Introduction to RabbitMQ
An open source message broker that just works

Alexis Richardson
Matthias Radestock
Tony Garnock-Jones
CohesiveFT, LShift and RabbitMQ

Google UK
25 September 2008
RabbitMQ is a messaging server that just works!

Im in yr serverz, queueing yr messagez

Photo credit: http://flickr.com/photos/53366513@N00/67046506/
copyright (c) Rabbit Technologies Ltd.
You might need messaging if ... you need to scale
You might need messaging if ... you need to monitor data feeds
You might need messaging if ... you need a message delivered responsibly
You might need messaging if ... you need things done in order
You might need messaging if ... you are using the cloud
Messaging is everywhere

- Service Bus
- Peer Network
- Federation and Cloud
- Client - Server (hub)
- Pipeline
Messaging is your friend

- Route data from point A to point B (or “pubsub” push to many points C)
- Decouple publishers and consumers
- Queueing and buffering for later delivery
- Asynchronous “hand off”
- Load balancing and scalability
- Monitoring and management

For more on messaging, see this great summary by Bob Pasker: http://blog.pasker.net/2008/06/16/you-might-need-messaging-if/
Don’t be evil
When middleware goes bad

complex, proprietary, closed

requires installation and customisation

integration services from consultants with knowledge of many platforms or languages

then maintenance is done by the customer

which is then followed by system aging, bloat, and eventual heat death
Beware of lock in
Messaging middleware market is stuck, stuck, stuck
What about everyone else’s needs?

integration

cloud and virtualization

web applications and services
Meet the good guys
OPEN INTERNET PROTOCOLS - TCP, SCTP, HTTP, SMTP - EPIC WIN

- simple
- standard
- ubiquitous substrate
- no customisation needed
- no integration required from consultants
- maintenance is done by the vendor
- proven to outlast the lifetime of the average software company
- (and many banks)
- scales
The world is getting more open every day

Then:

Imagine if we had no TCP and had to use ‘IBM NetSphere’
Imagine if we had no HTTP and had to use ‘Microsoft Home Network’
Imagine if we had no SMTP email and had to pay per message like SWIFT

Now:

Imagine if we had no XMPP chat and had to use .. oh, wait a minute :-(
AMQP - business messaging - like email but you can send money over it
## Application layer protocols made simple

<table>
<thead>
<tr>
<th>async</th>
<th>SMTP</th>
<th>?</th>
</tr>
</thead>
<tbody>
<tr>
<td>sync</td>
<td>HTTP</td>
<td>IIOP</td>
</tr>
</tbody>
</table>

What goes in here will clean up if it is OPEN, UBIQUITOUS, & ADAPTABLE.
Some key AMQP messaging protocol requirements

- Internet protocol - like HTTP, TCP - but ASYNCHRONOUS
- WHERE TO SEND MESSAGES (Routing)
- HOW TO GET THERE (Delivery)
- WHAT GOES IN MUST COME OUT (Fidelity)
AMQP in a nutshell

Publish/Subscribe
- detect

Messaging
- transact

File Transfer
- report
AMQP lets you program message flows dynamically

Each message is stateless

Consumers create queues; these buffer messages for push to consumers

Queues are stateful, ordered, and can be persistent, transient, private, shared.

Exchanges are stateless routing tables.

Consumers tell queues to bind to named exchanges; each binding has a pattern e.g. “tony” or “*.ibm.*”

Producers send messages to exchanges with a routing key e.g. “tony”, or ordered set of keys e.g. “buy.ibm.nyse”

Exchanges route messages to queues whose binding pattern matches the message routing key or keys
Evan and Anders want to follow what Tony says. They can follow Tony by binding their queues to a RabbitMQ exchange, using the pattern “tony”.

Tony publishes the message “is at work” to the same RabbitMQ exchange, using the routing key “tony”.

The exchange updates Evan’s and Anders’ queues accordingly, for subsequent consumption by their client applications.

Many other patterns are possible e.g. for filtering by topic similar to this: http://jchris.mfdz.com/posts/64
Producers and consumers logically interact through a broker cloud
Critical path == logical path
Developed by a Working Group of Users (yay!) as well as Vendors (boo....)
RabbitMQ is an implementation of AMQP, the emerging standard for high performance enterprise messaging.

**Features**

- A complete, conformant and interoperable implementation of the published AMQP specification
- Based on a proven platform, offering exceptionally high reliability, availability and scalability
- Good throughput and latency performance that is predictable and consistent
- Compact, easily maintainable code base, for rapid customisation and hot deployment
- Extensive facilities for management, monitoring, control and debugging
- Licensed under the open source Mozilla Public License

**Distribution**

- RabbitMQ server, written on top of the widely-used Open Telecom Platform
- RabbitMQ Java client with API guide
- RabbitMQ .NET/C# client, with support for WCF
- Experimental bindings supporting HTTP, STOMP, SMTP, POP3, ...
- Platform-neutral distribution, plus platform-specific packages and bundles for easy installation
- Several user-contributed packages that extend the core RabbitMQ functionality
- Extensive documentation, several demos and examples, and a functional/performance test suite
- [Download Now!](#)

**News**

- The RabbitMQ team is pleased to announce the release of RabbitMQ 1.4.0.
  - This release has beta status, and includes bug fixes, packaging enhancements, and performance improvements. Further details are available [here](#).
RabbitMQ - NOM NOM NOM NOM

“In my experience, you can have a clustered rabbitmq setup running at home in under 20 minutes. It's all in the admin guide.” - Steve Jenson, co-founder of Blogger

“RabbitMQ is a pleasure to use and it just works. Everyday, every time, every message” - Michael Arnoldus, project lead, algo trading firm

RabbitMQ is an implementation of AMQP, the emerging standard for high performance enterprise messaging.

Features
- A complete, conformant and interoperable implementation of the published AMQP specification
- Extensive facilities for management, monitoring, control and debugging

Distribution
- RabbitMQ server, written on top of the widely-used Open Telecom Platform
- Several user-contributed packages that extend the core RabbitMQ functionality

Further details are available [here].

copyright (c) Rabbit Technologies Ltd.
RabbitMQ is for everyone
RabbitMQ is fast
STOP - LOOK - LISTEN - THINK

(C) Javic @flickr.com
Clustered, highly available messaging is complex - don’t build this at home

Get RabbitMQ - it takes a couple of minutes to set up and JUST WORKS

RabbitMQ began in 2006 - first release Feb 2007 - four more releases since

Complete, conformant and interoperable implementation of the AMQP spec

RabbitMQ is FREE to use - open source MPL license - prolific

SUPPORTED commercially

“Ready to run” bundles install in minutes on most platforms (and the cloud)

Several extensions - HTTP, STOMP, XMPP, ... (PB?)
Show me some !!@ link love

- http://www.rabbitmq.com (product, documentation and mailing list)
- http://hg.rabbitmq.com/ (open source repositories)
- Run RabbitMQ right now on EC2 or a VM: http://es.cohesiveft.com/site/rabbitmq
- Jump page for Ruby and Python fans: http://github.com/tmm1/amqp/tree/master
- Introduction to AMQP, use cases and RabbitMQ community, from Dmitriy Samovskiy of the CohesiveFT Elastic Server team: http://www.slideshare.net/somic/introduction-to-amqp-messaging-with-rabbitmq/
- Infovore: http://del.icio.us/alexisrichardson/rabbitmq
- What people are saying: http://search.twitter.com/search?q=rabbitmq and IRC #rabbitmq on freenode
- RabbitMQ blogs: http://www.lshift.net/blog/category/lshift-sw/rabbitmq/ has detail from today on AMQP, erlang, and XMPP. Then http://hopper.squarespace.com/blog/category/amqp has many client examples
Join the project!