# MRG - AMQP trading system in a rack

Carl Trieloff
Senior Consulting Software Engineer/ Director MRG
Red Hat, Inc.

# Trading system in a rack...

We will cover a generic use case of a trading system in a rack, showing a few common patterns that have been used by set of customers with MRG to build AMQP based trading systems. Use case will briefly cover:

- Generally two approaches
  - Highly available, fully active market slices
  - Federated, non-active market slices
- Market data systematic using standard AMQP clients
  - LVQ – Last value queues (SYMBOL DATA)
  - Reliable transfers - (ORDERS)
  - Rings – (MARKET CACHES)
  - Sequencing
- Achieving market data throughput, latency, and order reliability
  - Latency, and throughput, tuning, etc
  - Disaster recover replication, flow control, etc
- Managing it...
  - Tools, consoles, QMF (Management over AMQP)

# View of market data slice with MRG

*-- AMQP based trading system deployment --*

Collocated trading engine
*-- your code/logic --*

MRG: trading semantics

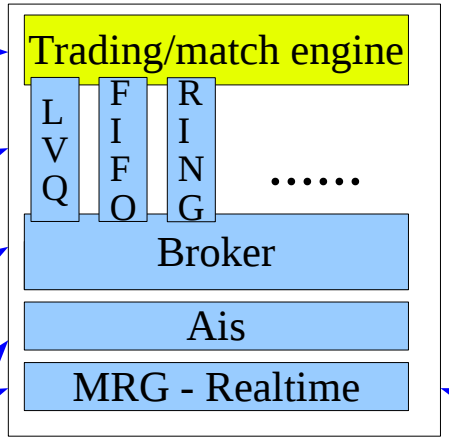
MRG: broker

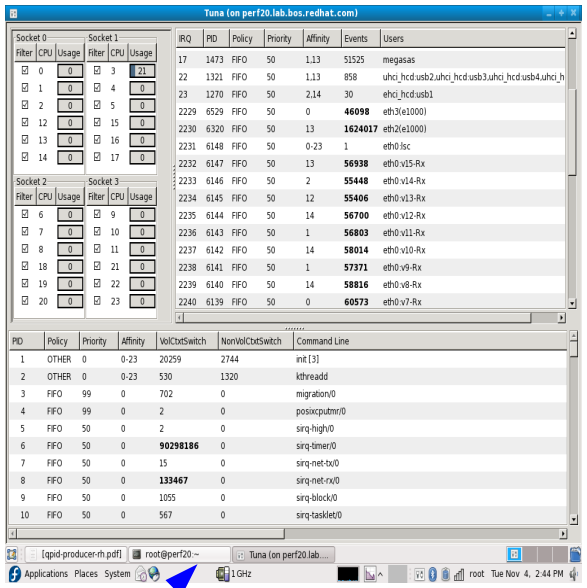
MRG: Realtime

RHEL: Ais – multicast network

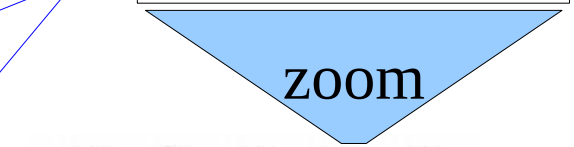
FT cluster, in slices

Separate networks for orders/ symbols etc

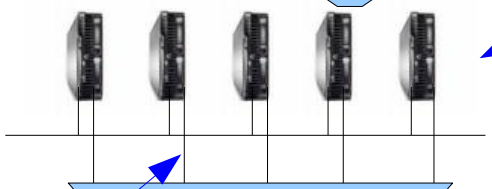


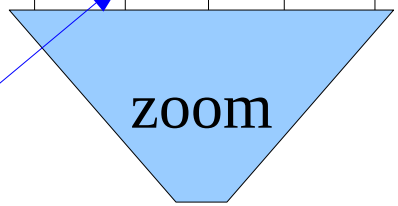




Tune it – MRG Tuna




MRG: Active, Active or Federated slice

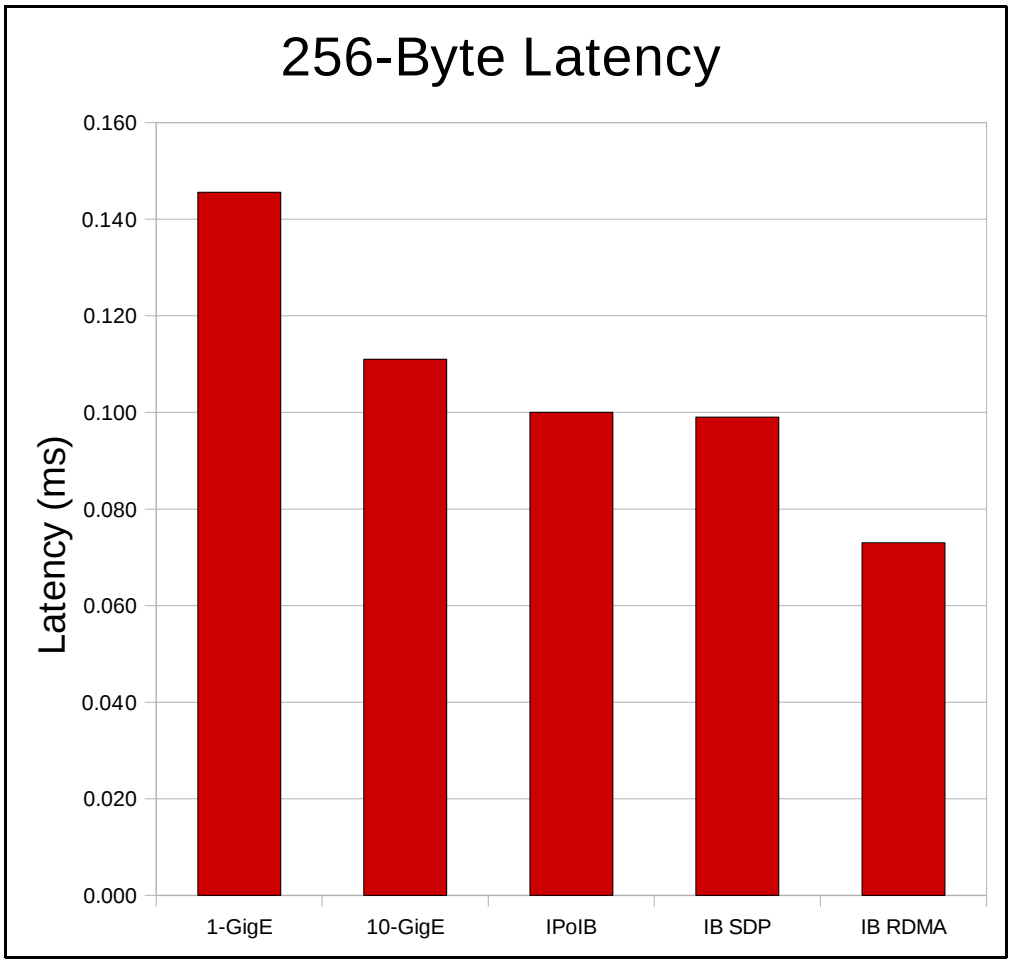


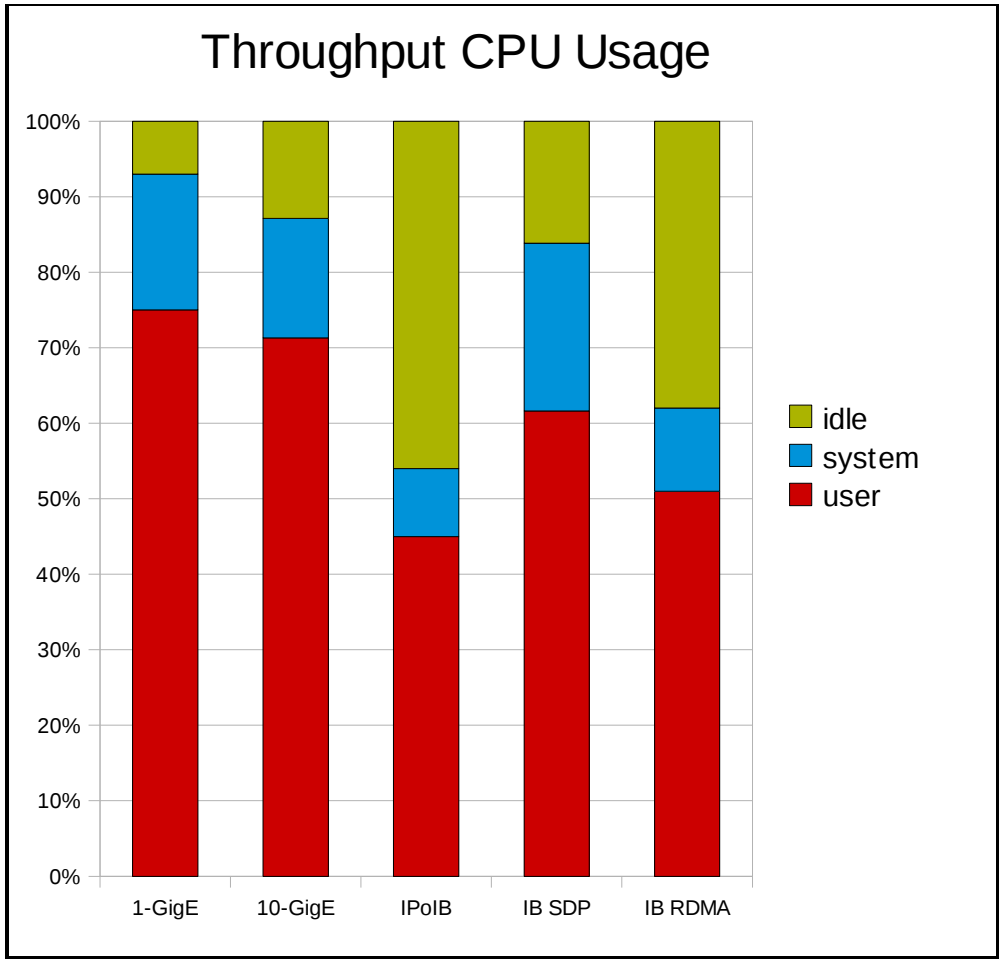



MRG: DR replication

# Selecting the network fabric:

Comparing Latency per technology, per CPU cost at full load.




All measurements are AMQP between 3 peers (brokered) and fully reliable

# Comment from a MRG Market data customer

*"After following for few years the progress of the open standard messaging AMQP development, our company was excited to see Red Hat's contribution to the Qpid open source effort in farther developing the messaging product. Their resulting messaging product (MRG) allows our company to deliver a mission critical trading service leveraging messaging features intrinsic to financial workflows and providing outstanding performance."* - a MRG market data customer

http://www.redhat.com/mrg
Or come interact at Qpid
http://qpid.apache.org

redhat

**AMQP is the emerging open standard for Messaging Middleware**
**Join us as at the AMQP Conference as we present AMQP 1.0 Public Review**

✓ **Learn** about AMQP directly from Working Group members
✓ **Explore** the motivations and objectives of the protocol
✓ Get the **inside track** with detailed exploration of the technology
✓ **Share** your thoughts with people who implement AMQP technology
✓ Join us at the **evening reception** over refreshments and get to know the **community**

**1st April, 2009 – 1pm to 8pm**
**University of California in San Diego, USA.**

**To reserve your place at this free event, please email amqp.event@gmail.com with your name, email address and contact telephone number to receive details and joining instructions.**

**AMQP Working Group – www.amqp.org**
Cisco Systems, Credit Suisse, Deutsche Börse Systems, Envoy Technologies, Goldman Sachs, iMatix, IONA, JPMorgan, Microsoft, Novell, Rabbit Technologies, Red Hat, Tervela, TWIST, WSO2, 29West


ADVANCED MESSAGE QUEUING PROTOCOL

## Internet Protocol for Business Messaging

# AMQP in Action

**By members of the AMQP Working Group**
**QCon London, March 2009**

Cisco Systems
Credit Suisse
Deutsche Börse Systems
Envoy Technologies
Goldman Sachs
iMatix
IONA
JPMorgan Chase
Microsoft
Novell
Rabbit Technologies
Red Hat
Tervela
TWIST
WSO2
29West


# Who's talking today

## John O'Hara, JPMorgan

- Started AMQP, Chair of AMQP Working Group
- JPMorgan Senior Architect and Distinguished Engineer

## Pieter Hintjens, iMatix

- CEO iMatix and creator of OpenAMQ and 0MQ

## Alexis Richardson, CohesiveFT

- Director of CohesiveFT who sponsor Rabbit MQ

## Carl Trieloff, Red Hat

- Senior Consulting Engineer, Director for Red Hat MRG
- Chair of Apache Qpid project

# AMQP was born of frustration

## MOM needs to be *everywhere* to be useful

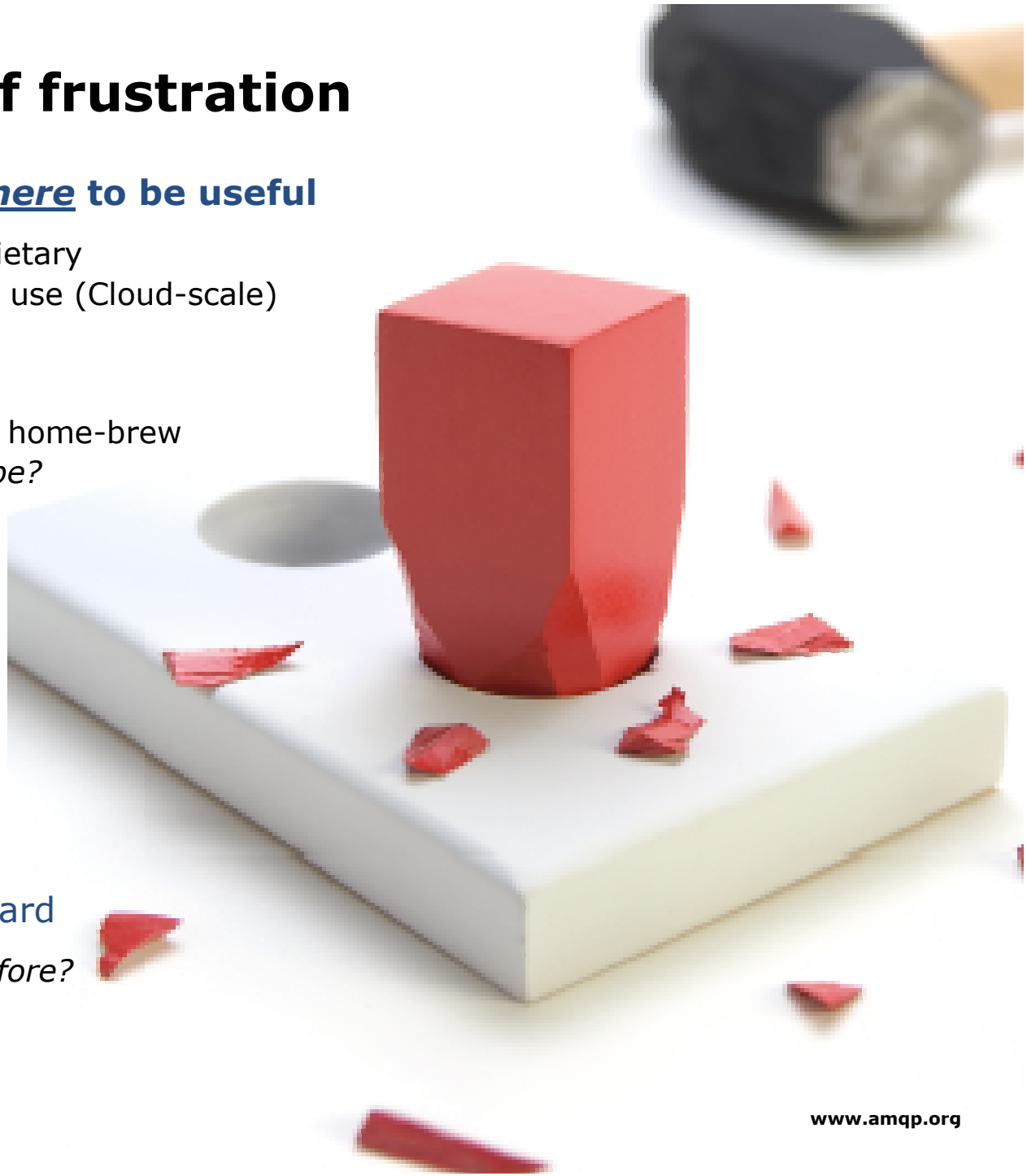
- dominant solutions are proprietary
  - too expensive for everyday use (Cloud-scale)
  - they don't interoperate
  - incumbents stagnating
- has resulted in lots of ad-hoc home-brew
  - *how hard can middleware be?*

### Middleware Hell

- 100's of applications
- 10,000's of links
- every connection different
- massive waste of effort

### The Internet's missing standard

- *Why has no one done this before?*



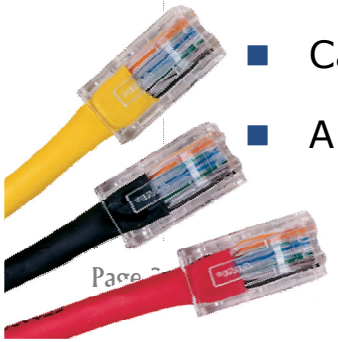
# AMQP Capabilities

*An **Internet Protocol** for **Business Messaging***

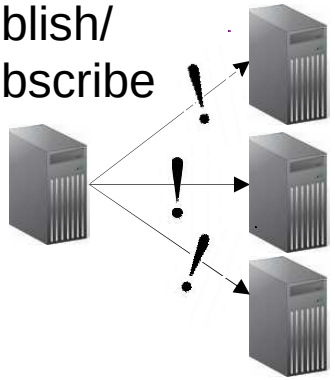
- A Protocol (not an API) for maximum interoperability
- Queuing with strong Delivery Assurances
- Event distribution with Flexible Routing
- Large Message capability (gigabytes)
- Global Addressing Scheme (email-like)
- Meet common requirements of mission-critical systems
  - Robust, available, scalable, secure, resilient
  - Aims to be stable over the long run
  - **Platform agnostic and totally open**

AMQP delivers this in one handy protocol

- Candidate for a common infrastructure for the enterprise
- A foundation for other protocols and products

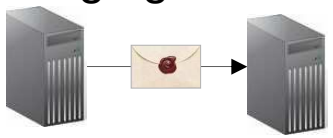


Publish/
Subscribe



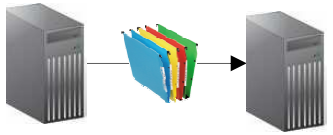
***detect***

Messaging



***transact***

File Transfer



***report***

# AMQP Roadmap

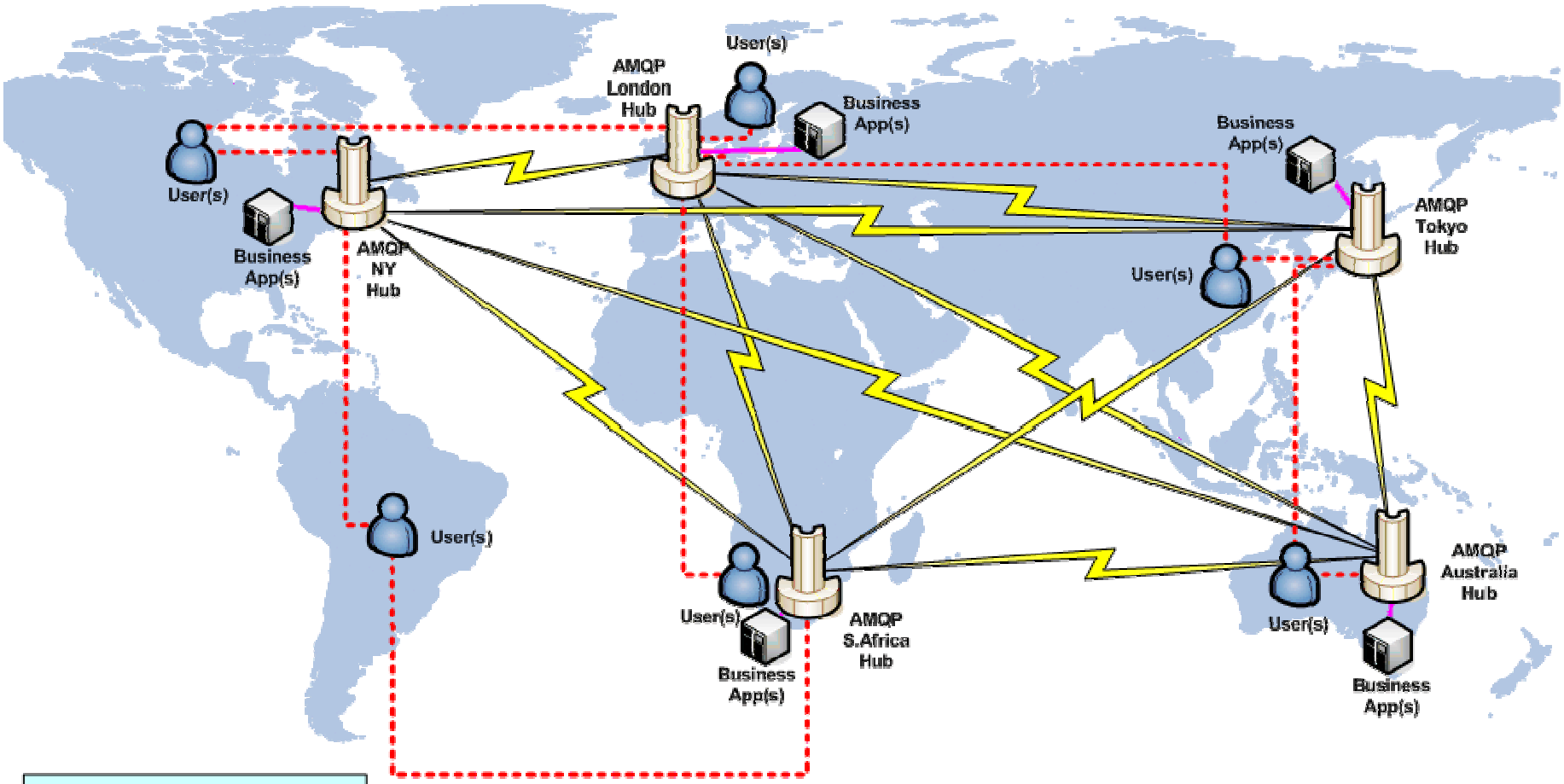
- **AMQP 1.0 this year**

- Key features
  - Simplified Architecture
  - Stronger Transaction Model
  - Global Addressing
    E.g. "forex@jpmorgan.com"

- Current implementations are AMQP v0-91 and v0-10
  - Implementer commitment to painless 1.0 migration
  - 1.0 will be finalised only after extensive inter-op testing
  - Now is the right time to embrace AMQP

- **Join us at the AMQP Conference**
  - Presenting AMQP 1.0 Public Review
  - April 1, University of California at San Diego
  - **www.amqp.org for details**

# "High touch" global trading application




- Hundreds of client GUI's subscribe and unsubscribe to only update the visible screen as users scroll through thousands of stocks!
- Routes messages based on header properties – not a topic tree
- Runs across Windows, Linux, Solaris; C++, Java, .NET
- 1 billion messages per day; live for 3 years