NodeJS & Shell

@luisfaustino

github.com/dublx/nsq-nodejs-demo

@Luisfaustino

- Software Eng,
  @openjawtech.com

  - NodeJs, NSQ.io, AngularJS
    SQL & NoSQL

  - Etcd, Docker, AWS, Jenkins

- Photographer
  @Luis-faustino.com

- Living in Dublin, IE

- From Lisbon, PT

screen.pipe(agenda());

- Queues

- NSQ.io

- Demo

- Q&A

Queue

|kjuː|

noun

1 chiefly brit. a line or sequence of people or vehicles awaiting their turn to be attended to or to proceed.

2 computing: a list of data items, commands, etc., stored so as to be retrievable in a definite order, usually the order of insertion.

3 archaic: a plait of hair worn at the back.

Queues

- Logically unified system used to emit and receive data

- Natural fit for distributed architectures

- Allow realtime & batch/offline

- Easy decoupling of code from protocol and specs
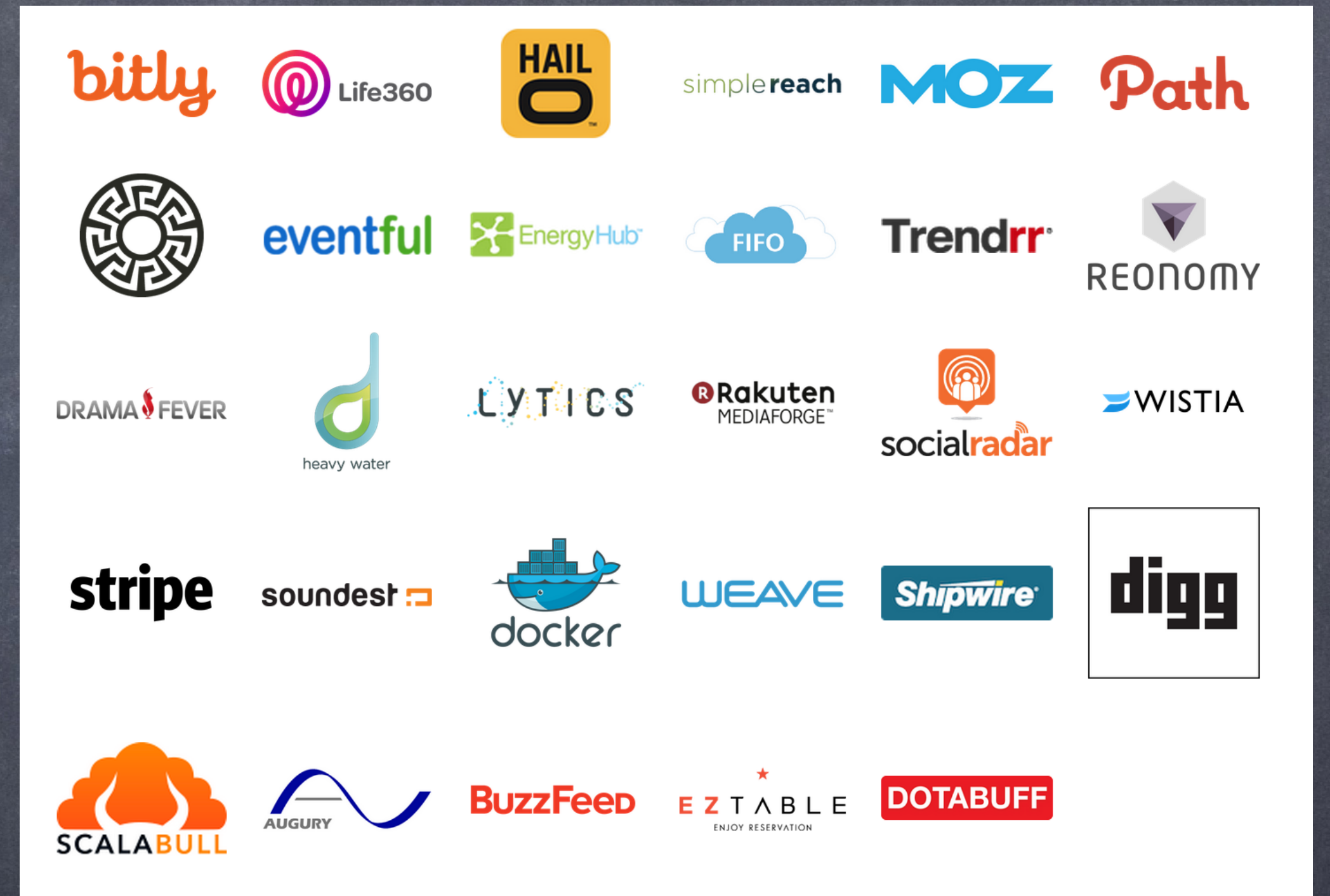
- Allow plug&play[tm] workflows

Common Queue Patterns

- Unicast (producer to one consumer)

- Broadcast (producer to all consumers... PubSub™)

- Multicast (producer to groups of consumers)

NSQ.io

github.com/bitly/nsq

nsq-users@googlegroups.com
#nsq on freenode
@nsqio

NSQ.io

- realtime distributed messaging platform

- designed to operate at scale

- can handle billions msg/day

- distributed and decentralised topologies

- no SPOF, fault tolerant, HA

- reliable message delivery guarantee

Libraries & tooling

- All main languages, official NodeJS, Go and Python

- Nodejs: npm install nsqjs

- nsqlookupd: topic & producers/consumers discovery

- nsqd: reading/writing messages

- nsq_to_*: shell utilities

- nsqadmin (GUI)

NSQ binaries

```
$ ls -lh /opt/nsq/bin/

-rwxr-xr-x 6.8M nsq_pubsub
-rwxr-xr-x 6.5M nsq_stat
-rwxr-xr-x 6.8M nsq_tail
-rwxr-xr-x 6.9M nsq_to_file
-rwxr-xr-x 6.9M nsq_to_http
-rwxr-xr-x 7.0M nsq_to_nsq
-rwxr-xr-x 8.3M nsqadmin
-rwxr-xr-x 8.8M nsqd
-rwxr-xr-x 8.0M nsqlookupd
-rwxr-xr-x 6.7M to_nsq
```

NSQ start

```
$ nsqlookupd &> nsqlookupd.log &

$ nsqd --lookupd-tcp-address=127.0.0.1:4160 &> nsqd.log &

$ nsqadmin --lookupd-http-address=127.0.0.1:4161 &> nsqadmin.log &
```

NSQ in Nodejs

```
//subscribe to a NSQ topic

var nsq = require('nsqjs');

var reader = new nsq.Reader('topic', 'channel', {
  lookupdHTTPAddresses: '127.0.0.1:4161'
});

reader.connect();

reader.on('message', function (msg) {
  console.log('Received message [%s]: %s', msg.id, msg.body.toString());
  msg.finish();
});
```

NSQ in Nodejs

```javascript
//writes a message to a NSQ topic

var nsq = require('nsqjs');
var writer = new nsq.Writer('127.0.0.1', 4150);


writer.connect();


writer.on('ready', function () {
    writer.publish('topic', 'message', function (err) {
      if (err) { return console.error(err.message); }
      writer.close();
    });
});


writer.on('closed', function () {
  console.log('Writer closed');
});
```

NSQ in Shell

```
#stream to nsq
while read x
do
    echo $x | to_nsq —nsqd-tcp-address=127.0.0.1:4150 --topic=stdin
done

#nsq to http
nsq_to_http -lookupd-http-address=127.0.0.1:4161 -topic=stdin -content-type="text/plain" -post="http://127.0.0.1:3000/stdin"

#tail
nsq_tail -lookupd-http-address=127.0.0.1:4161 -topic=stdin

#persist
nsq_to_file -lookupd-http-address=127.0.0.1:4161 -topic=stdin -output-dir="."

#cross Datacenter / AZ
nsq_to_nsq -destination-nsqd-tcp-address=dc2:4160 -destination-topic=stdin
```

NSQ in Shell

```
#post a message to nsqd http endpoint
curl -d 'message' 'http://127.0.0.1:4151/pub?topic=thetopic'

#post multiples messages to nsqd http endpoint
curl -d 'message1\nmessage2\nmessage3' 'http://127.0.0.1:4151/mpub?topic=thetopic'

# other nsqd http endpoints:
/topic/
   /create  /delete  /empty   /pause  /unpause


/channel/
   /create  /delete  /empty   /pause  /unpause


/stats
/ping
/info
```

NSQ Admin

NSQ stdin

localhost:4171/topic/stdin

| NSQ | Streams | Nodes | Counter | Lookup | NSQ on github | v0.3.2 |

Streams / stdin

Topic: **stdin**

Empty Queue   **Delete Topic**   **Pause Topic**

Topic Message Queue

NSQd Host	Depth	Memory + Disk	Messages	Channels
✗ luis.local:4151	0	0 + 0	16	2
Total:	0	0 + 0	16	2

Channel Message Queues

Channel	Depth	Memory + Disk	In-Flight	Deferred	Requeued	Timed Out	Messages	Connections
node	0	0 + 0	0	0	0	0	4	1
nsq_to_file	0	0 + 0	0	0	0	0	13	1

DEMO

read stdin → pipe → to_nsq → PUB topic A → nsqd

nsq_to_http ← SUB topic A ← nsqd

nodeJS ← POST /topic ← nsq_to_http

ExpressJS → PUB topic B → nsqd

nsq_tail ← SUB topic B ← nsqd
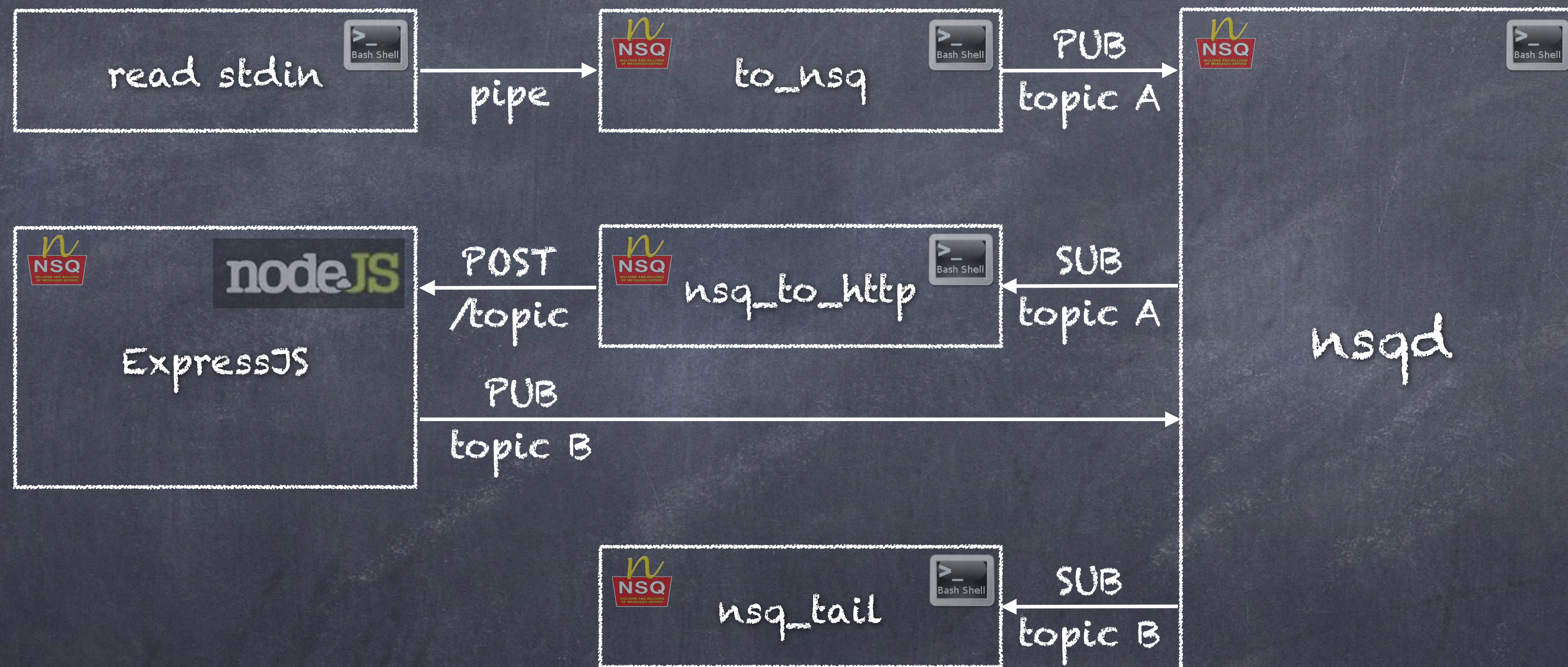
Q&A / References

- speakerdeck.com/snakes/spray-some-nsq-on-it

- github.com/bitly/nsq