PUSH TECHNOLOGY

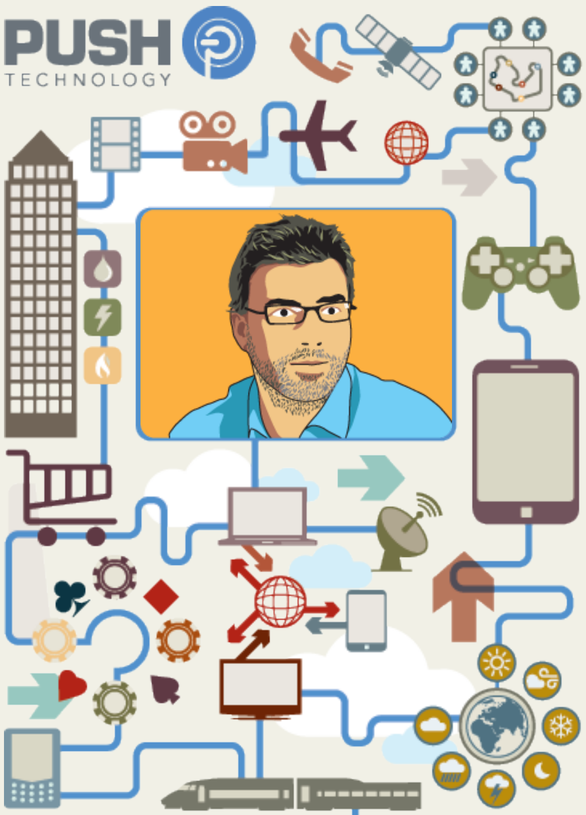
QCON
London 2013

Copyright Push Technology 2012

My ‘number’

#6752

Copyright Push Technology 2012

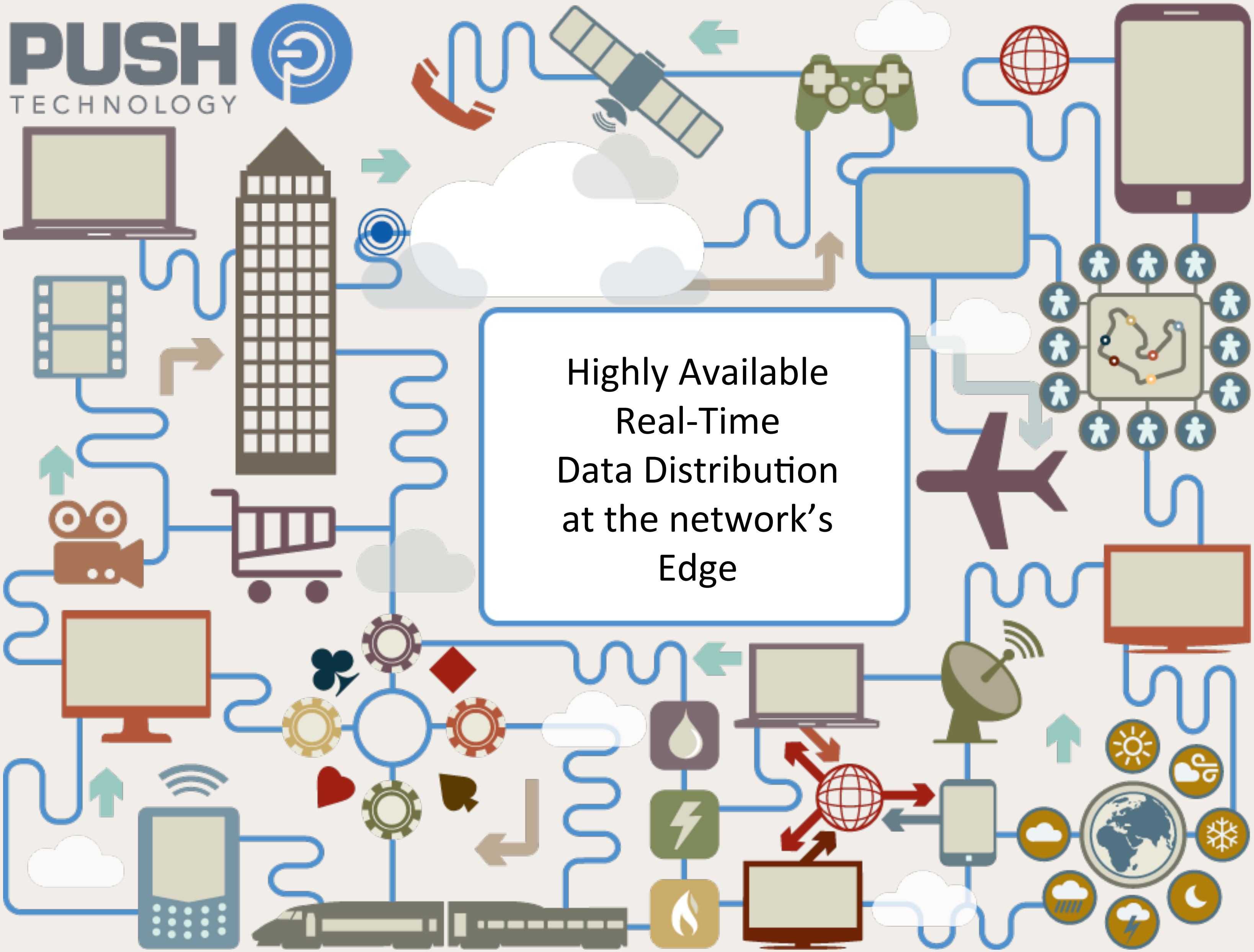


- Distributed Systems / HPC guy.
- **Chief Scientist** :- at **Push Technology**
- Alumnus of :-

  Motorola, IONA, Betfair, JPMC, StreamBase.
- School: Trinity College Dublin.
  - BA (Mod). Comp. Sci.+
  - M.Sc. Networks & Distributed Systems
- Responds to: Guinness, Whisky

About me?

Copyright Push Technology 2012



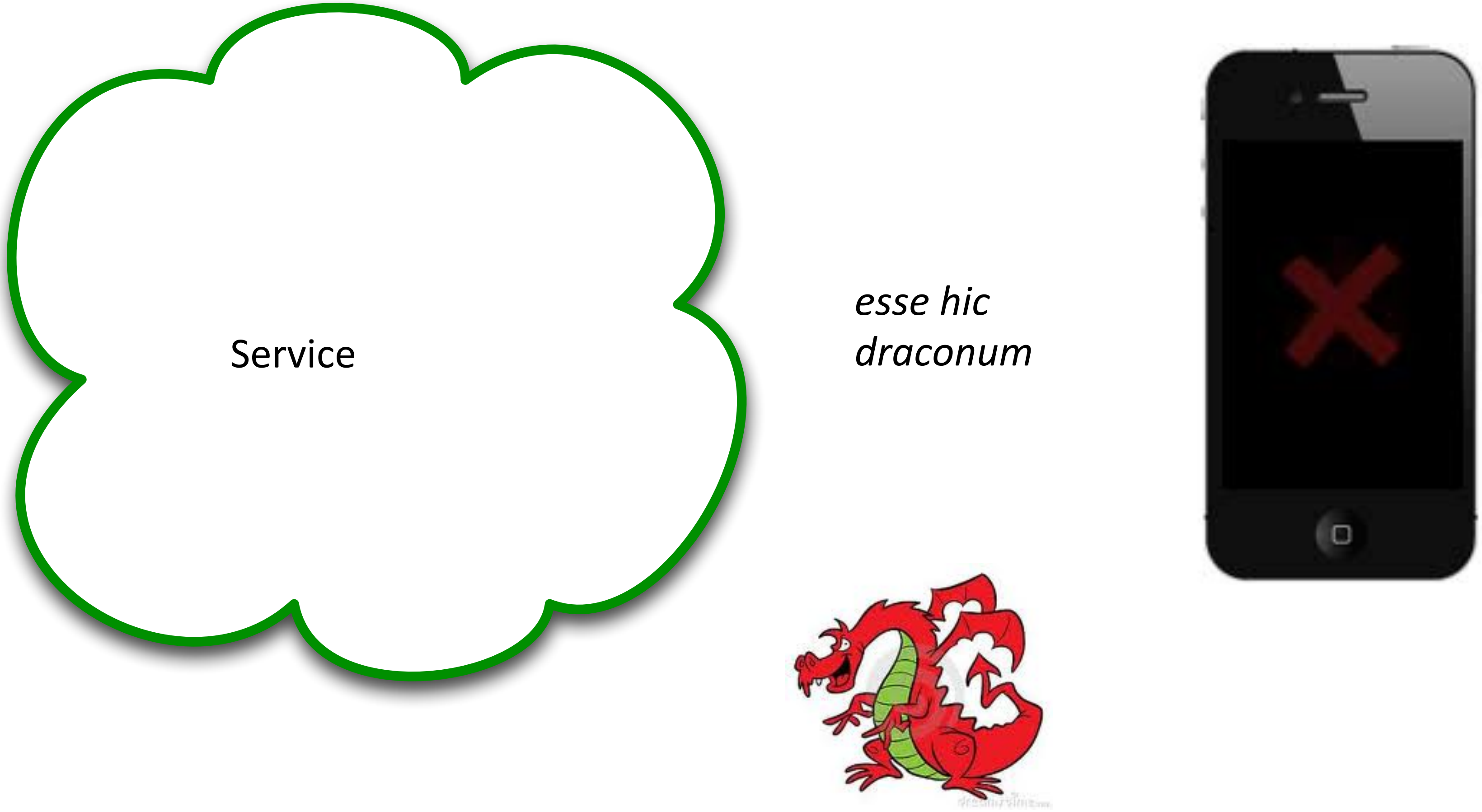
Highly Available Real-Time Data Distribution at the network's Edge

Copyright Push Technology 2012

HA. Black swans & Pigfish!

Copyright Push Technology 2012

When this …



Service

esse hic draconum

… inevitably happens

Over the Edge, where Fallacies roam

- The network is reliable
- Latency is zero
- Bandwidth is infinite
- There is one administrator
- Topology does not change
- The network is secure
- Transport cost is zero
- The network is homogeneous

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

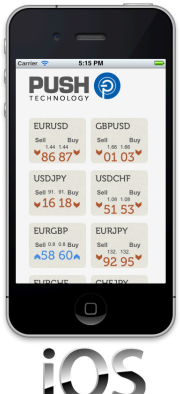
Copyright Push Technology 2012

& the last mile. Dragons there be.

Buffer Bloat

Cell Coverage

Battery!



Carrier 5:15 PM

PUSH TECHNOLOGY

EURUSD Sell 1.44 1.44 Buy 86 87	GBPUSD Sell 1.66 1.66 Buy 01 03
USDJPY Sell 91. 91. Buy 16 18	USDCHF Sell 1.08 1.08 Buy 51 53
EURGBP Sell 0.8 0.8 Buy 58 60	EURJPY Sell 132. 132. Buy 92 95

4:33 PM

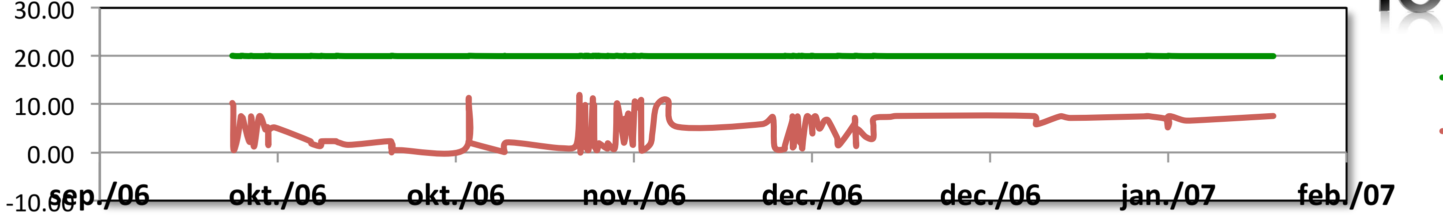
PushToQuote

PUSH TECHNOLOGY

Symbol	Bid Price	Ask Price	Spread	Mid
EURUSD	1.53	1.53	0.01	1.53
GBPUSD	1.73	1.82	0.09	1.78
USDJPY	99.81	98.31	1.5	99.06
USDCHF	1.13	1.12	0.0	1.12
EURGBP	0.91	0.92	0.01	0.91
USDHKD	8.71	8.4	0.31	8.55
USDDKK	5.29	5.04	0.25	5.16
EURNOK	9.16	9.09	0.07	9.13
GBPNZD	2.43	2.36	0.07	2.4
EURSEK	10.88	11.11	0.24	10.99
EURJPY	136.89	139.89	3.0	138.39
EURCHF	1.57	1.64	0.07	1.6
USDZAR	7.71	7.75	0.04	7.73
USDSEK	7.09	7.16	0.07	7.12
GBPCHF	1.74	1.8	0.05	1.77
GBPJPY	165.51	153.87	11.64	159.69
USDSGD	1.5	1.56	0.06	1.53
CHFJPY	93.77	97.97	4.2	95.87
GBPCAD	1.83	2.07	0.24	1.95
USDNOK	5.79	5.74	0.15	5.77
USDCAD	1.82	1.81	0.01	1.81
AUDUSD	0.88	0.9	0.02	0.89

Connected to dpt://192.168.53.17:8080

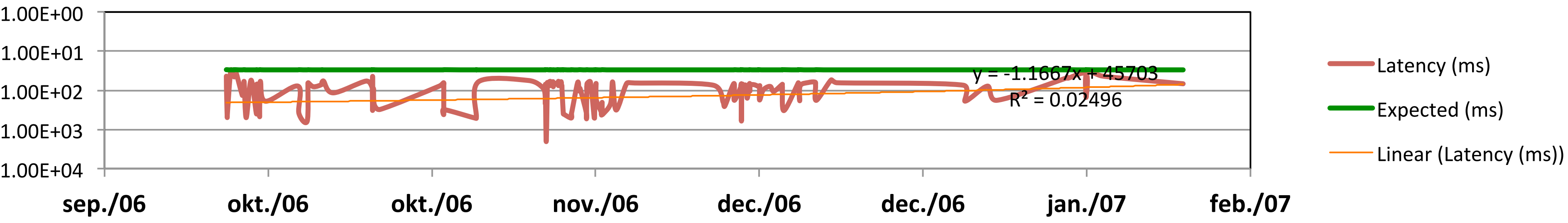
Download Bandwidth



🟩 Expected (Mbps)

🟥 Download (Mbps)

Latency



Copyright Push Technology 2012

Distributed Systems

"A **distributed system** is one in which the **failure** of a computer **you didn't even know existed** can **render your computer** **unusable**"

- Leslie Lamport

Copyright Push Technology 2012

High Availability

“The price of **reliability** is the pursuit of the **utmost simplicity**.”

- Sir Tony Hoare, Turing Award speech, 1980

“This is the unfortunate truth: **Simplicity** is **Hard**.”

- Mosely & Marks, Out of the Tar Pit, 2006

Copyright Push Technology 2012

High **Experiences ~ Man vs Machine**

Image: http://www.phillymarketinglabs.com/man-vs-machine/

Copyright Push Technology 2012

Machines are 'Easy'

A more unfortunate truth is:

Humans* are Harder.

Allegedly Sergey* can be *Emasculated by Glass*. But, I bet he owns a god phone.

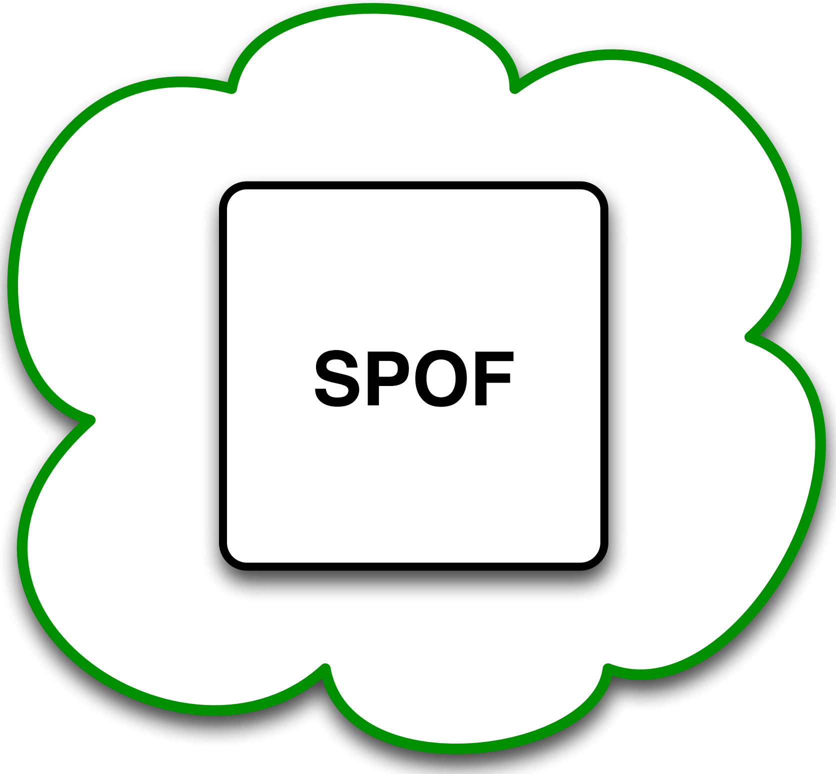
** Humans are grumpy!* ** Colors chosen at random*

Once Upon a time …

SPOF

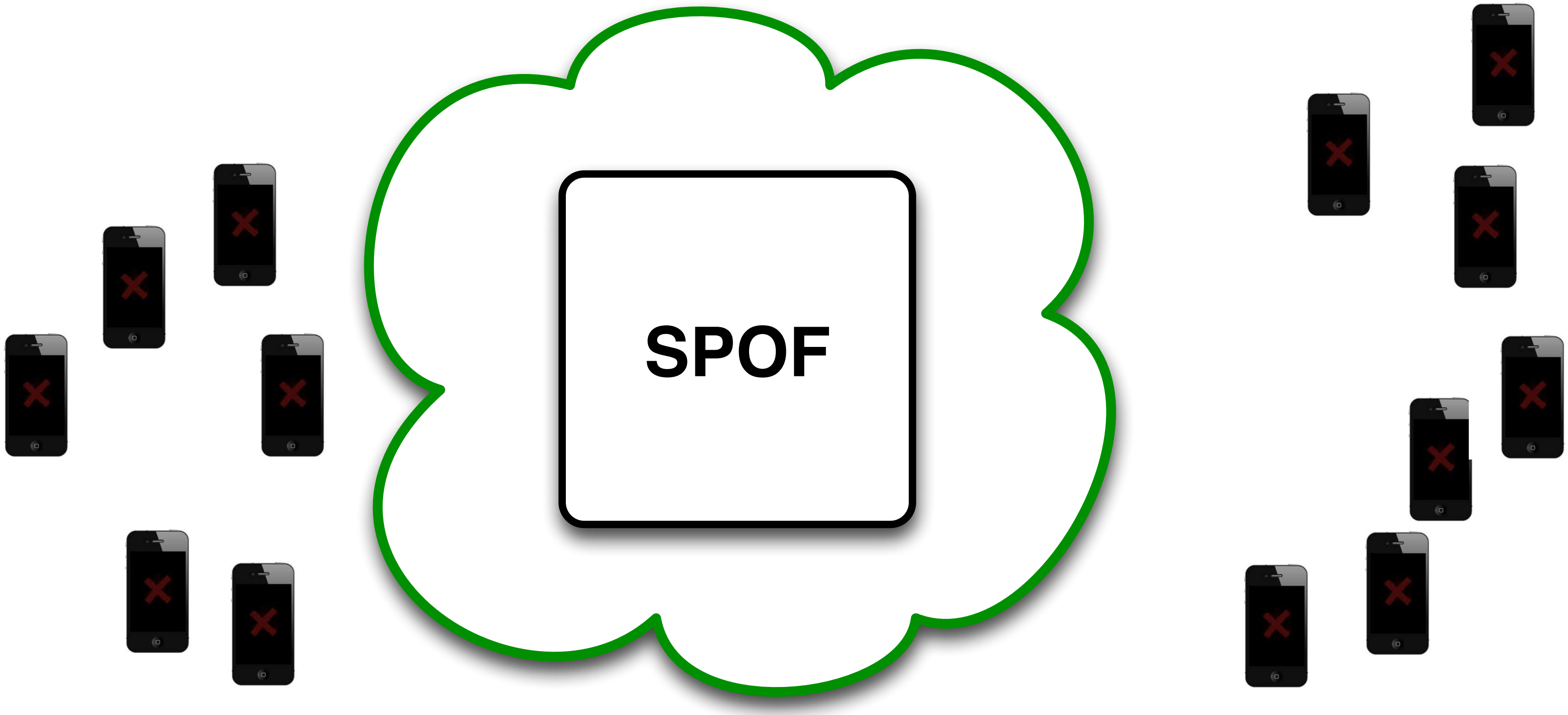
There was a very useful little single point of failure.

It was so successful



It grew and it grew! So we put it in the cloud.

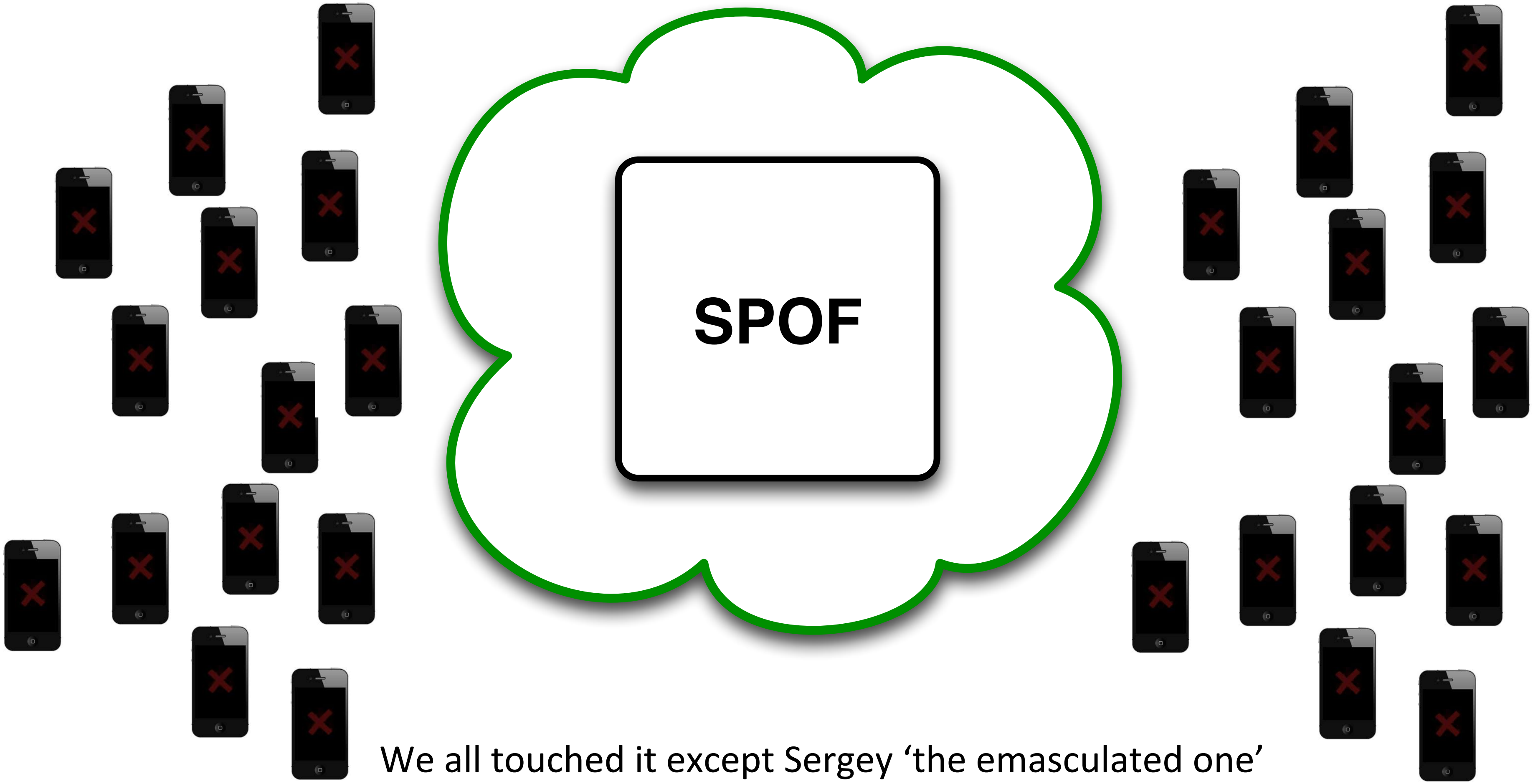
It grew, and it grew …



We touched it with all of our devices.

Copyright Push Technology 2012

See SPOF run!



We all touched it except Sergey 'the emasculated one'

Copyright Push Technology 2012

See SPOF spew ☹

Is there nothing we can do? Esse hic draconum, basically. But, yes we can too …

Copyright Push Technology 2012

Bad things happen …

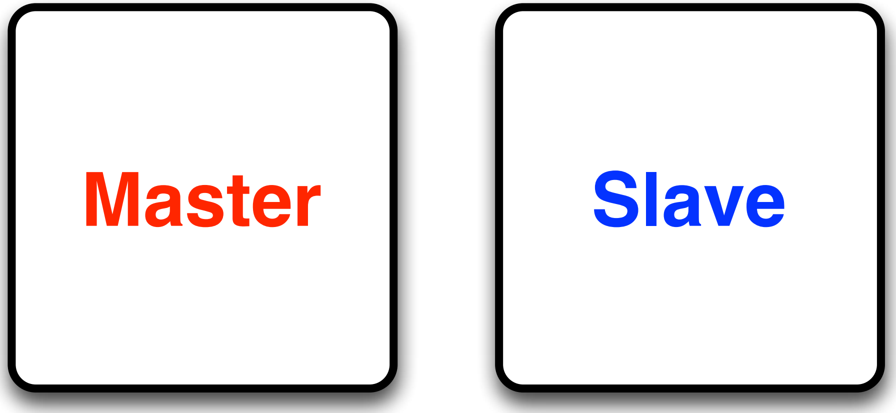
RapGenius
vs
Heroku.

Anyone?
Ferris?

It’s either dyno park this or whale fail in the zoo …

Copyright Push Technology 2012

So we spun up another instance



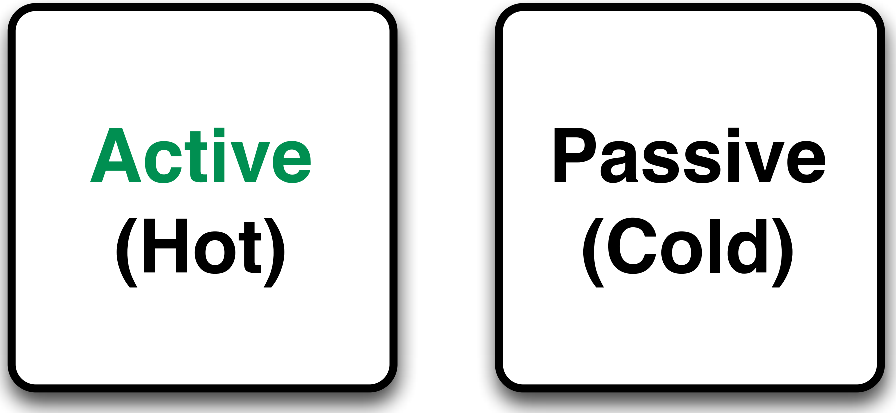
Master

Slave

And master / slave dabbled with us

Copyright Push Technology 2012

Active/Passive – Maybe?



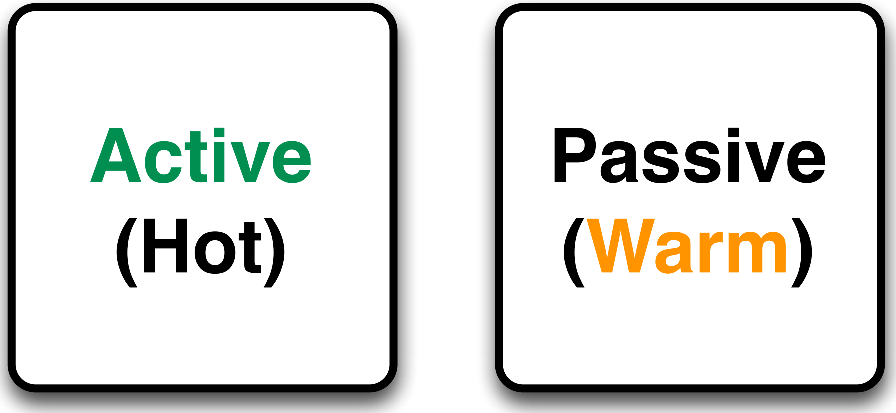
Active
(Hot)

Passive
(Cold)

MTTR relatively high. MTBF relatively low. Ok for
‘nice-to-have’ or ‘too-late-to-cry-now’ facilities.

Copyright Push Technology 2012

Active/Passive – Hmm?



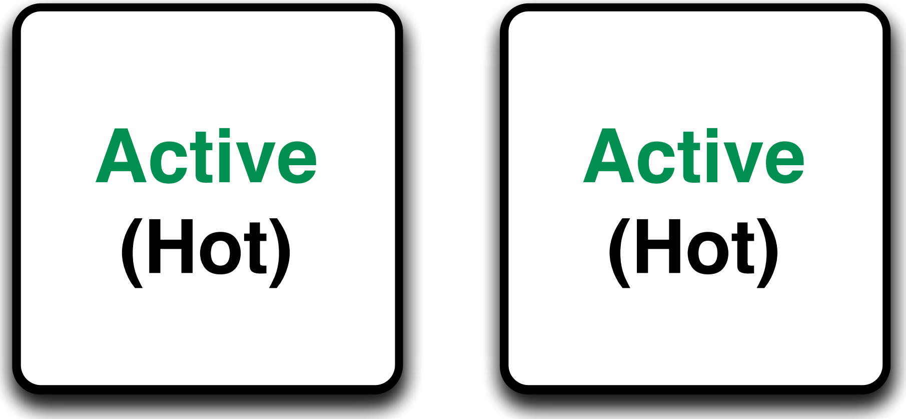
Active
(Hot)

Passive
(Warm)

MTTR slightly better. Can replicate **state** actively for cheaper recovery.

Copyright Push Technology 2012

Active/Active – Ugh…



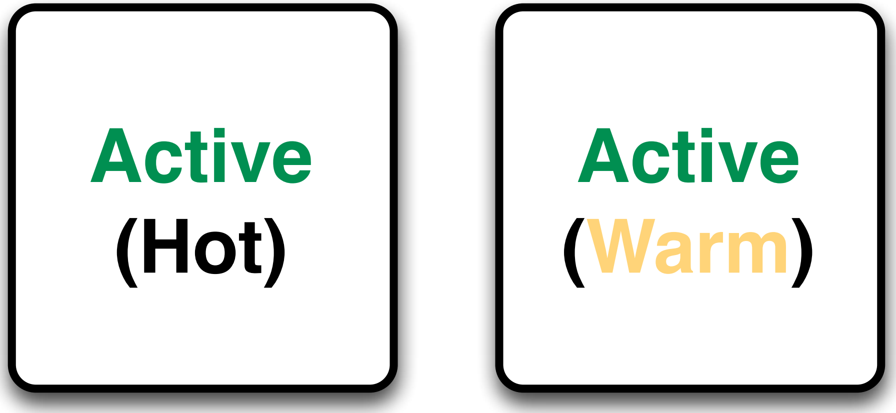
Active
(Hot)

Active
(Hot)

MTTR negligeable. Use 2x the bandwidth. Get ‘half’ the value. Dups

Copyright Push Technology 2012

Active/Active – Ugh…



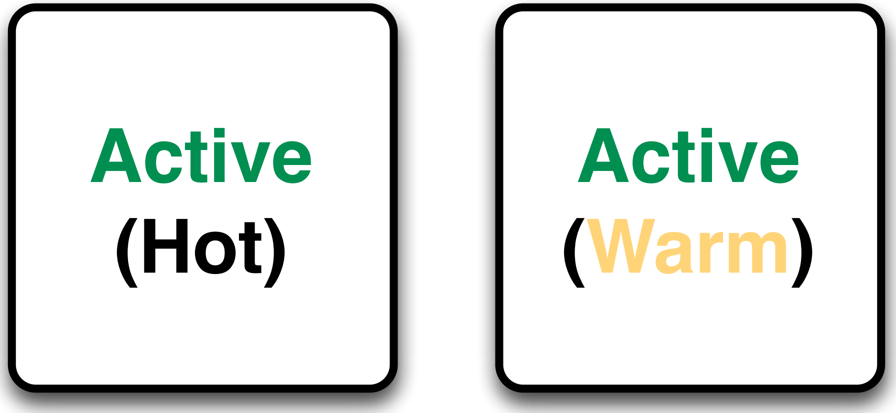
Active
(Hot)

Active
(Warm)

MTTR negligeable. Minify bandwidth usage mostly. Dups

Copyright Push Technology 2012

Active/Active – Ugh…



Active

(Hot)

Active

(Warm)

MTTR negligeable. Minify bandwidth usage mostly. Dups

Copyright Push Technology 2012

Spin up another box?

Active
(Hot)

Speak no evil

Passive
(Cold)

Hear no evil

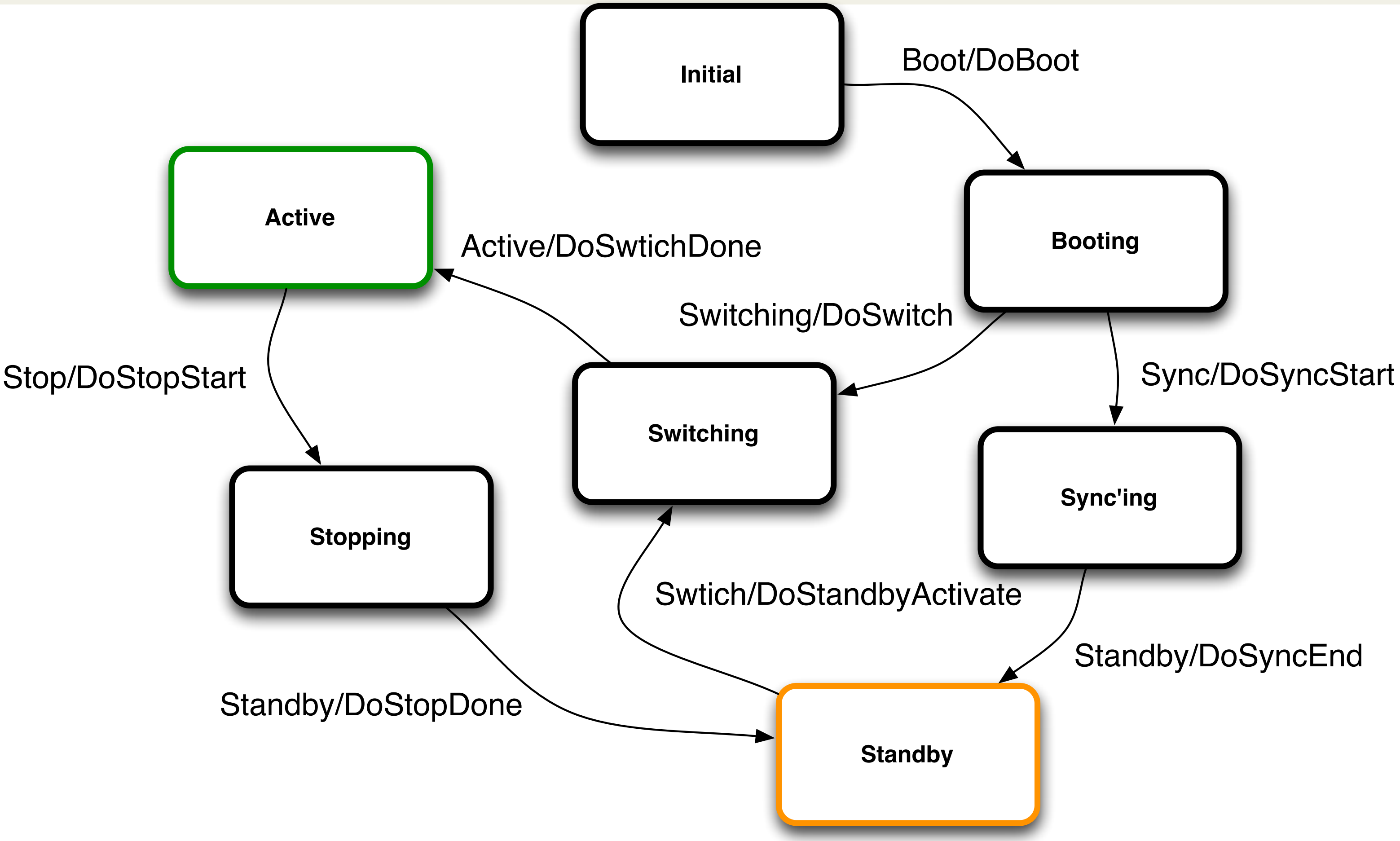
Standby
(Cold)

See no evil

Failback, Failover, Takeover? Load Balance? This is getting costly!

Copyright Push Technology 2012

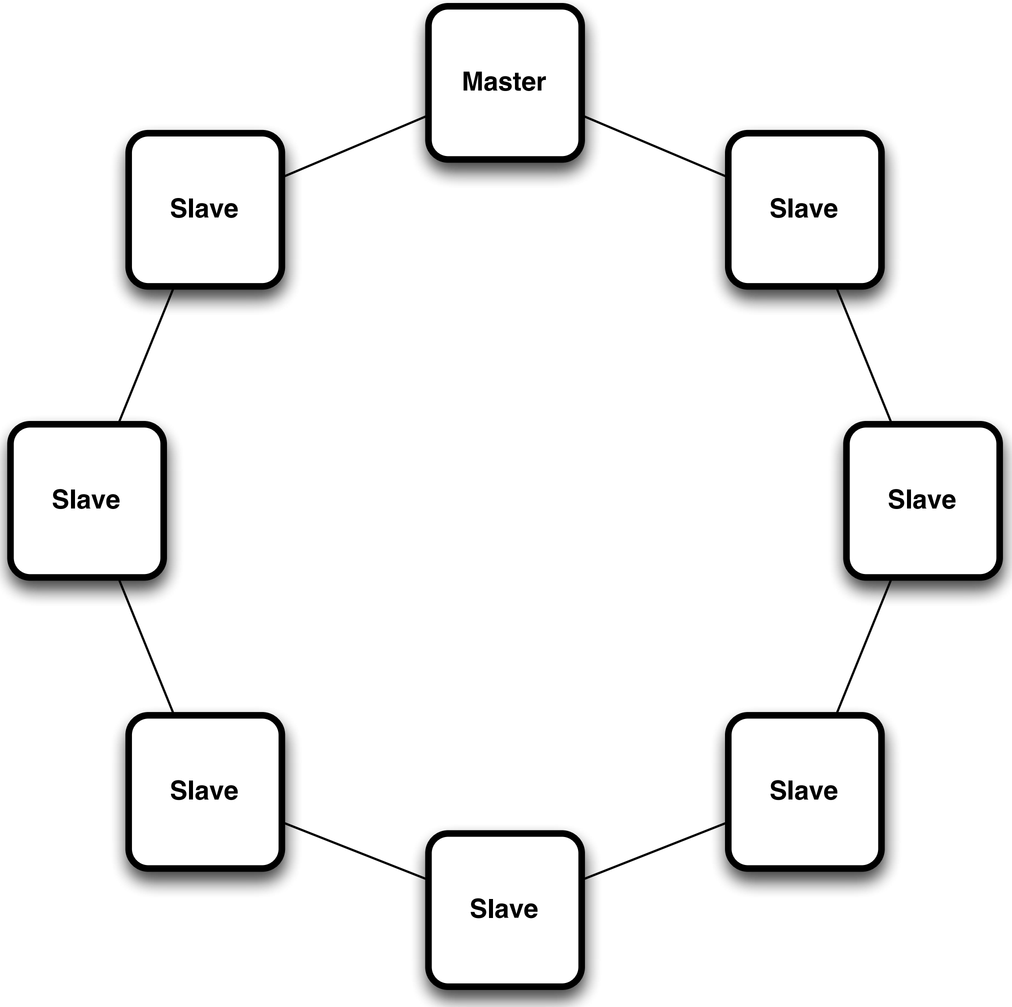
Can implement N-Plex HA with a ‘simple’ FSM



Consider roles for Application, Controller, callbacks, …

Copyright Push Technology 2012

Spin up a cluster, ring?



A simple implementation can be easier to build than HA pairs. Your mileage may vary. Can be masterless too.

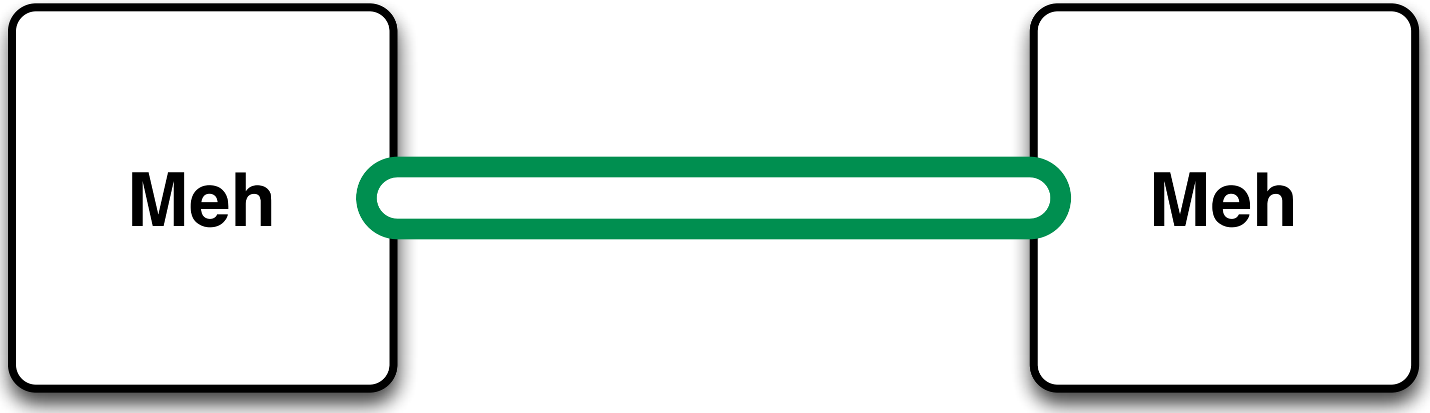
Where possible, use existing solutions such as Akka, riak_core, doozer, Apache Zookeeper.

You'll still need to roll your own data recovery - mostly. Although CRDT's are changing that in the database community.

Copyright Push Technology 2012

One more thing … Real-Time streams

Data!



Real-time data streams are long-lived, conversational, can be stateful.

Copyright Push Technology 2012

Streaming Data Recovery

Type	Before			After			Comments
Precise	T1	T2	T3	T4	T5	T6	Completely masks failure.
Gap	T1	T2	T3	?	T5	T6	May result in data loss
Repeating Rollback	T1	T2	T3	T2	T3	T4	IO preservation. P/S dups are equivalent.
Convergent Rollback	T1	T2	T3	X2	X3	X4	IO preservation. P/S dups may differ initially but EC.
Divergent Rollback	T1	T2	T3	X2	X3	X4	IO preservation. P/S dups will differ

http://nms.lcs.mit.edu/papers/ha-icde05.pdf - HA algos for DSPs

Copyright Push Technology 2012

But, problem: The bird, basically.

Immediately Inconsistent.

But, Eventually Consistent

… Maybe.

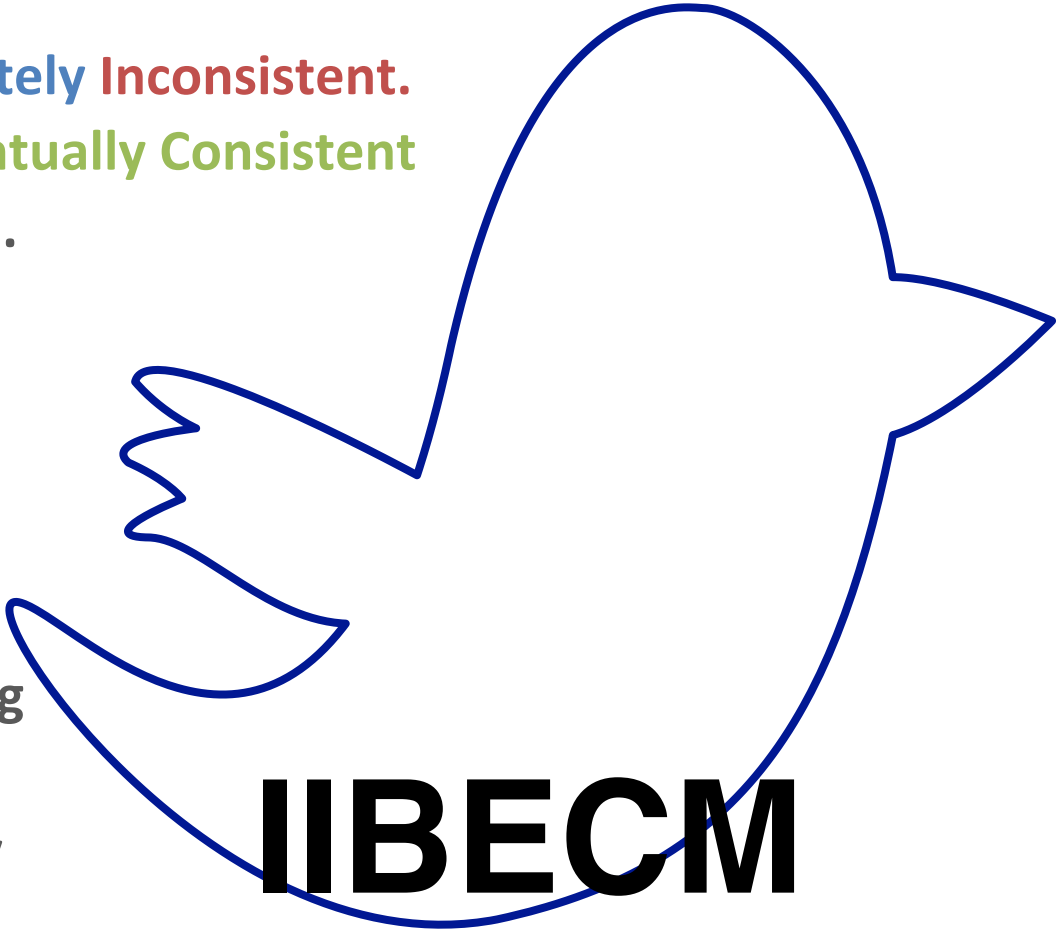
Humans

Hate

Repeating

Rollback

Recovery

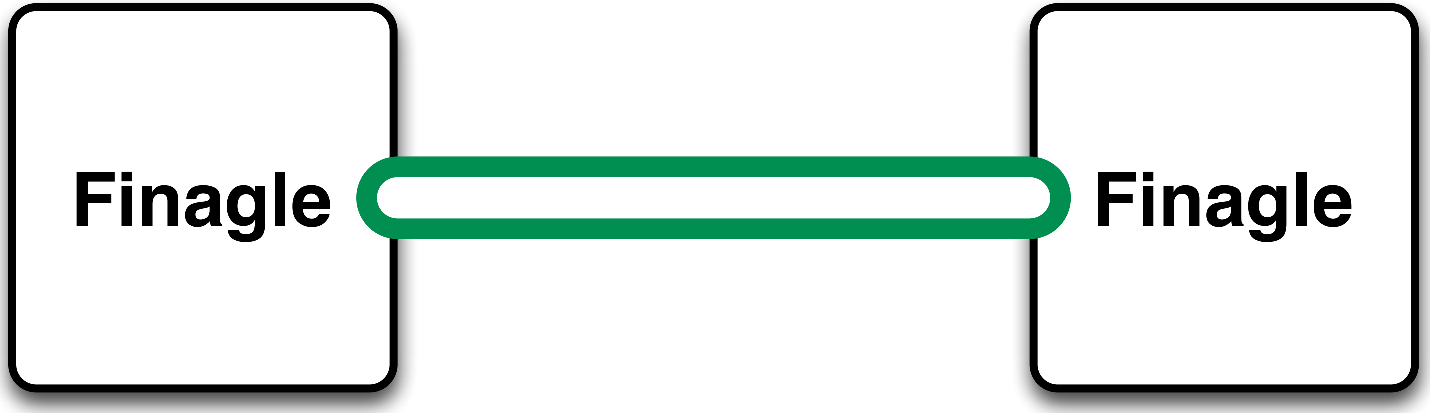


IIBECM

Copyright Push Technology 2012

Ugh, just one more thing …

State?



Data rarely passes through a box without being fiddled with.

Copyright Push Technology 2012

Streaming Operations – 4 types

Type	Comments
Arbitrary	Completely masks failure.
Deterministic	Given Same Input, Same State, produce Same Output
Convergent-capable	Given Same Input, Earlier Start, Empty State, converge to Same Output as Deterministic case
Repeatable	Given Same Input, Earlier State, Empty State, requires at most one input to product at most one output to be convergent-capable and same as deterministic case

http://nms.lcs.mit.edu/papers/ha-icde05.pdf - HA algos for DSPs

Copyright Push Technology 2012

DOH?

Data Recovery Type x Operation Type = The Hard Part

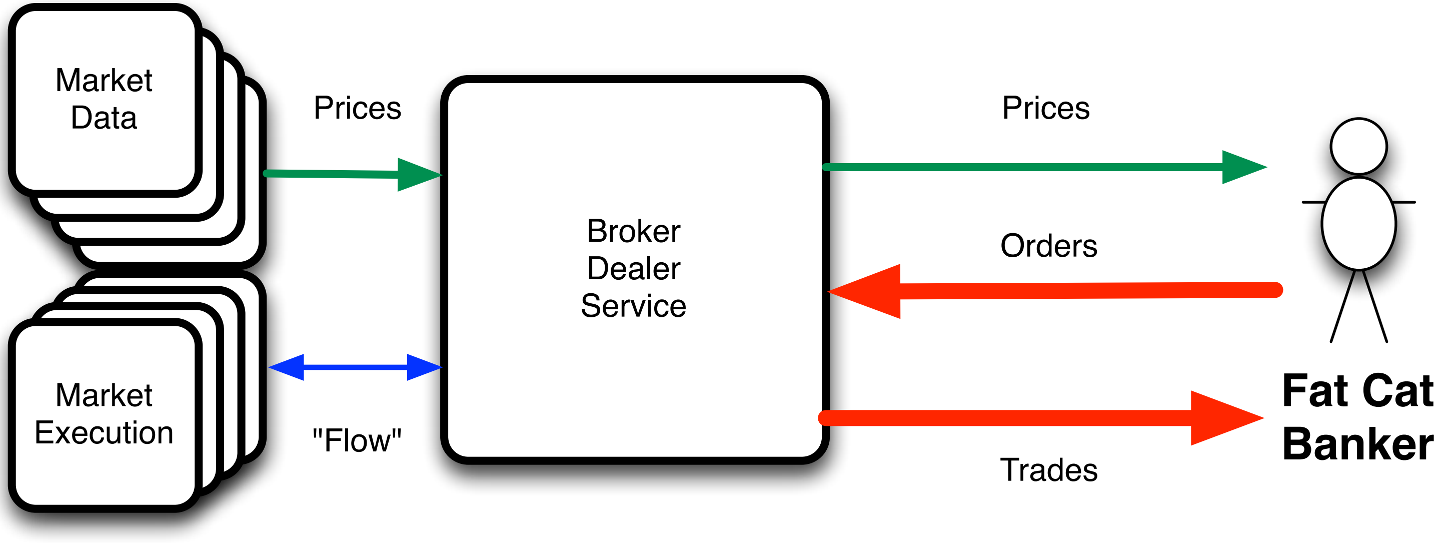
Copyright Push Technology 2012

Just one last one more thing …

Human Expectations

We are fickle, grumpy, changeable, intolerant …

Let’s see. Trading systems?



It Depends…
It's Complicated!

Precise Recovery

Gap Recovery

There are at least 26 race conditions in FIX 4.2 Order Cancel/Replace/Reject

And please. No stale data!

Copyright Push Technology 2012

A most inconvenient truth

Simple is Hard

So **relax**. Let it **fail**. Let **live** and let’s **learn**.

Copyright Push Technology 2012

Stop.

- The network is **not** reliable
  nor is it cost free.
- Latency is **not** zero
  nor is it a democracy.
- Bandwidth is **not** infinite
  nor predictable especially the last mile!
- There is **not only** one administrator
  trust, relationships **are** key
- Topology **does** change
  It should, however, converge eventually
- The network is **not** secure
  nor is the data that flows through it
- Transport cost is **not** zero
  but what you **don't do** is **free**
- The network is **not** homogeneous
  nor is it **smart**

Copyright Push Technology 2012

Look.

- **High Availability is not enough for Machine to Human**
- **High Experience requires more**
    - Data has a time value – out of date data is bad.
    - Data has variable relevance, based on location, device …
    - Data rates are faster than human perception
    - You cannot send 100mb/sec down 3G networks
    - You cannot send 100mb/sec down 3G networks
    - You cannot … waste is bad, bloats buffers, and 'slows' data.
    - Currency & (Immediate) Consistency are important in M2H.
- **M2H 'High Available Experience' might work for M2M …**

Listen.

- **Every** nuance comes with a set of tradeoffs.
- Choosing the right ones can be hard, but it pays off.
- Context, Environment are critical
- Break all the rules, one benchmark at a time.

- **Benchmark Driven Development**

Copyright Push Technology 2012

Action: Data Distribution

Messaging remixed around:

Relevance - Queue depth for conflatable data should be 0 or 1. No more

Responsiveness - Use HTTP/REST for things. Stream the little things

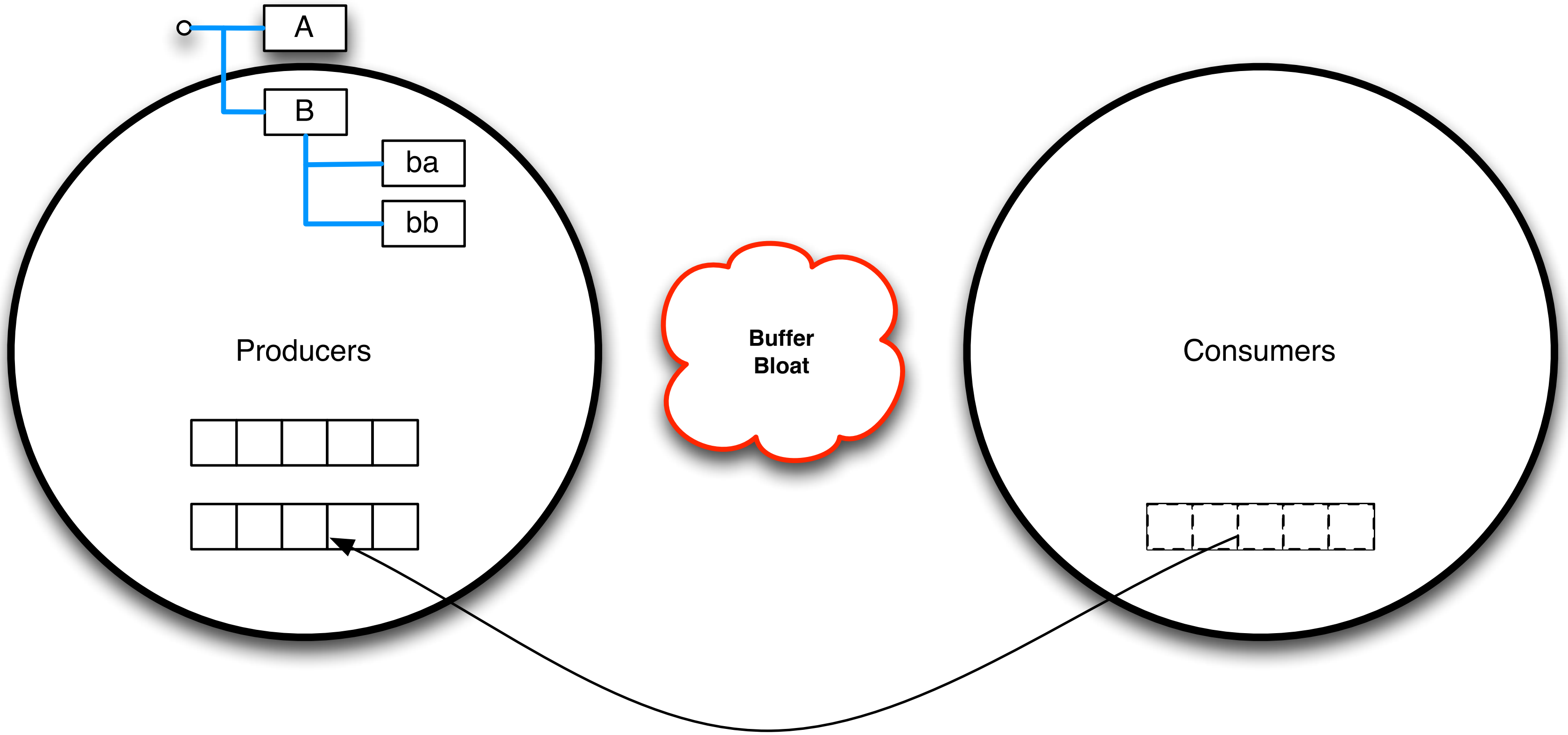
Timeliness - It’s relative. M2M != M2H.

Context - Packed binary, deltas mostly, snapshot on subscribe.

Environment- Don’t send 1M 1K events to a mobile phone with 0.5mbps.

Copyright Push Technology 2012

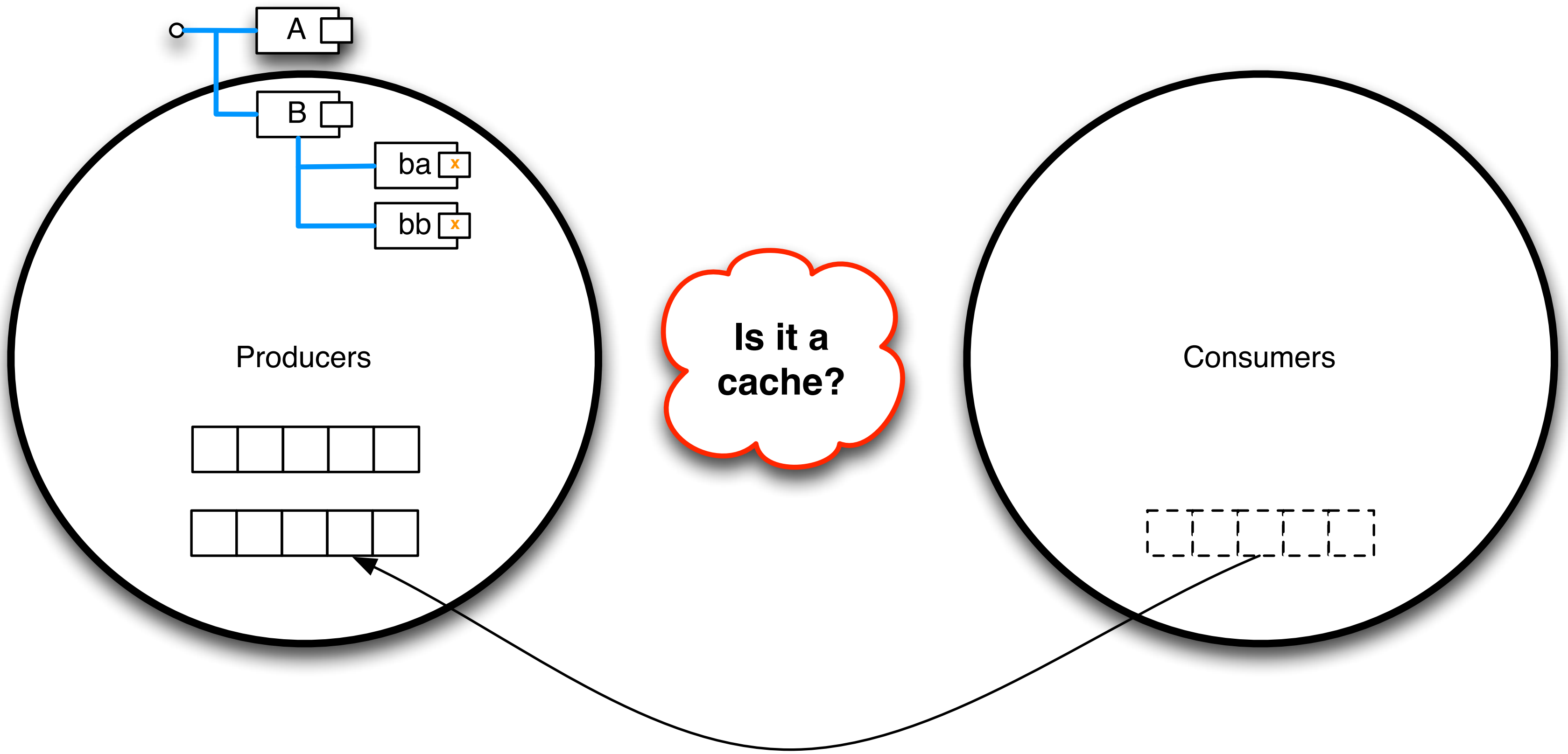
Action. Virtualize Client Queues



Nuance: Client telemetry. Tradeoff: Durable subscriptions harder

Copyright Push Technology 2012

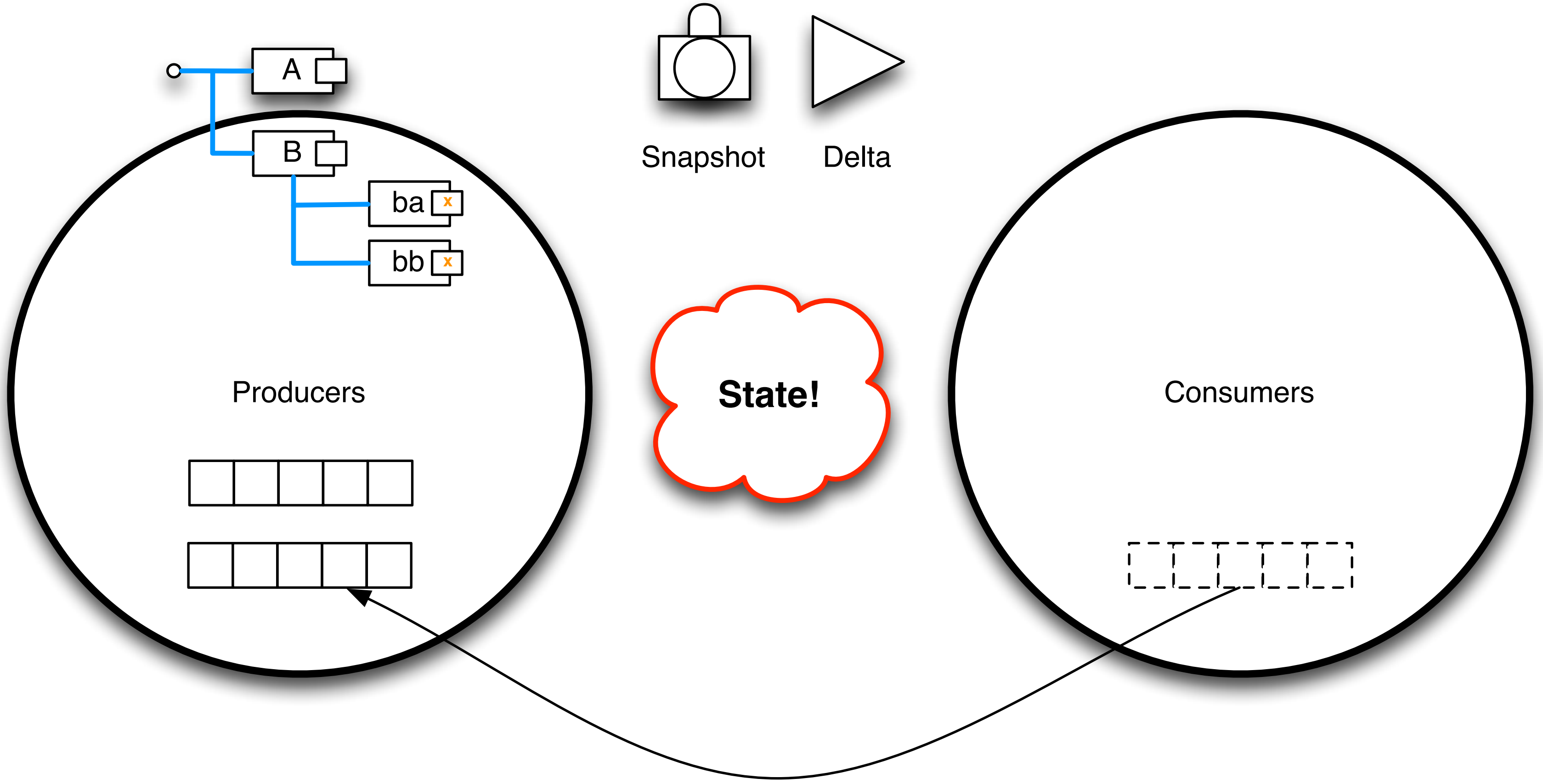
Action. Add data caching



One hop closer to the edge …

Copyright Push Technology 2012

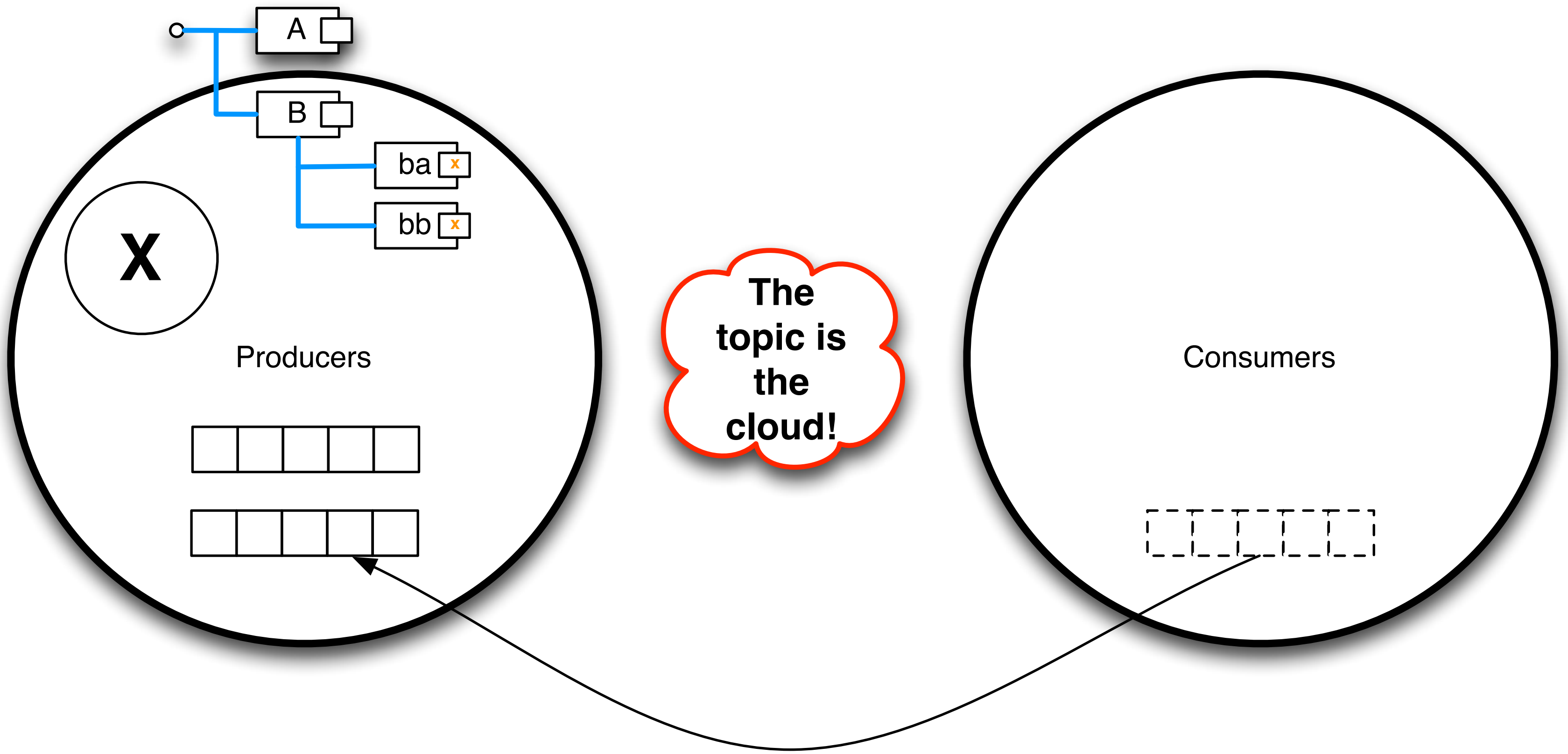
Action. Exploit data structure



Snapshot recovery. Deltas or Changes mostly. Conserves bandwidth

Copyright Push Technology 2012

Action. Behaviors



Extensible. Nuance? Roll your own protocols. Tradeoff? 3rd party code in the engine :/

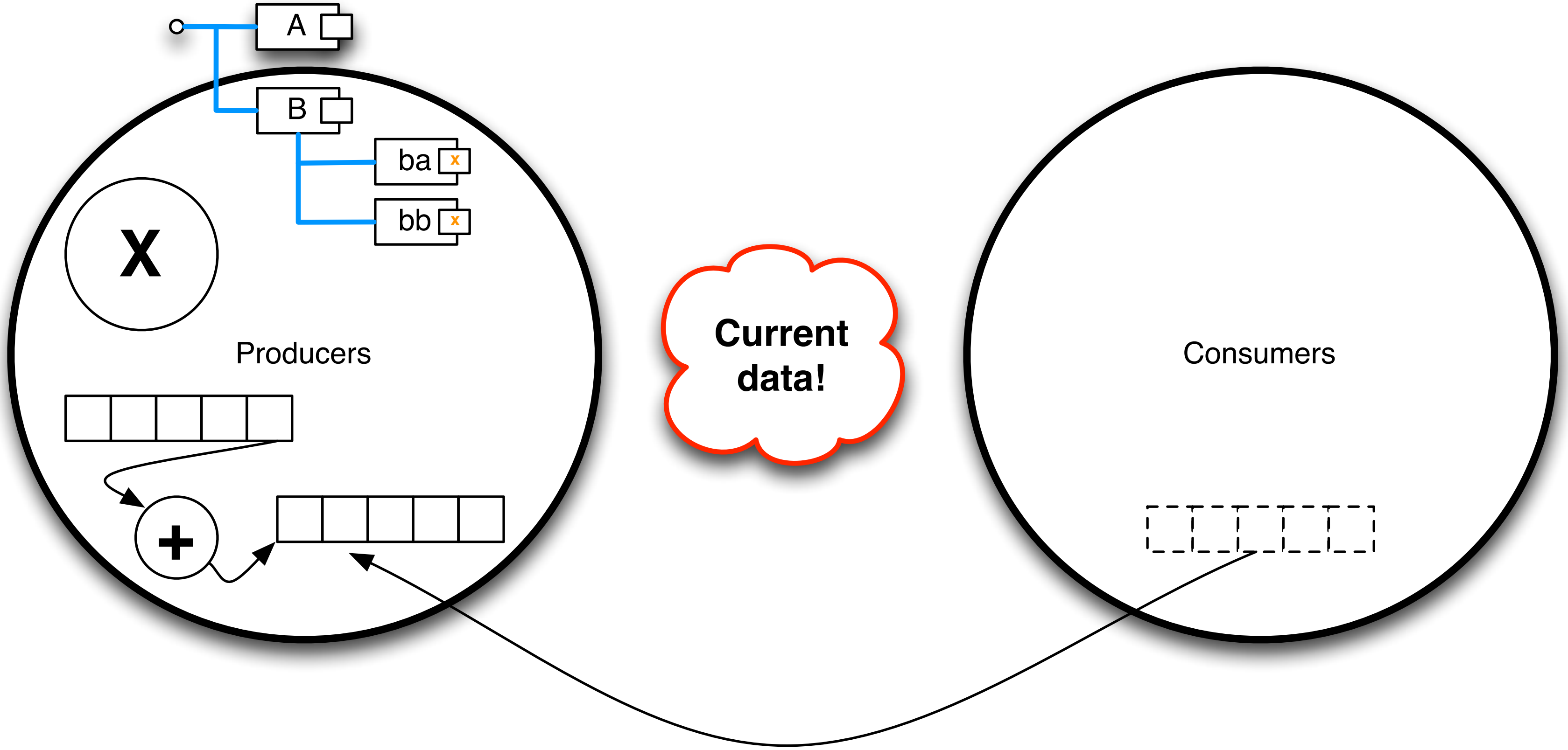
Copyright Push Technology 2012

3rd Party Code in the Engine?

- A bug in 3rd party code can take out the system.
- Is necessary in many environments.
- Can force low density deployments.
- Solutions are appearing, such as Waratek for the JVM.
  - You can hive out 3rd party code into a lightweight fully isolated container.
  - You can manage the container lifecycle.
  - You can reallocate memory, CPU, network.
  - It's language level virtualization.

Copyright Push Technology 2012

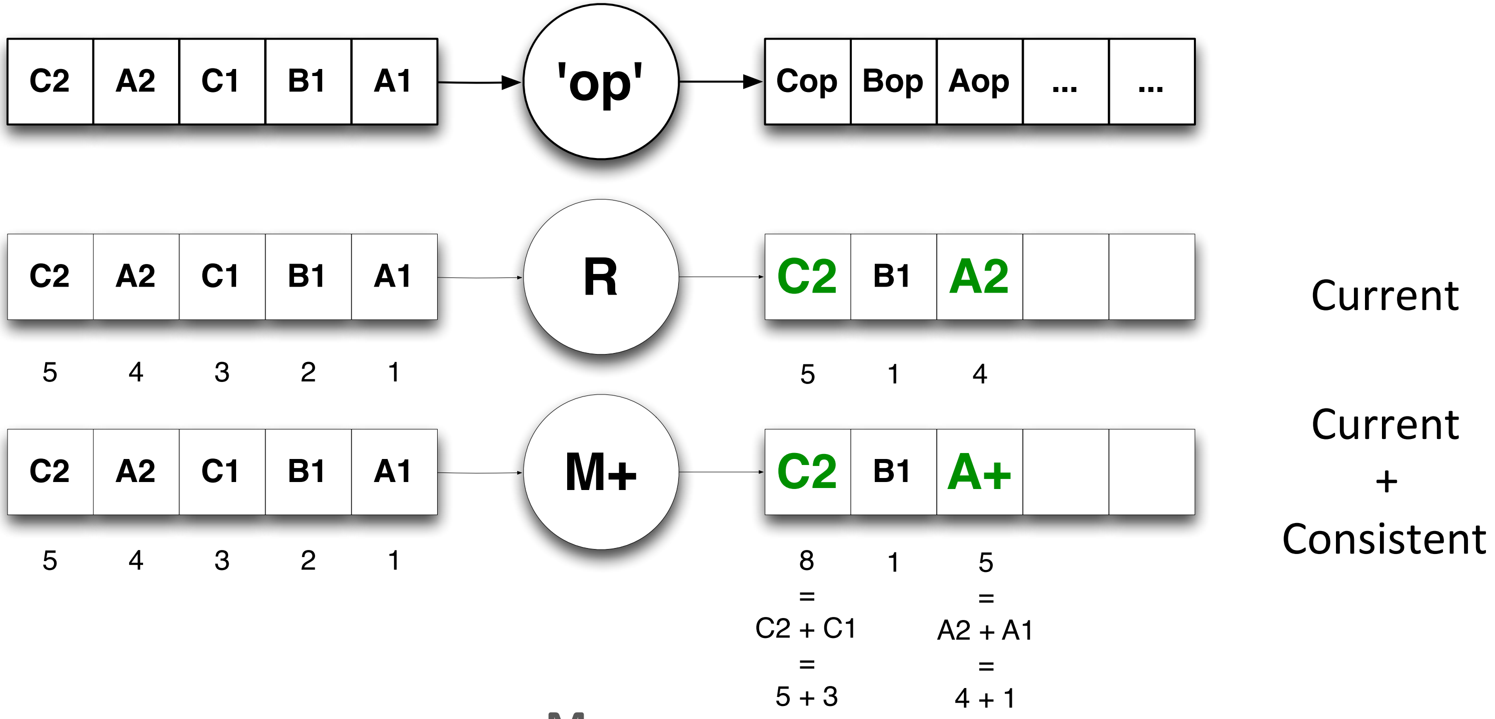
Action. Structural Conflation



Ensures only current + consistent data is distributed. Actively soaks up bloat. Extensible!

Copyright Push Technology 2012

Action. Structural Conflation [EEP]



Replace

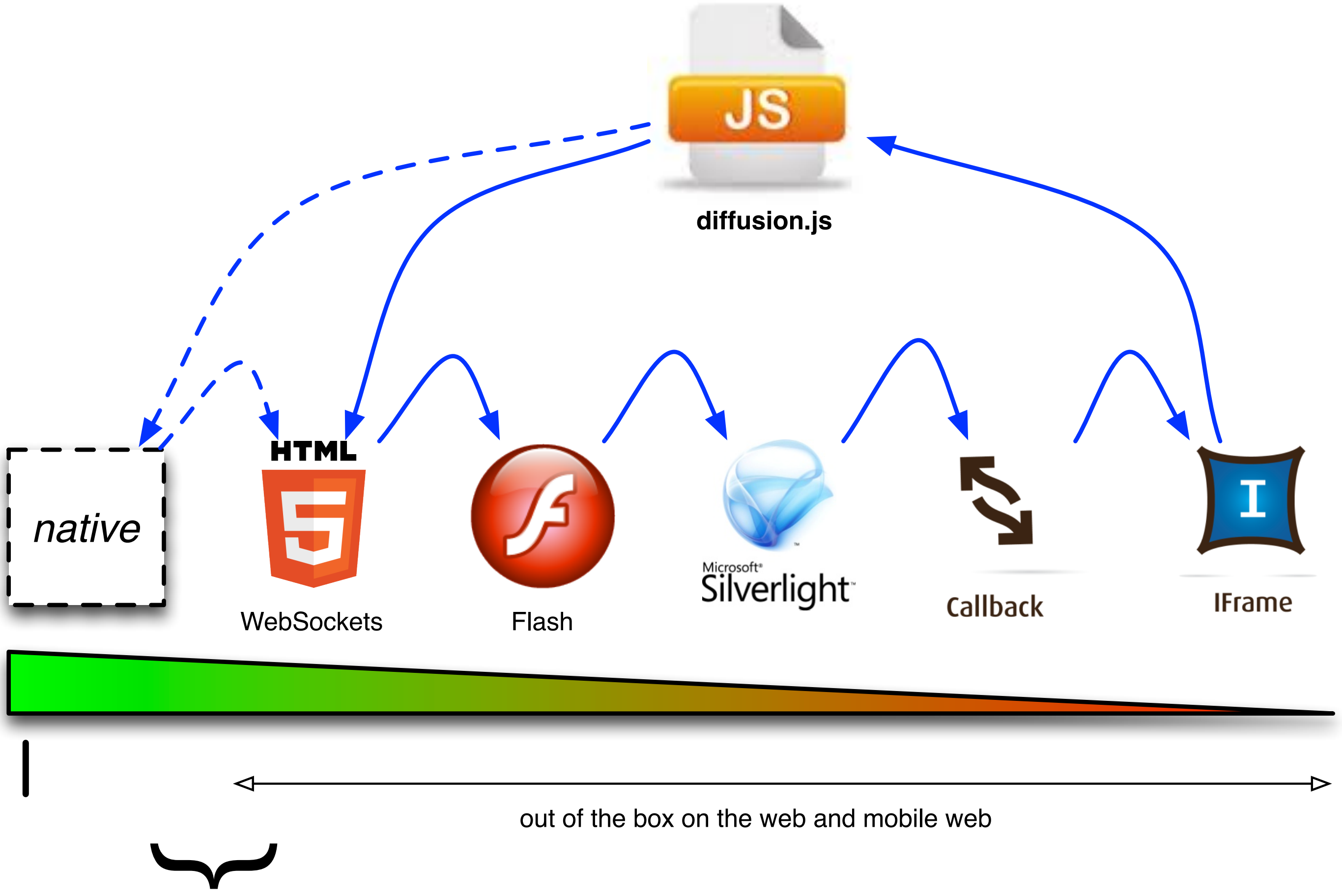
- Replace/Overwrite 'A1' with 'A2'
- 'Current data right now'
- Fast

Merge

- Merge A1, A2 under some operation.
  - Operation is pluggable
  - Tunable consistency
  - Performance f(operation)

Copyright Push Technology 2012

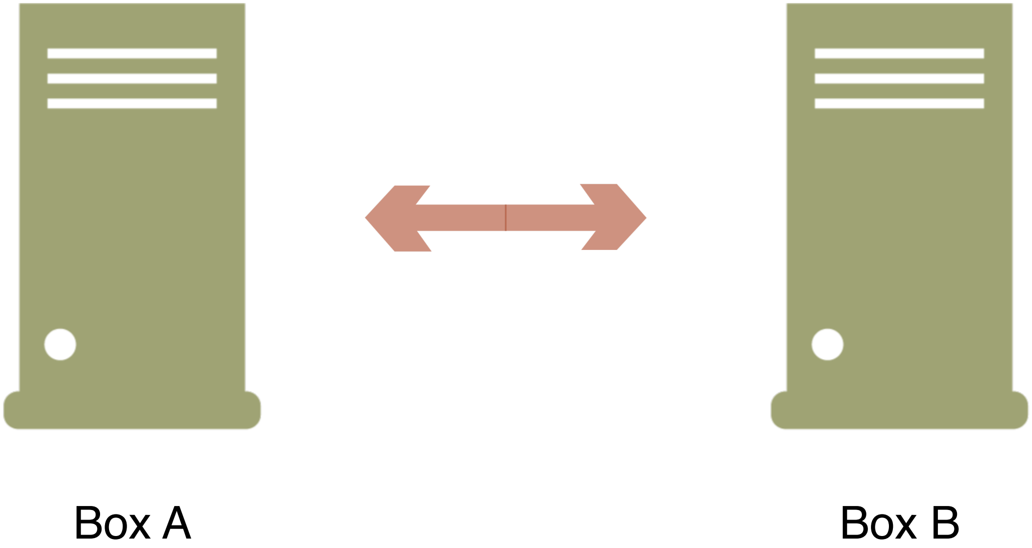
Client? Connection & Transport Cascading



Bespoke cascading -
Have your cake and eat it

Copyright Push Technology 2012

An extreme example.



Throughput (Worst case)

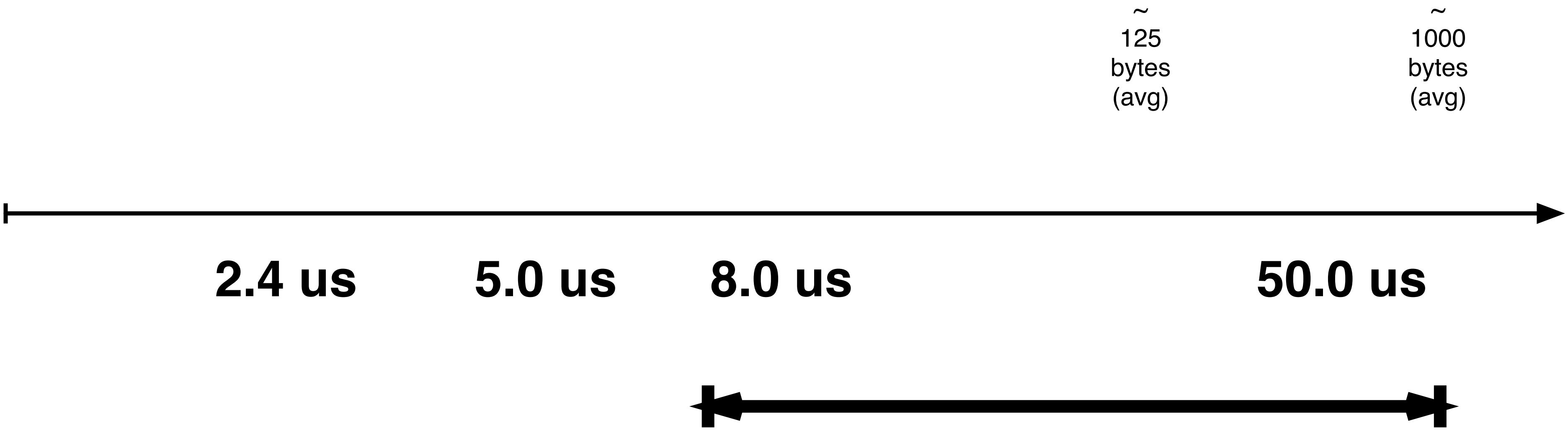
- Ramp clients continuously
- 100 messages per second per client
- Payload: 125 .. 2000 bytes
- Message style vs with conflation

Latency (Best case)

- Really simple. 1 client
- Ping – Pong. Measure RTT time.
- Payload: 125 .. 2000 bytes

Copyright Push Technology 2012

Latency in perspective



~ 125 bytes (avg)

~ 1000 bytes (avg)

2.4 us

5.0 us

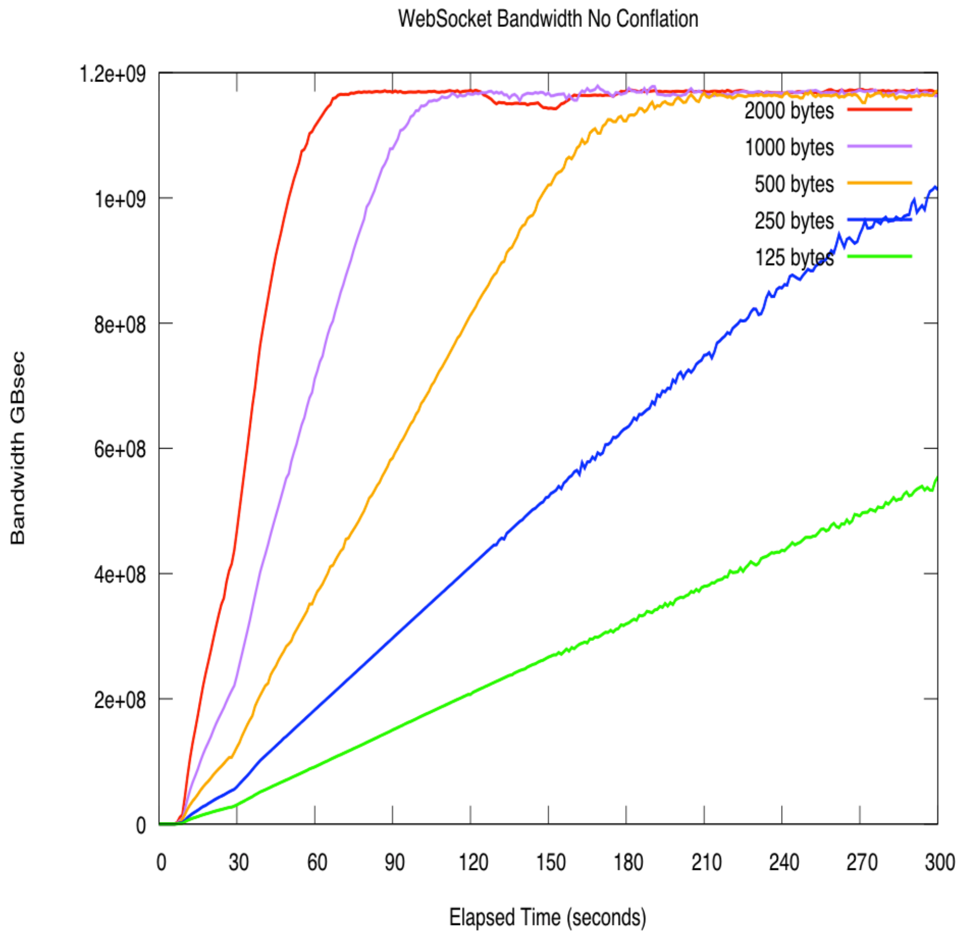
8.0 us

50.0 us

- 2.4 us. A tuned benchmark in C with low latency 10Ge NIC, with kernel bypass, with FPGA acceleration
- 5.0 us. A basic java benchmark – as good as it gets in java
- Diffusion is measurably ‘moving to the left’ release on release
- We’ve been actively tracking and continuously improving since 4.0

Copyright Push Technology 2012

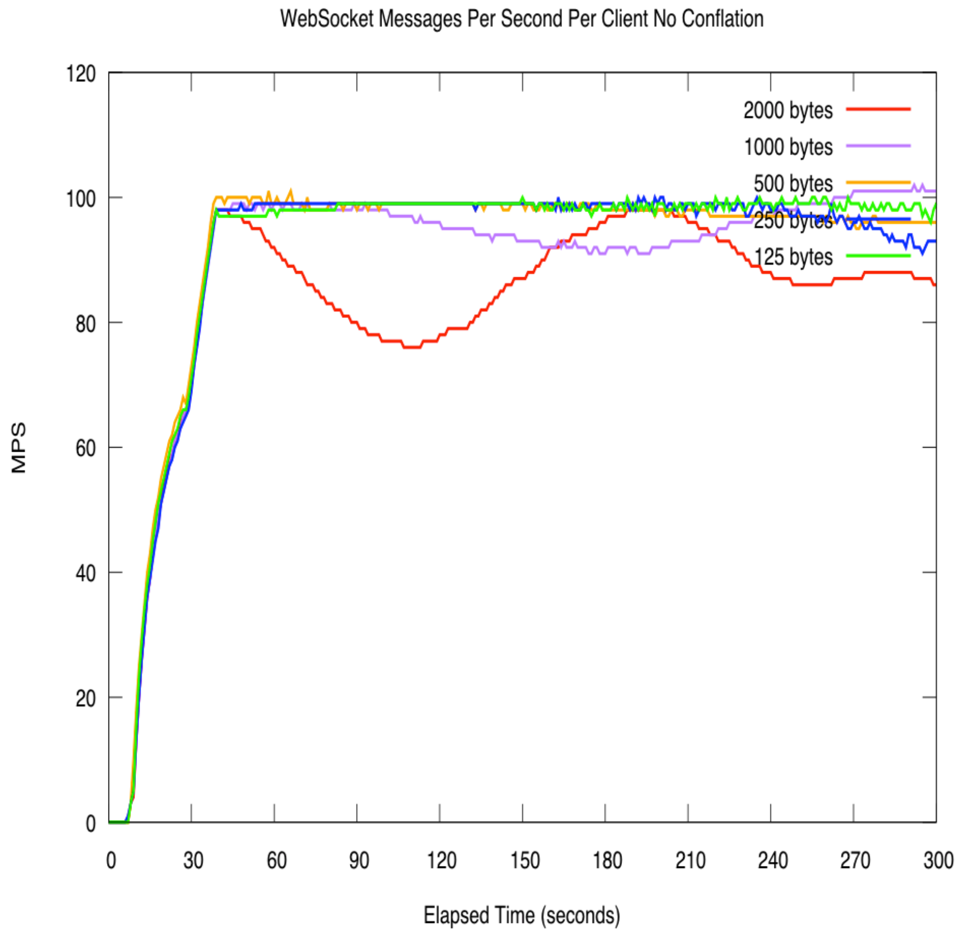
Throughput. Server view. Non-conflated data



- Cold start server
- Ramp 750 clients 'simultaneously' at 5 second intervals
- 5 minute benchmark duration
- Clients onboarded linearly until IO (here) or compute saturation occurs.
- What the 'server' sees

Copyright Push Technology 2012

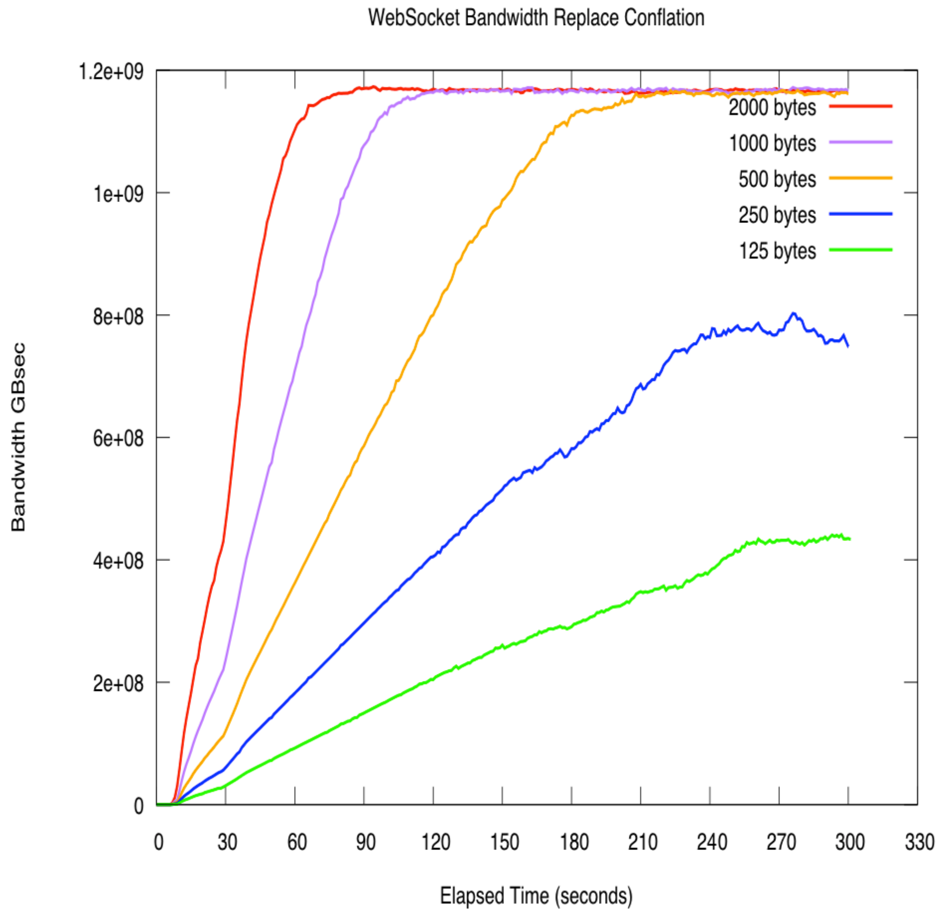
Throughput. Client view. Non-conflated data.



- What the ‘client’ sees
- At and beyond saturation of some resource?
- Things break!
- New connections fail. Good.
- Long established connections ok.
- Recent connections timeout and client connections are dropped. Good.
- Diffusion handles smaller message sizes more gracefully
- Back-pressure ‘waveform’ can be tuned out. Or, you could use structural conflation!

Copyright Push Technology 2012

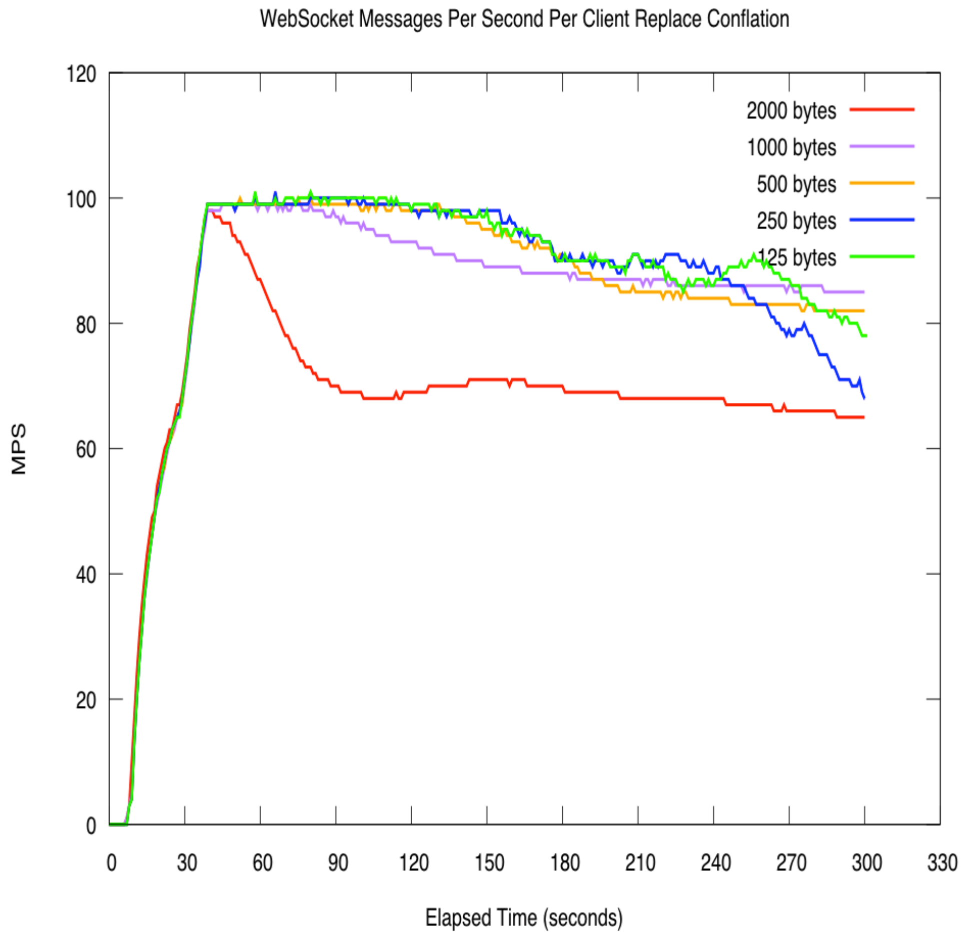
Throughput. Server view. Replace-conflated



- Cold start server
- Ramp 750 clients ‘simultaneously’ at 5 second intervals
- 5 minute benchmark duration
- Clients onboarded linearly until IO (here) or compute saturation occurs.
- Takes longer to saturate than non-conflated case
- Handles more concurrent connections
- Again, ‘server’ view

Copyright Push Technology 2012

Throughput. Client view. Replace-Conflated



- What the 'client' sees
- Once saturation occurs Diffusion adapts actively by degrading messages per second per client
- This is good. Soak up the peaks through fairer distribution of data.
- Handles spikes in number of connections, volume of data or saturation of other resources using a single load adaptive mechanism

Copyright Push Technology 2012

Can failure be fun?

- Where do **HA algorithms fail** humans?
- Can ***consistent experiences*** be delivered elapsing short, medium and longer term failures?
- Does an **M2H eye for an M2M** guy even work?

Copyright Push Technology 2012

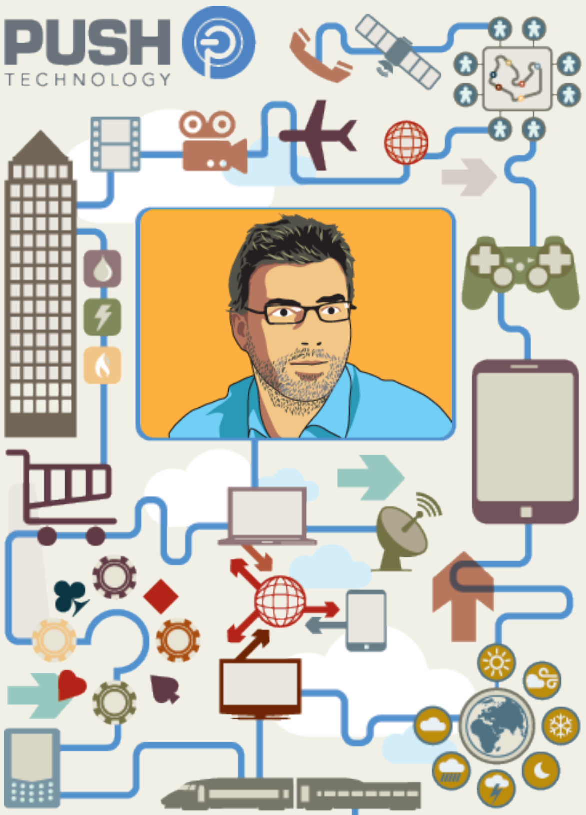
Who knows. Let's figure it out! ☺

Copyright Push Technology 2012

My ‘number’

#6752

Copyright Push Technology 2012



- Thank you for listening to, having me
- Le twitter: @darachennis
- Work. http://www.pushtechnology.com/
- Fun. http://github.com/darach/
- Grab me any time on Weds/Thurs.
- Like Erlang? Big Data?
  London User Group meetup.
  Thurs March 7th, QE2 18:30 – 20:00.
  (Google for registration details).

Questions?

Copyright Push Technology 2012