RabbitMQ at Skills Matter Cloud Exchange

Cloud Messaging Use Cases

April 2010
Alexis Richardson
VMware Inc.
RabbitMQ is a messaging server that just works

Im in yr serverz, queueing yr messagez
We estimate 300-500 or more in production, >20K in dev
Using RabbitMQ is easy

RabbitMQ integrated with more than 100 developer platforms

Rabbit distributed by most Linux type platforms

RabbitMQ.NET and Microsoft IVA partnership

We offer packages for many other OS platforms
Using RabbitMQ is easy

RabbitMQ integrated with more than 100 developer platforms

Rabbit distributed by most Linux type platforms

We offer packages for many other OS platforms

Cloud messaging is the future
Cloud has re-energised IT
Cloud matters because we operate at real time on a global scale

- Mobile, video, presence, live streams - anywhere, anytime
- How to deliver this without lock-in to any one platform or location?
- Massive amounts of data everywhere and changing all the time
- Security, privacy, consistency of customer experience eg latency, relevance

- Cloud Messaging matters!
Generating Thousands of PDFs on EC2 with Ruby

December 23rd 2009 by Sean Cribbs

The Problem

For about two months, we’ve been working on a static website that exposes the results of complicated economics model to non-economists. We decided to make the site static because of the overhead involved in computing the results and the proprietary nature of the model. We would simply pre-generate the output for all valid permutations of the inputs. The visitor could then choose her inputs from a questionnaire, click a button and immediately be shown the results.

The caveat of this decision is that in addition to the numerical outputs, three graphs and a summary (both in HTML and PDF) would need to be generated for each permutation. Since there were 3600 permutations, this would amount to 18000 files in total! Initial local runs of our generation process took about 30 seconds for each permutation, mostly due to embedding the graph images into the PDF. On a single machine, that would take 30 hours of uninterrupted processing! Clearly, this was a job for “the cloud”.

Generating Thousands of PDFs on EC2 with Ruby

December 23rd 2009

The Problem

For about two months, we have been using a one-server economics model to normalize three sets of data. We involved in computing the data to generate the output for a report from a questionnaire, which takes about 30 seconds for each request on a single machine, that was "in the cloud".

The caveat of this decision (both in HTML and PDF) was that permutations, this would be done on about 30 seconds for each request on a single machine, that was "in the cloud".

```
#!/usr/bin/env ruby

$: << File.expand_path(File.join(File.dirname(__FILE__), '..', 'lib'))
require 'rubygems'
require 'eventmachine'
require 'mq'
require 'custom_libraries'

Signal.trap('INT') { AMQP.stop
  EM.stop }
Signal.trap('TERM') { AMQP.stop
  EM.stop }

AMQP.start(:host => ARGV.shift) do
  MQ.prefetch(1)
  MQ.queue('jobs').bind(MQ.direct('jobs')).subscribe do |header, body|
    GenerationJob.new(body).generate
  end
end
```

Basically, it connects to the RabbitMQ host specified on the command line, subscribes to the job queue, and starts processing messages.
Cloud + Queues makes it easy to manage an ARMY of worker bots
We also care about SCALE ...
We also care about SCALE ... when we cannot predict demand
Cloud applications are component based and need intra-app messaging to scale

See more here: http://aws.typepad.com/aws/2008/12/running-everything-on-aws-soocialcom.html
Inter-cloud applications use inter-application messaging to route and deliver data
Inter-cloud “secure network overlay” - uses RabbitMQ under the covers

See http://www.cohesiveft.com/vpnCubed/
PUBSUB at massive scale: “Twitter for Data in the Cloud”

Ocean Observatories Initiative
See: http://www.oceanobservatories.org/spaces/display/CIDev/Home
Pubsub?  wtf?
At scale, coping with changing data can be a problem
Social applications store data
Social applications store data

And there’s terabytes of it
And there’s terabytes of it

And it’s in the cloud
Tell me when THIS changes
Pubsub wins because Polling Sucks ... “Are we there yet?”, “Are we there yet?”,..

Pubsubhubbub provides web pubsub

Pubsub Hubs are Cloud Messaging Technology

Check out RabbitHub on github for more info/code

Check out the Google team’s video at

http://www.youtube.com/watch?v=B5kHx0rGkec
Monitoring done wrong

He’s probably not seeing a whole lot at this point.
Monitoring done right
Monitoring done right

See: Ben Hyde’s post at http://enthusiasm.cozy.org/archives/2009/02/listening-to-the-system
Monitoring done right
Monitoring done right

Lots more event processing goodness here: http://blog.cloudeventprocessing.com/
Pubsub gets you Monitoring - and if you add Queues you get Logging
Recording information is of fundamental human value

Twitter hits Library of Congress: Would Founding Fathers tweet?

The Library of Congress plan to archive all communication on Twitter has us wondering how history would sound if Twitter had been around in 1776.

Tweet? Then-presidential candidate Barack Obama checks his wireless device March 20, 2008, before a town hall meeting in West Virginia. The Library of Congress will archive all content posted to Twitter, it announced Thursday.

Alex Brandon/AP/File

copyright (c) VMware Inc.
So what is messaging?
messaging is ‘data in motion’ and it is really important
and ‘the move to cloud computing’ will put messaging everywhere
There are LOTS of cloud messaging technologies!

<table>
<thead>
<tr>
<th>SMTP, HTTP, XMPP, AMQP</th>
<th>Messages</th>
<th>Email, Chat, Comet, BOSH, ..</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMQP, HTTP PSHB, XMPP</td>
<td>Pubsub</td>
<td>SMS, Twitter, ..</td>
</tr>
<tr>
<td>AMQP, (XMPP)</td>
<td>Queues</td>
<td>Trading, Cash transfer</td>
</tr>
<tr>
<td>AMQP, NoSQL, SQL</td>
<td>Log Storage</td>
<td>Routing, transforming, alerting</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eg SNS, ESBs and Esper CEP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Buffering, caching, filtering</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eg SQS, Redis</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Archiving + search</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eg Riak, Cassandra</td>
</tr>
</tbody>
</table>
... and RabbitMQ’s goal is to make all of this really simple

Im in yr serverz, queueing yr messagez
RabbitMQ

Towards a PLATFORM for Cloud Messaging
If cloud requires a Cloud OS, then is messaging the Cloud Bus?
For Infra, yes. Many clouds, eg NASA Nebula, have a RabbitMQ cloud bus:

Use cases: Inter-VM chatter
And lots of Management:
Request capacity
Start 100 VMs. Shut them down.
“I am overloaded”
“Find me - who am I?”
“Move me”
“Back me up”
Scale back capacity
That’s great but we also need a PLATFORM if we are to deploy and run apps

- **SaaS**
  - cloud applications

- **PaaS**
  - eg RabbitMQ on Heroku

- **IaaS**
  - eg Rabbit at NASA
Platform as a service example: Heroku!

SaaS
cloud applications

PaaS
eg RabbitMQ on Heroku

IaaS
eg Rabbit at NASA
Conceptually: a platform represents freedom from complexity

IaaS infra is hidden

Scalable cloud bus - run as many as you like
(Rabbit is open source and can run on any cloud as its ‘central nervous system’)

Copyright (c) VMware Inc.

Friday, 23 April 2010
Pubsub is made simple - like apps tweeting at each other!

"hey"

"hey"

"hey"

"hey"
Queues are easy too
Queues are easy too

“hey”
By being 100% open - we can also do intercloud just as simply!

<table>
<thead>
<tr>
<th></th>
<th>Intra-App</th>
<th>Intra-Cloud</th>
<th>Inter-Cloud</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PaaS</strong></td>
<td>Rabbit Cloud Service</td>
<td>Rabbit Cloud Service</td>
<td>Rabbit Cross-cloud Services</td>
</tr>
<tr>
<td><strong>IaaS</strong></td>
<td>Rabbit inside the Cloud VM</td>
<td>RabbitMQ as infra: “Cloud Bus”</td>
<td>Federated RabbitMQ</td>
</tr>
</tbody>
</table>
To bridge private and public clouds, open source and open standards are best

SAY NO TO CLOUD LOCK IN

(C) Giara @flickr.com
Summary - clouds break the link between application and location

- Can’t predict where apps and VMs are running
- Can’t predict scale

- Design for DATA IN MOTION
- In motion VMs need PUBSUB and ROUTING like with cell phones
- VMs suspended or offline need QUEUES
- Messaging lets you ‘dial up’ VMs and broadcast changes safely
Solution - RabbitMQ is the open cloud messaging platform

- Fantastic community support for messaging intra app
- Language neutral open protocols for messaging inter app
- Federation - bridge private and public ‘intercloud’
- Cloud Bus for IaaS
- Messaging PaaS
Solution - RabbitMQ as an open cloud messaging platform

- We’ll manage messaging for you
- Scalable, easy to use
- Completely portable - no lock in
- Pre-provisioned - zero install, pay as you go, self-service
- Latency is low: Application logic and data are CO-LOCATED

WIN
Rich Wolski, CTO and Founder, Eucalyptus

“pretty much everything you own is going to be trying to send you data”

Stefan Norberg, Chief Architect, Unibet

“If you remove the need to invest in infrastructure, the need to train people on the operational aspects and then get excellent scalability and low latency guaranteed by contract, I’d buy it in a second. Who will provide me with the Real Time Web as a service?”
Thank-you

Email: alexis@rabbitmq.com

Get started: http://www.rabbitmq.com/how.html