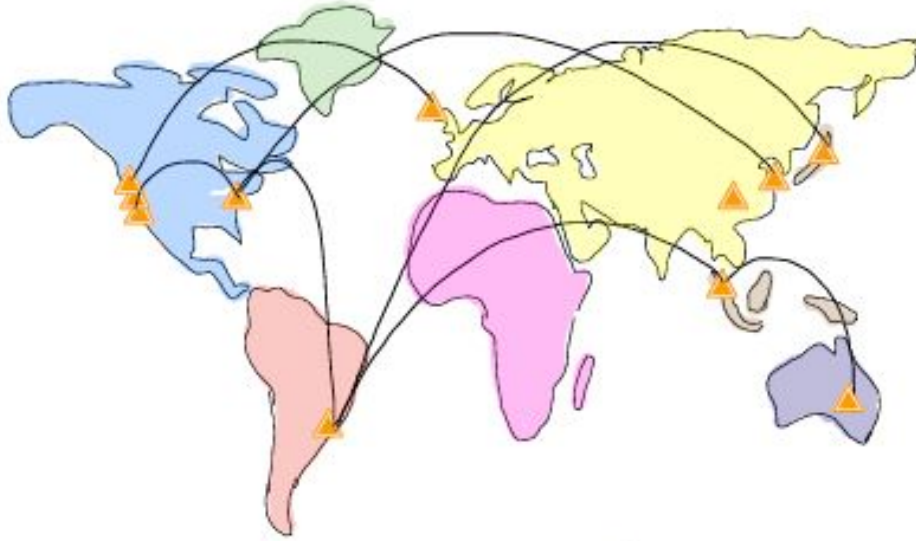
NETFLIX



We use Kafka message queues to decouple message senders from receivers



Fire and Forget

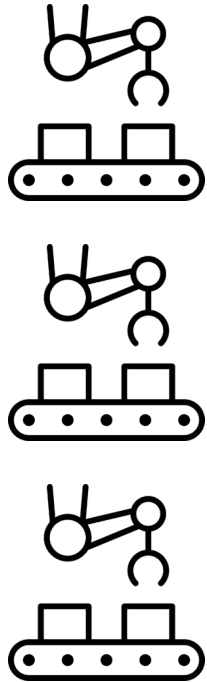


Cross-region Replication



PRIORITY

Different queues for
different priorities



We run multiple message processor instances in parallel to scale our message processing throughput.

Operating Zuul Push

Different than REST of them



Persistent connections make Zuul Push server stateful

Long lived stable connections

Persistent connections make Zuul Push server stateful

Long lived stable connections

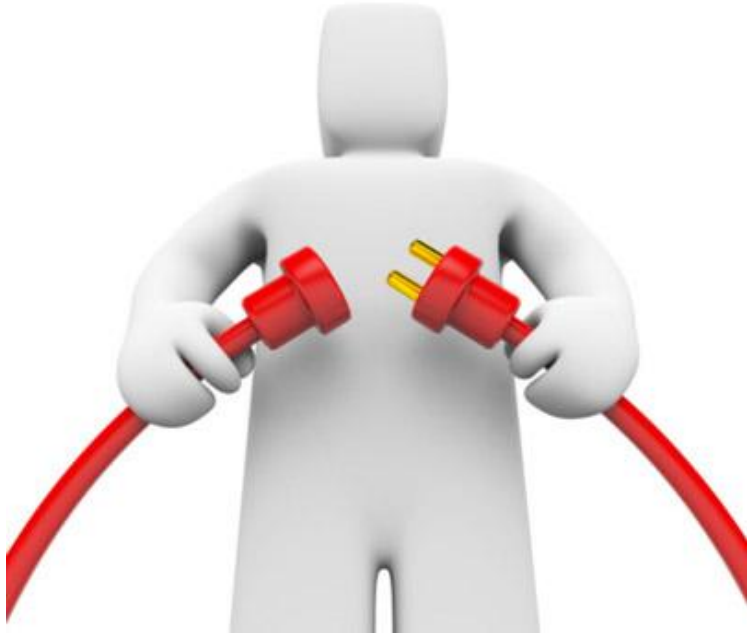
- Great for client efficiency

Persistent connections make Zuul Push server stateful

Long lived stable connections

- Great for client efficiency
- Terrible for quick deploy/rollback

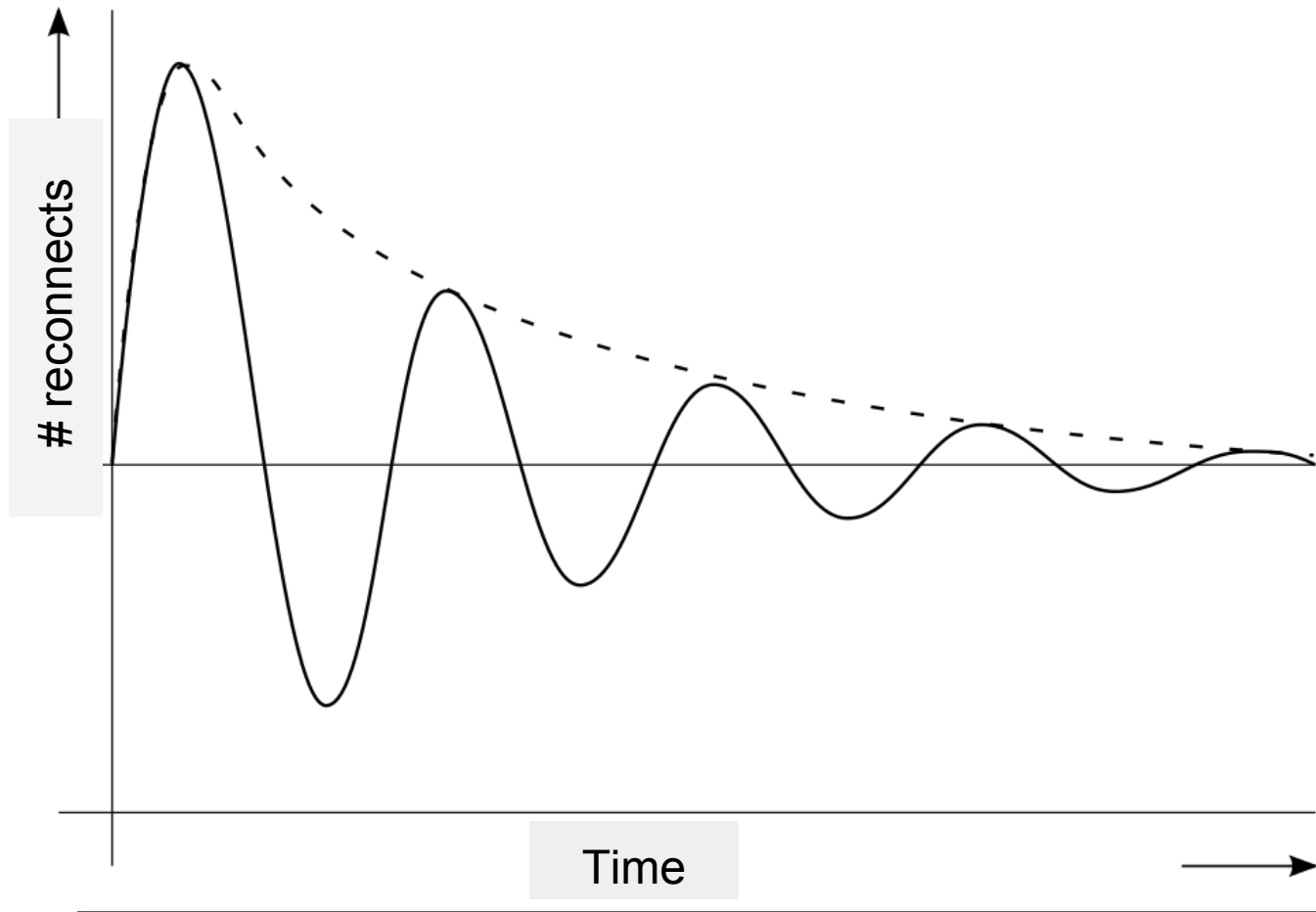
If you love your clients set them free...



Tear down connections
periodically



Randomize each
connection's lifetime



Effect of
randomizing
connection
lifetime on
reconnect peaks

EXTRA CREDIT

Ask client to close its connection.

How to optimize push server



Most connections
are idle!

BIG Server, tons of connections



```
ulimit -n 262144
```

```
net.ipv4.tcp_rmem="4096 87380  
16777216"
```

```
net.ipv4.tcp_wmem="4096 87380  
16777216"
```



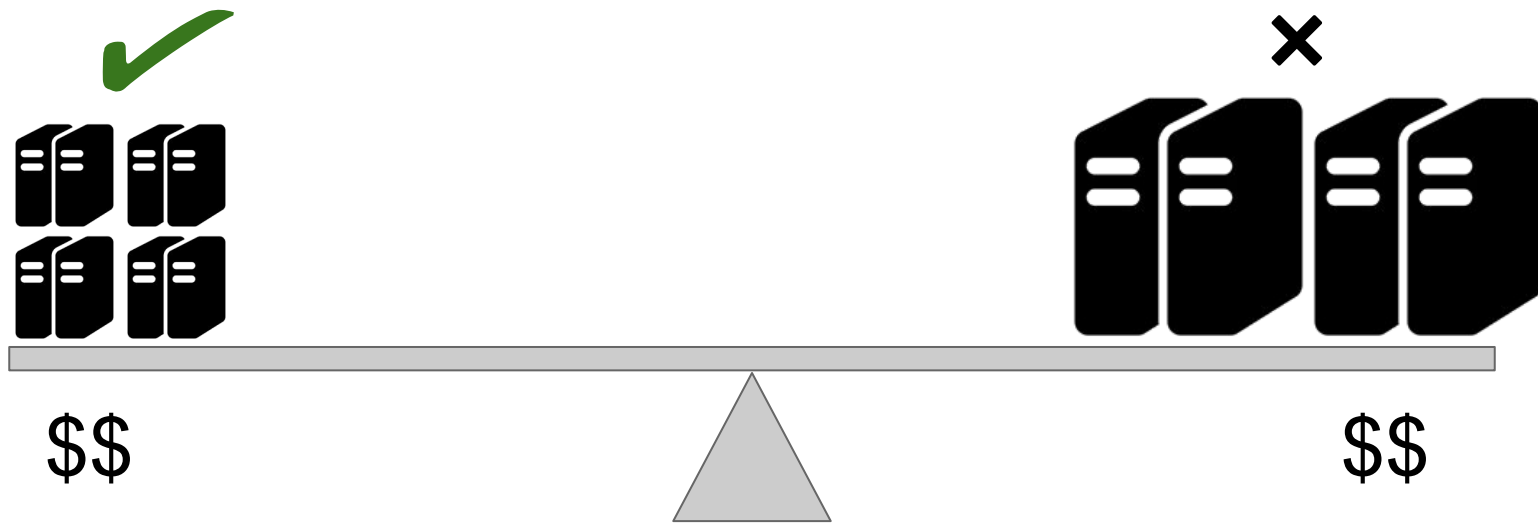
© 2015 P. G. H. S.



Goldilocks strategy



Optimize for cost, NOT instance count



How to auto-scale?

How to auto-scale?

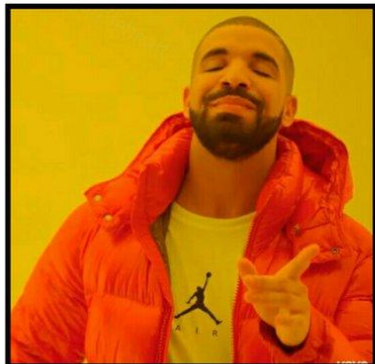


RPS? CPU??

How to auto-scale?



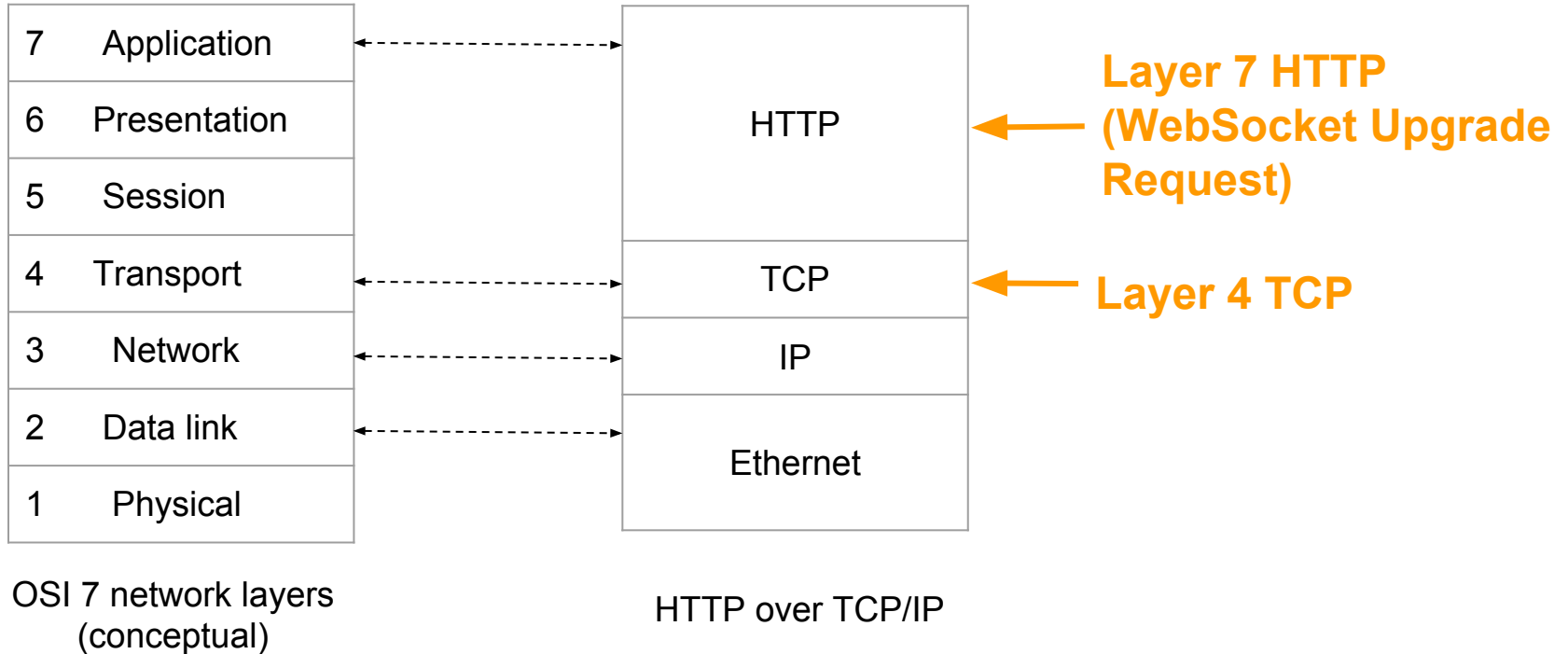
RPS? CPU??



Open
Connections

**Amazon Elastic Load Balancers cannot proxy
WebSockets.**

Solution - Run ELB as a TCP load balancer



Managing push cluster - a quick recap

- Recycle connections after tens of minutes



Managing push cluster - a quick recap

- Recycle connections after tens of minutes
- Randomize each connection's lifetime



Managing push cluster - a quick recap

- Recycle connections after tens of minutes
- Randomize connection's lifetime
- More number of smaller servers >> few BIG servers



Managing push cluster - a quick recap

- Recycle connections after tens of minutes
- Randomize connection's lifetime
- More number of smaller servers >> few BIG servers
- Auto-scale on number of open connections per box



Managing push cluster - a quick recap

- Recycle connections after tens of minutes
- Randomize connection's lifetime
- More number of smaller servers >> few BIG servers
- Auto-scale on number of open connections per box
- WebSocket aware vs TCP load balancer



**If you build it,
They will push**





On-demand diagnostics



Remote recovery



User messaging

WHAT WILL YOU



USE IT FOR?

Call to action

PULL!

PULL!



<https://github.com/Netflix/zuul>

In conclusion, push can make you

**In conclusion, push can make you
rich (in functionality),**

**In conclusion, push can make you
rich (in functionality),
thin (by getting rid of polling)**

**In conclusion, push can make you
rich (in functionality),
thin (by getting rid of polling)
and happy!**

Thank you.



Questions?

Susheel Aroskar

Senior Software Engineer

Cloud Gateway

saroskar@netflix.com



github.com/raksoras

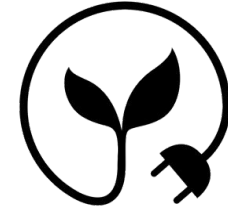


[@susheelaroskar](https://twitter.com/susheelaroskar)

NETFLIX



Rich,
exciting
Apps

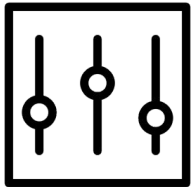


More
efficient
systems

Battle tested



Zuul Push



Easy to
customize



Easy to
operate