Databases suck for Messaging

Alexis Richardson

Oxford Geek Night May 2009

RabbitMQ
Open Source Enterprise Messaging

Computers were meant to get rid of this

A new kind of fail?

Solution - use a database?

Databases were meant to get rid of this too

I want to know when THIS changes

Problem - databases suck for messaging

- DATA is facts (“state”) persisted on disk
- Databases are great for storing facts and asking questions about them
- If you know what to ask them and are willing to keep asking them

- INFORMATION is always changing
- Networks are great pushing changes (“messages”) to recipients
- Databases are not optimised for this

Social applications store data

PostZinger
facebook
RSSready
YouTube
Broadcast Yourself
*clearspring
blip.tv
sprout
viddler
REVVER
YAHOO! GROUPS
digg
blogtalkradio
del.icio.us
social bookmarking
tubemogul
empowering online video
AOL video
Google
myspace.com
a place for friends
StumbleUpon
Discover new sites
flickr
twitter
Big Boards
collective intellect
MyBlogLog
Howcast
LinkedIn
Technorati
friendfeed

Social applications store data

And there's terabytes of it

Social applications store data

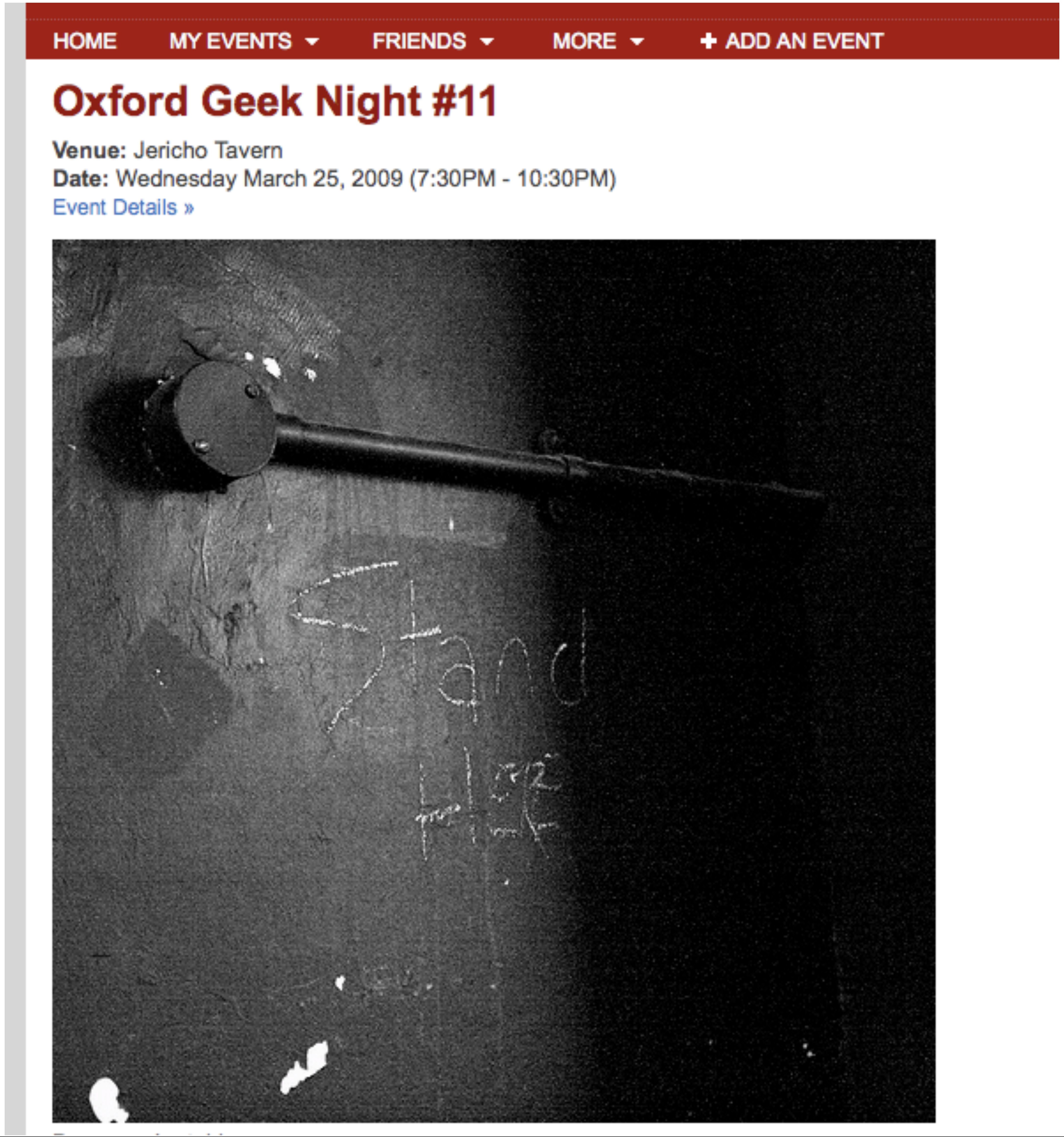
And there’s terabytes of it

Enterprises are made of people too
So: the same issues arise

I want to know when THIS changes

Email doesn't scale

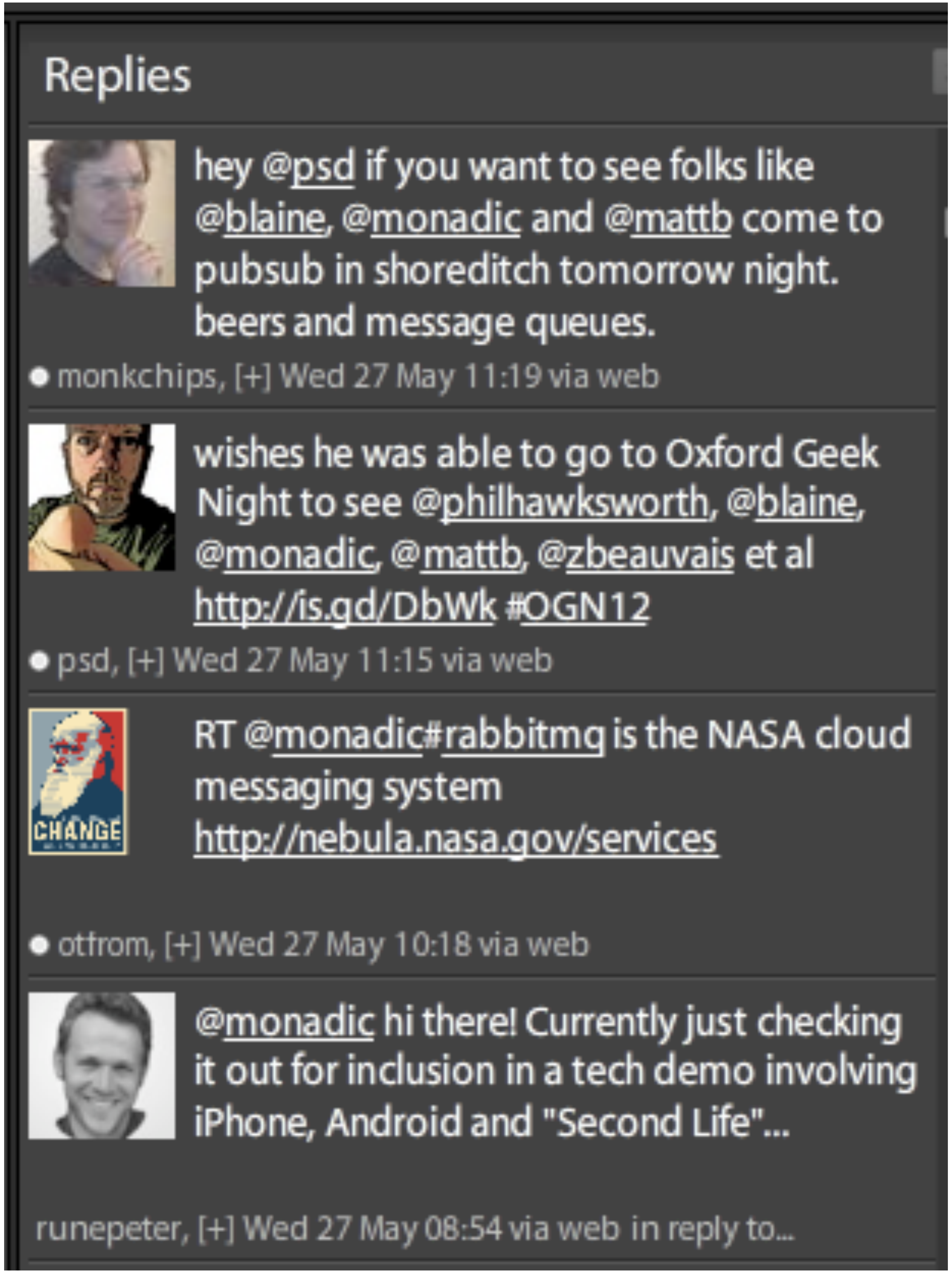
Information gets old - real quick



Oxford Geek Night #11

Venue: Jericho Tavern
Date: Wednesday March 25, 2009 (7:30PM - 10:30PM)
Event Details »

But current information is valuable



Replies

hey @psd if you want to see folks like @blaine, @monadic and @mattb come to pubsub in shoreditch tomorrow night. beers and message queues.

● monkchips, [+] Wed 27 May 11:19 via web

wishes he was able to go to Oxford Geek Night to see @philhawksworth, @blaine, @monadic, @mattb, @zbeauvais et al http://is.gd/DbWk #OGN12

● psd, [+] Wed 27 May 11:15 via web

RT @monadic#rabbitmq is the NASA cloud messaging system http://nebula.nasa.gov/services

● otfrom, [+] Wed 27 May 10:18 via web

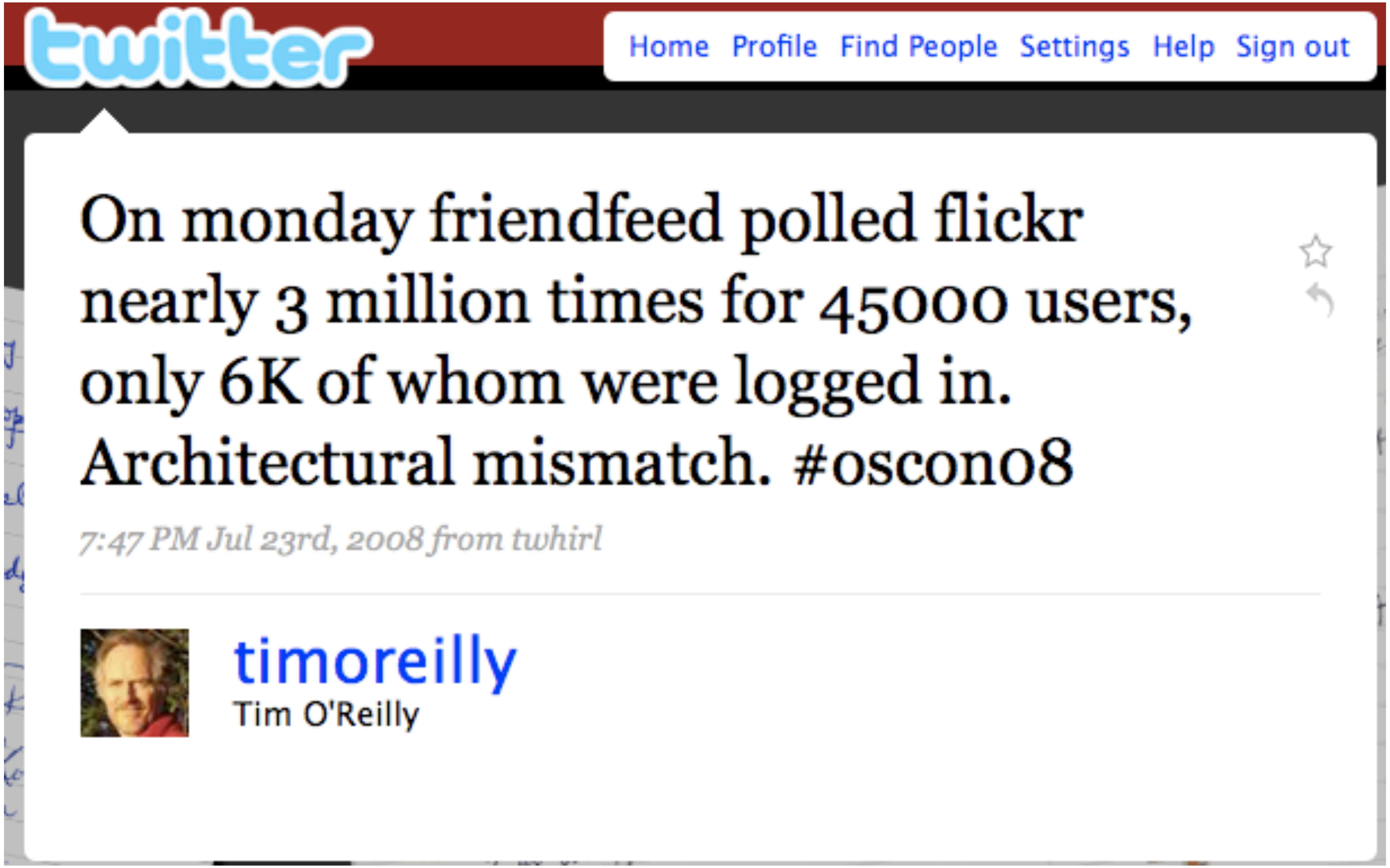
@monadic hi there! Currently just checking it out for inclusion in a tech demo involving iPhone, Android and "Second Life"...

runepeter, [+] Wed 27 May 08:54 via web in reply to...

Example: Flickr

- Flickr is a vast database of social objects
- Filtered by interest and relationships
- So - tell me what’s currently relevant
- Without sending me more emails....

Polling sucks

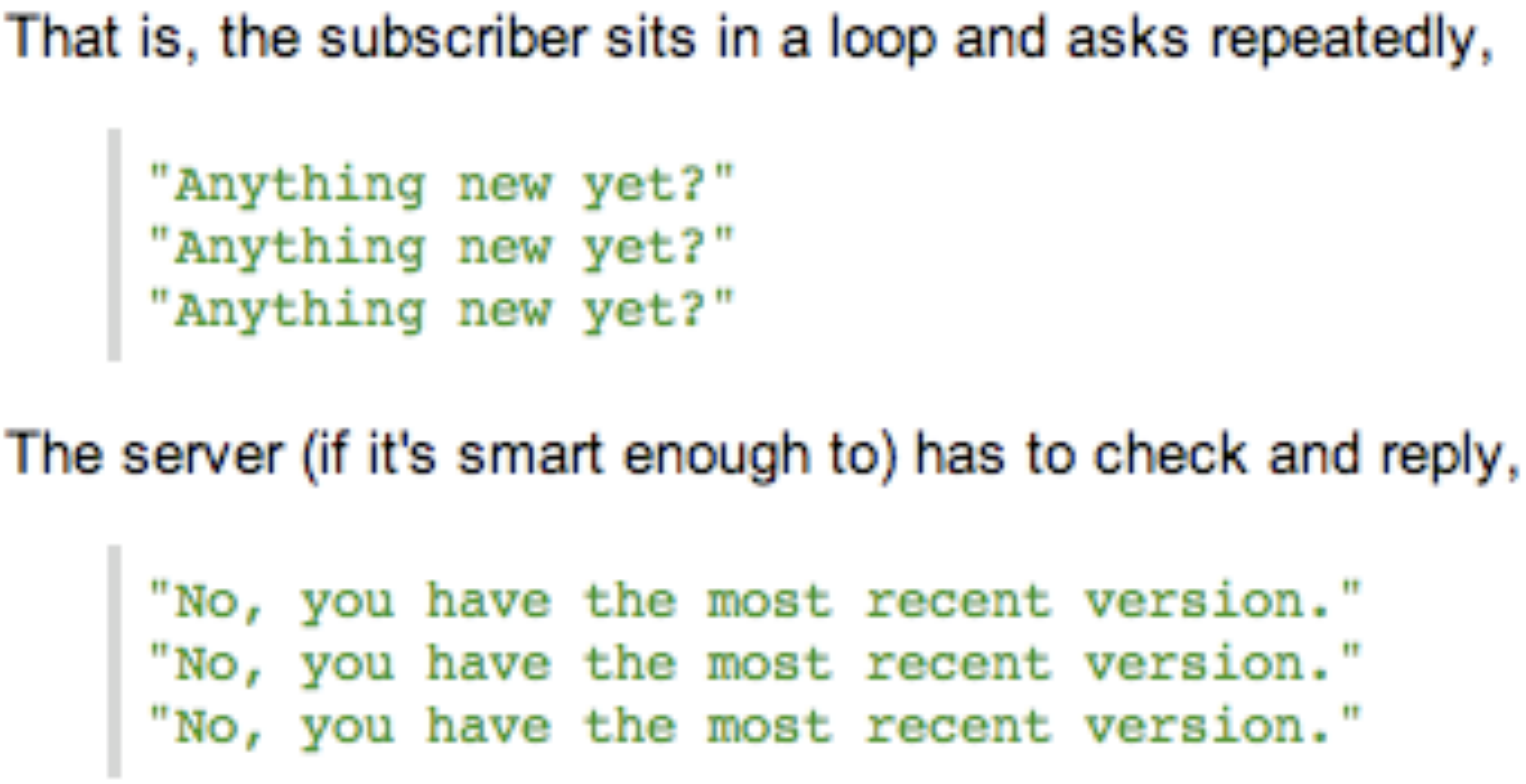


On monday friendfeed polled flickr nearly 3 million times for 45000 users, only 6K of whom were logged in. Architectural mismatch. #oscon08

7:47 PM Jul 23rd, 2008 from twhirl

timoreilly
Tim O'Reilly

What’s going on here? We are trapped in the database world view



That is, the subscriber sits in a loop and asks repeatedly,

```
"Anything new yet?"
"Anything new yet?"
"Anything new yet?"
```

The server (if it's smart enough to) has to check and reply,

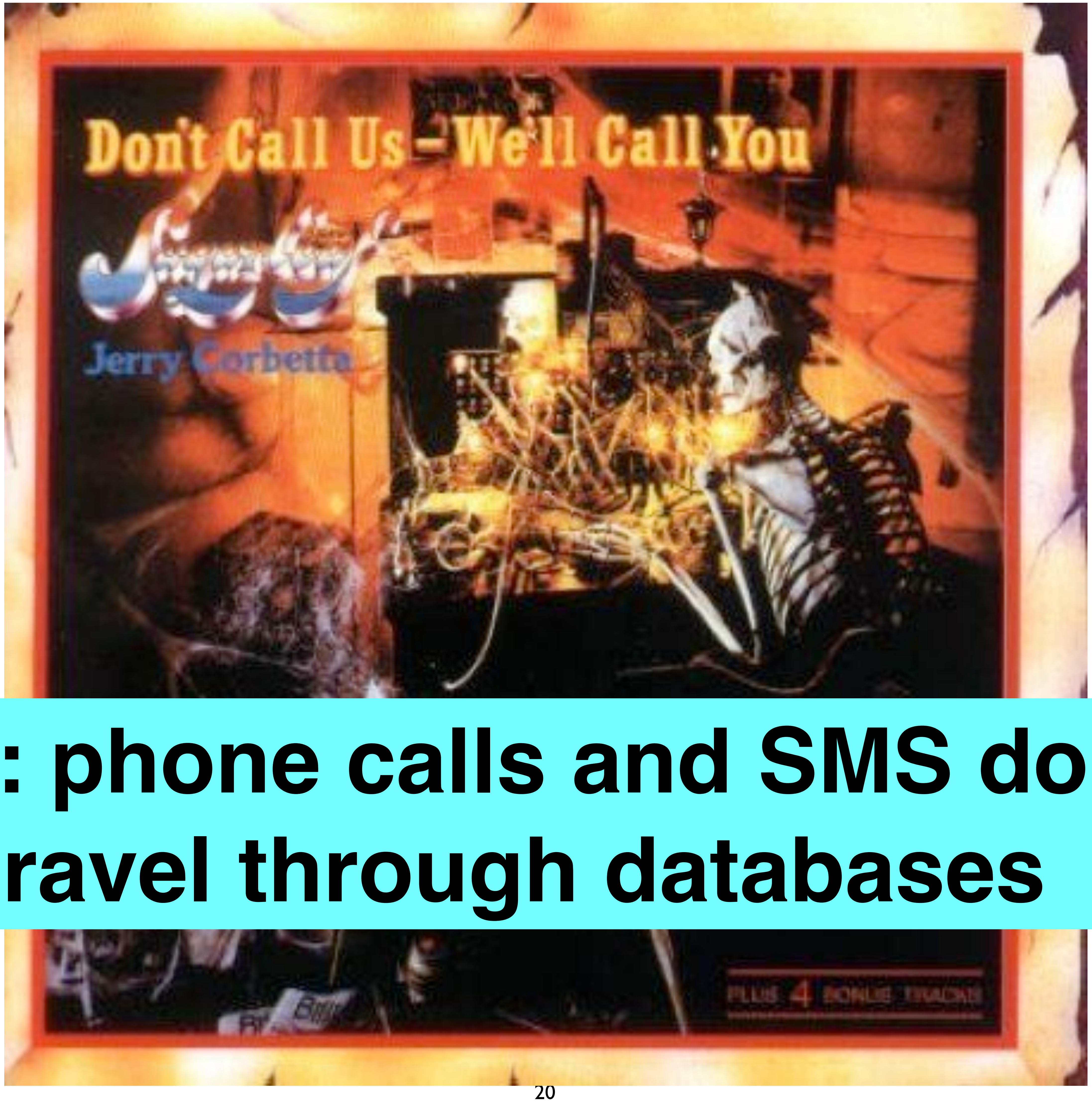
```
"No, you have the most recent version."
"No, you have the most recent version."
"No, you have the most recent version."
```

Example due to Brett Slatkin and Brad Fitzpatrick at Google
http://code.google.com/p/pubsubhubbub/wiki/WhyPollingSucks

How do we defeat this evil?

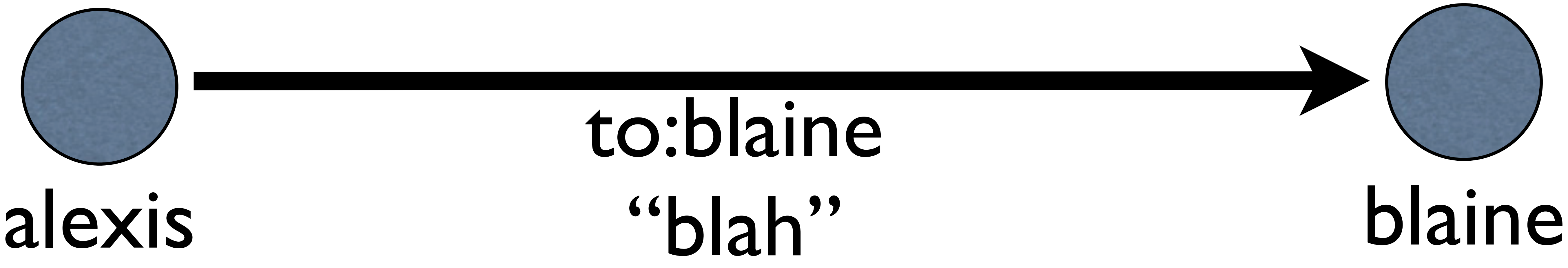
Can we apply the Hollywood Principle?

Can we apply the Hollywood Principle?

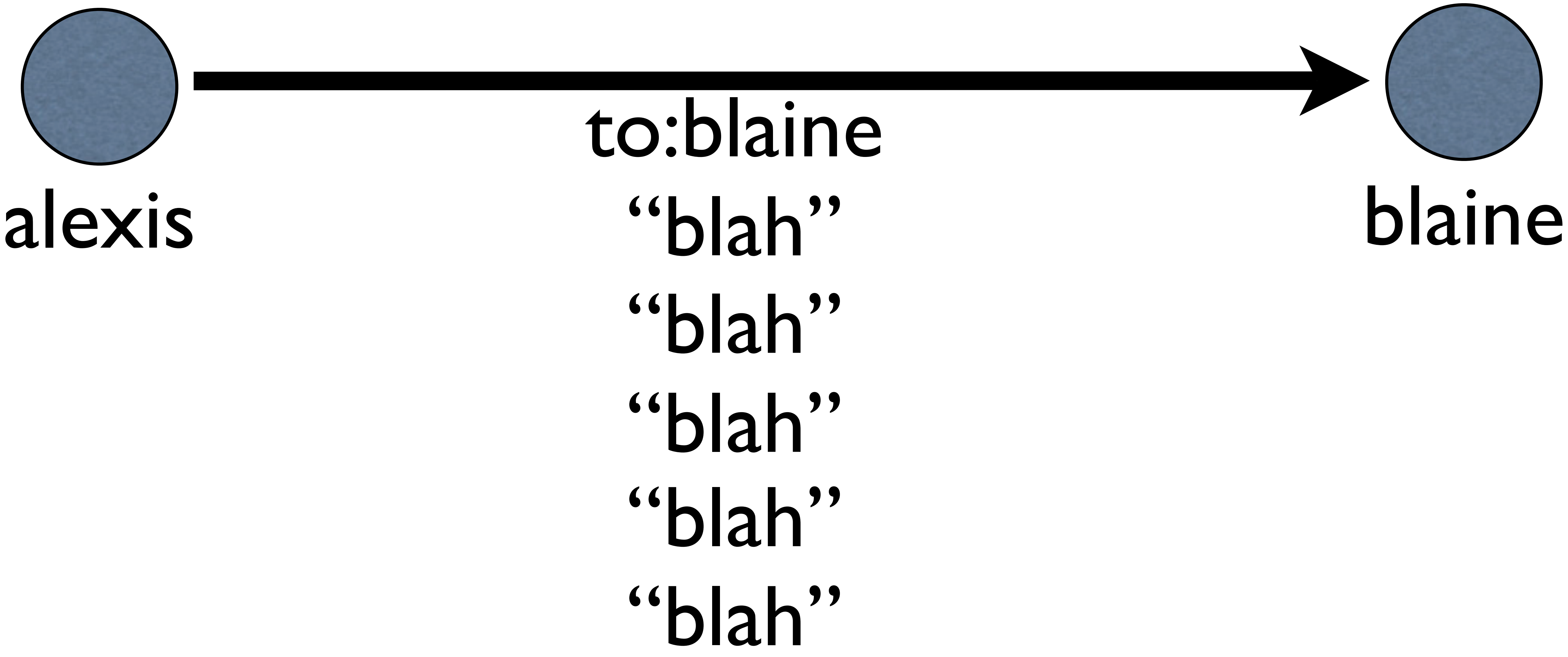


Hint: phone calls and SMS don't travel through databases

Email push is direct



Email push is direct



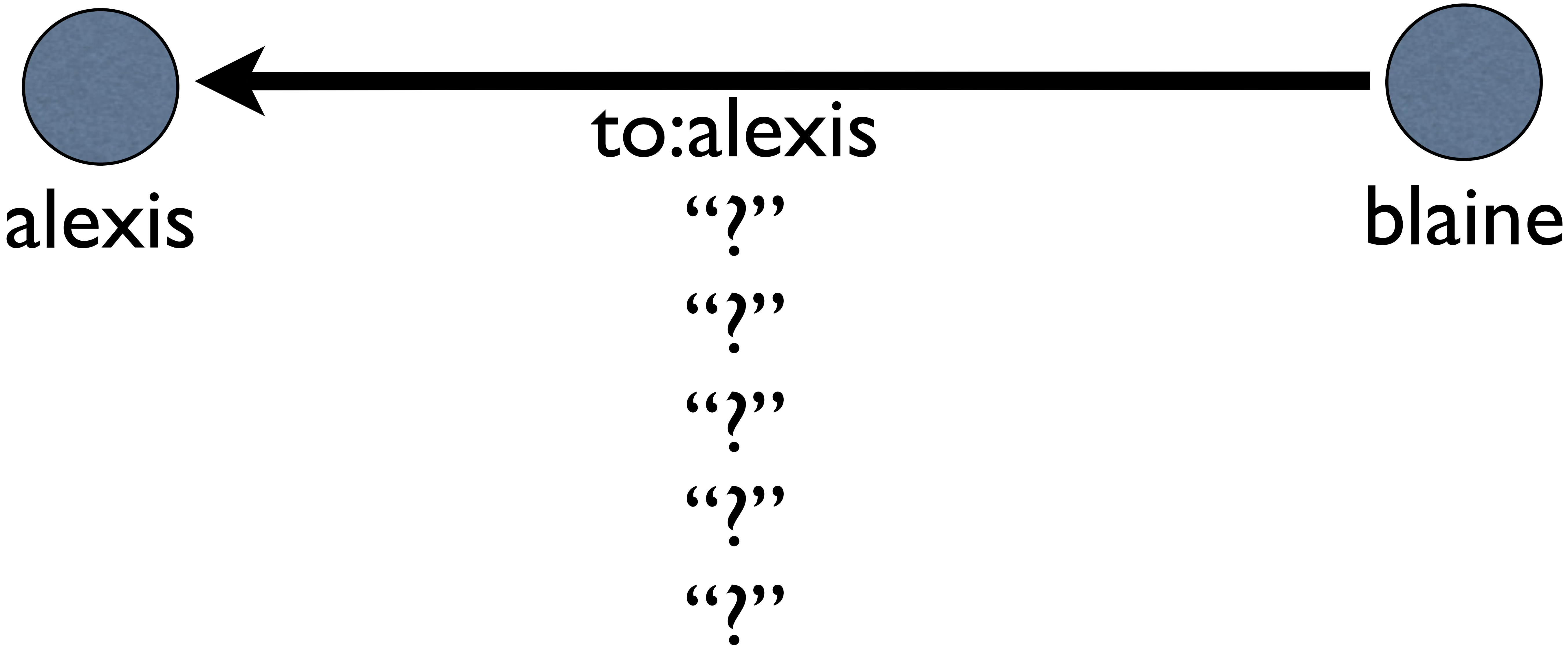
alexis

to:blaine
“blah”
“blah”
“blah”
“blah”
“blah”

blaine

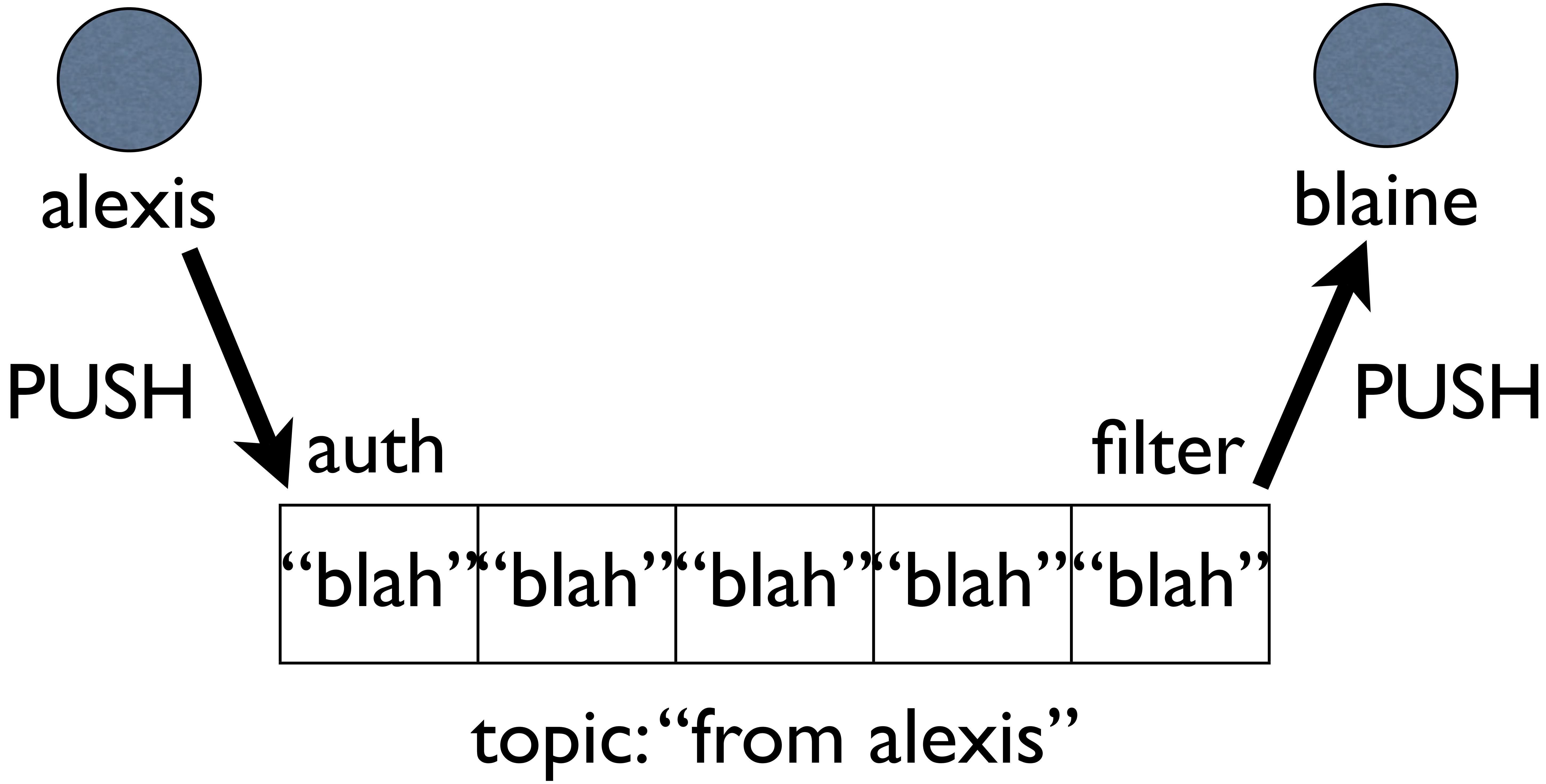
RabbitMQ
Open Source Enterprise Messaging

Polling is just reverse spam



RabbitMQ
Open Source Enterprise Messaging

Publishing to a queue (“topic”) takes the spam burden off the receiver



alexis

blaine

PUSH

auth

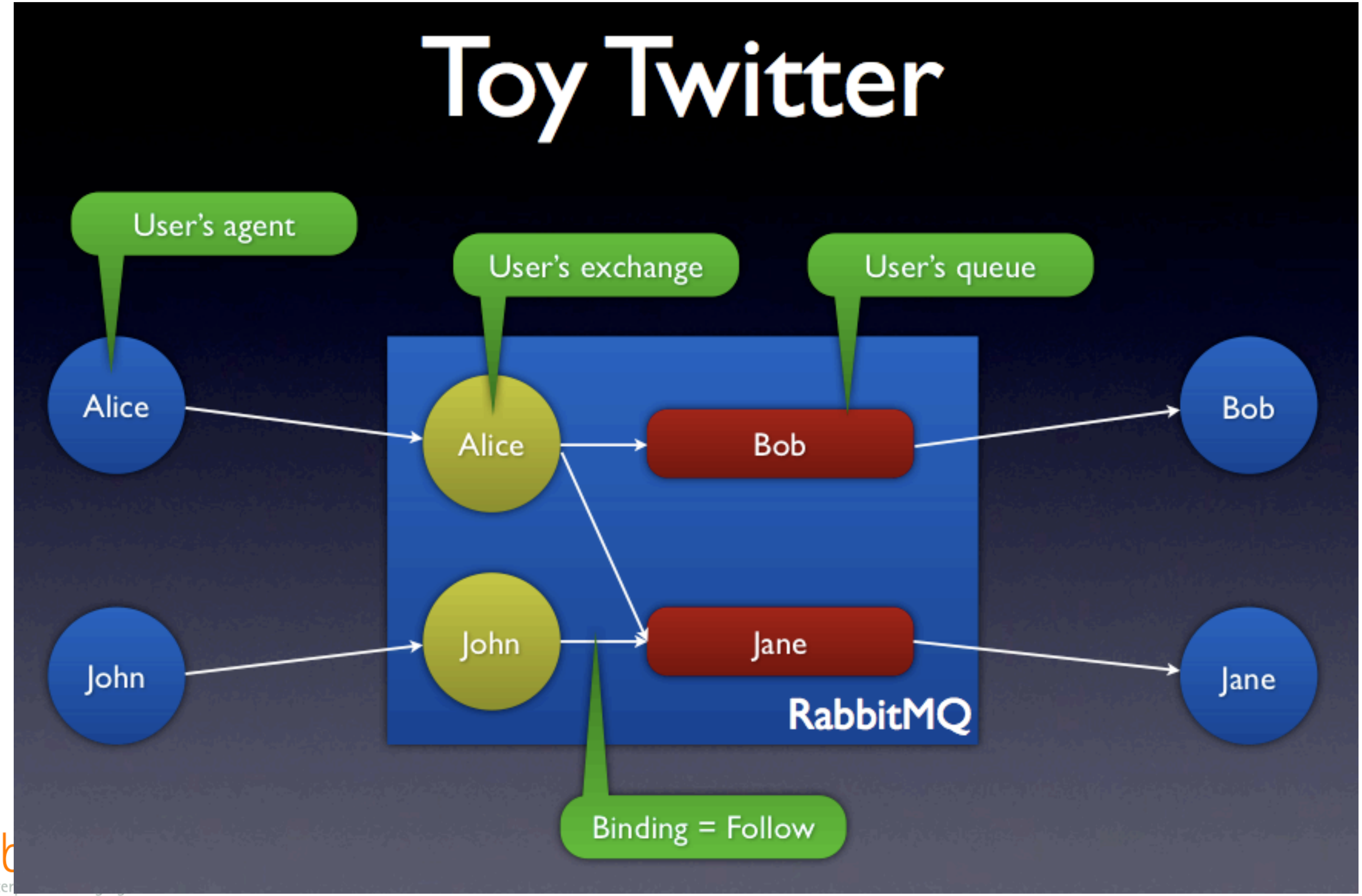
filter

PUSH

“blah”	“blah”	“blah”	“blah”	“blah”

topic: “from alexis”

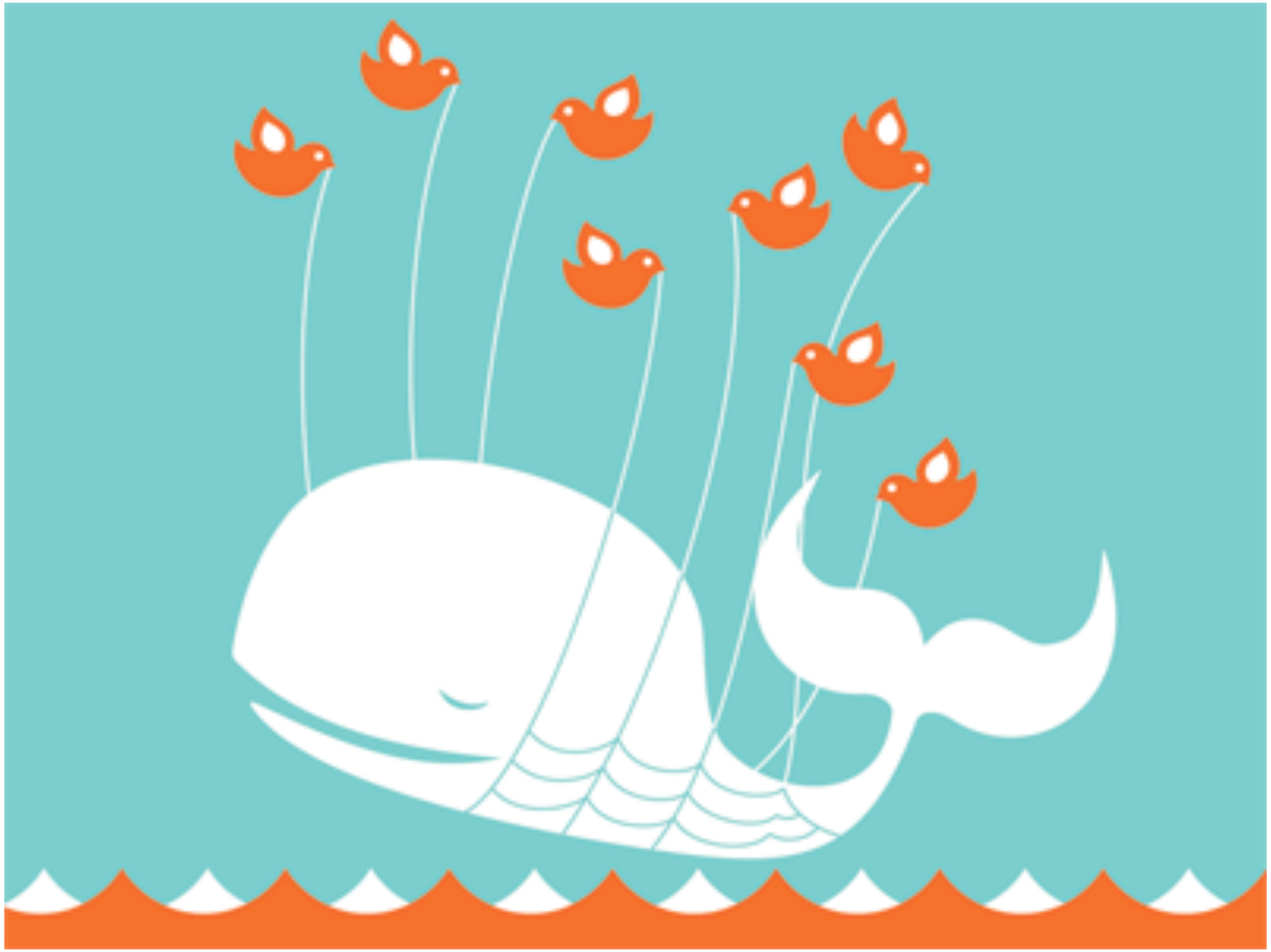
And you can use pubsub and queues in all sorts of ways



Toy Twitter

Databases are not meant to do Pubsub

SELECT * FROM queue WHERE done = 0 ORDER BY created LIMIT 1



RabbitMQ
Open Source Enterprise Messaging

A true story

- most people would just create a simple "queue" table
- and: SELECT * FROM queue WHERE done = 0 ORDER BY created LIMIT 1
- “concurrency issues on that thing now - inserting into the queue occasionally takes longer than doing the task that needs to be executed synchronously”
- “middle management did not want to have "new third party software" because it would be too much Operations to learn and manage”
- so they decided for the time being a MySQL based queue would be sufficient (only few million messages/day) and implemented it in PHP/MySQL resulting in lots of dev hours for implementation / testing, and more hours because of performance issues ..
- so i think we have spent about the same amount of time as we would to get a descent thing up and running, but now we're stuck with a lame and unscalable mysql database machines that is dressed as a message broker
-

Flickr from a Pubsub point of view

- Flickr = people <u>publishing</u> to a vast set of streams (photostreams)
- Users express interest through <u>subscription</u>
- I don’t need to see everything - only changes on what I follow

- This seems better - but what’s missing?
  - I am still trapped by the database world view.
  - I still poll for changes (that’s what RSS does)
  - I want the PUSH that email gave me, without the spam.....

Characteristics of PUBSUB

- A means of authenticated communication (network transport)
    - eg HTTP, OAuth
- An addressable place to publish to
    - Usually a topic, feed, endpoint or address.
- A way to name, enrol, share, and discover these addressable places
    - For example “blaine@twitter.com” or - TBD!
- A way to deliver and ack delivery (or “take responsibility”)
- The above is in fact a distributed object system

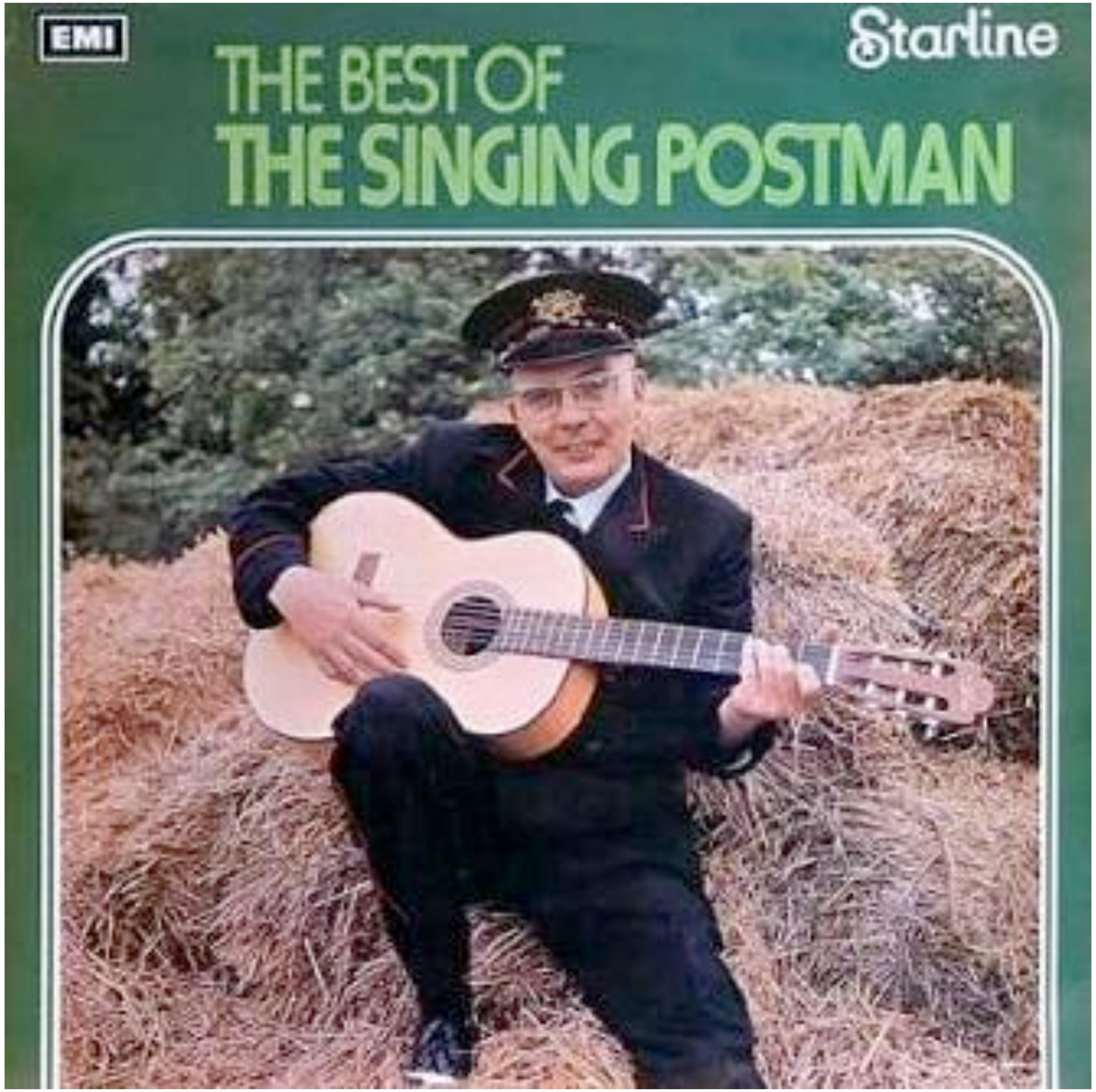
How do you solve a problem like Flickr/Twitter/...

	Database	Pubsub/MQ
Data	Objects serialised as rows in tables	Messages organised into streams
Interest	Filter by query	Durable "follow"
Notification	Pull (polling sucks!)	Push
Buffering (non-idempotent)	Put = add row to table Get = delete top row	PUT and GET are symmetric
Scale	Overheads tend to grow indefinitely	Data flows out to destinations

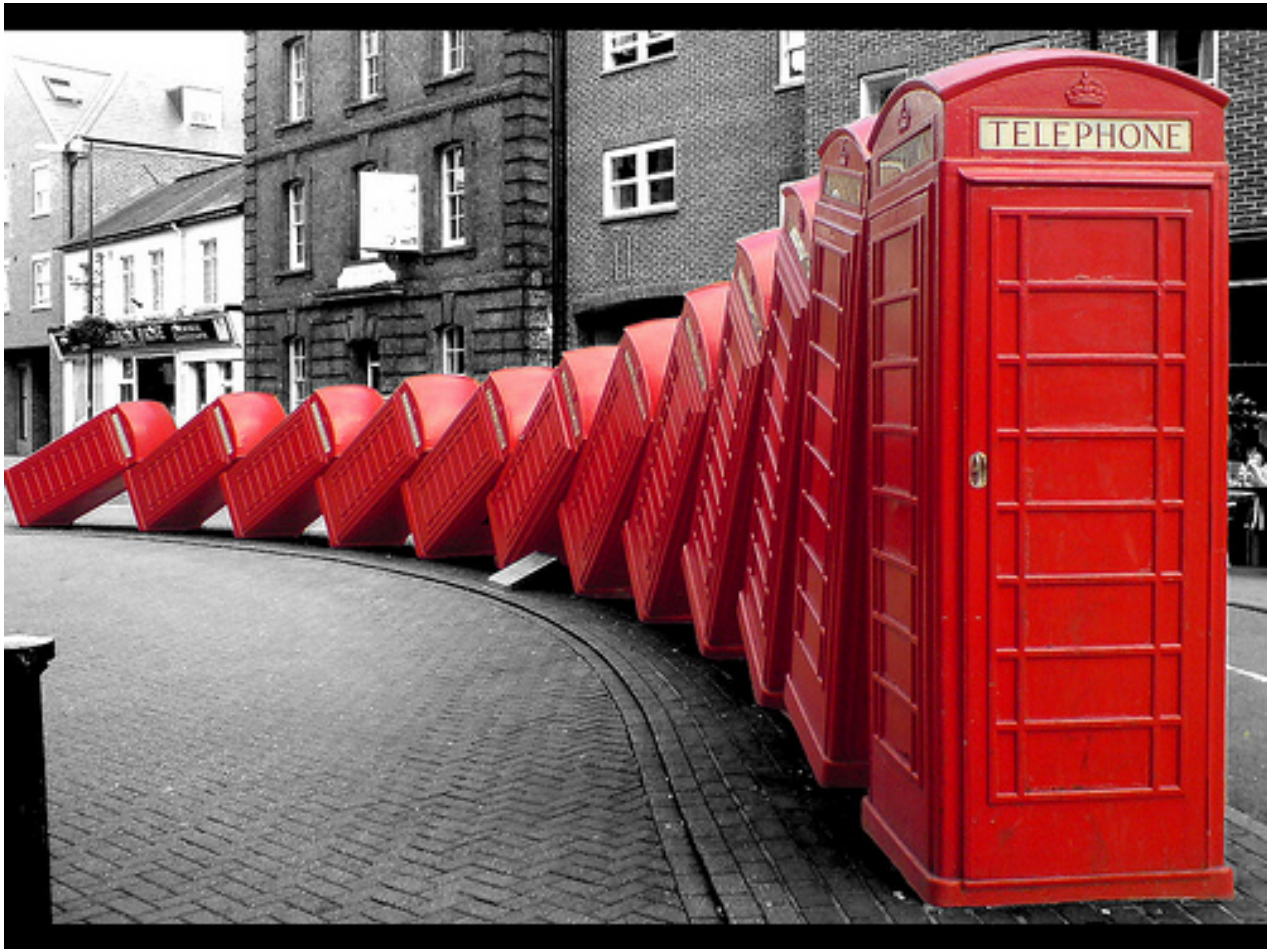
You might need messaging if ... you need to monitor data feeds

(CC) Kishore Nagarigari

You might need messaging if ... you need a message delivered responsibly



You might need messaging if ... you need things done in order



(CC) David Mach

You might need messaging if ... you are using the cloud

RabbitMQ is a messaging server that just works!