



DEVNATION

April 13-17, 2014 San Francisco, California



DEVNATION April 13-17, 2014
San Francisco, California

Resilient Messaging with JBoss A-MQ

Scott McCarty

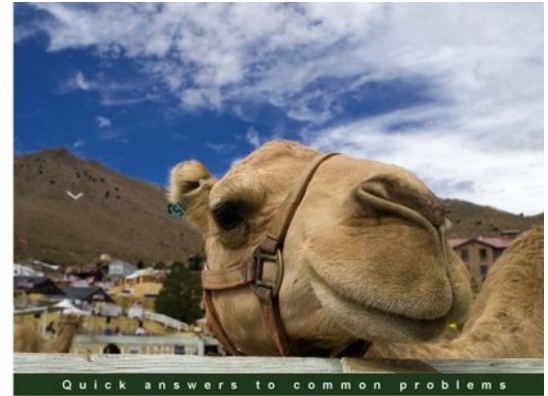
Scott Cranton

Agenda

- Quick Overview of JBoss A-MQ
- JBoss A-MQ HA with Fabric8
- Quick Overview of Red Hat High Availability
- JBoss A-MQ HA with Red Hat High Availability

@scottcranton

- 20+ years in Middleware software coding and sales
- Apache Camel Developer's Cookbook – Dec 2013
- FuseSource World Wide Solution Architect manager; joined FuseSource in 2009
- 5+ years helping companies use Fuse and A-MQ



Apache Camel Developer's Cookbook

Solve common integration tasks with over 100 easily accessible
Apache Camel recipes

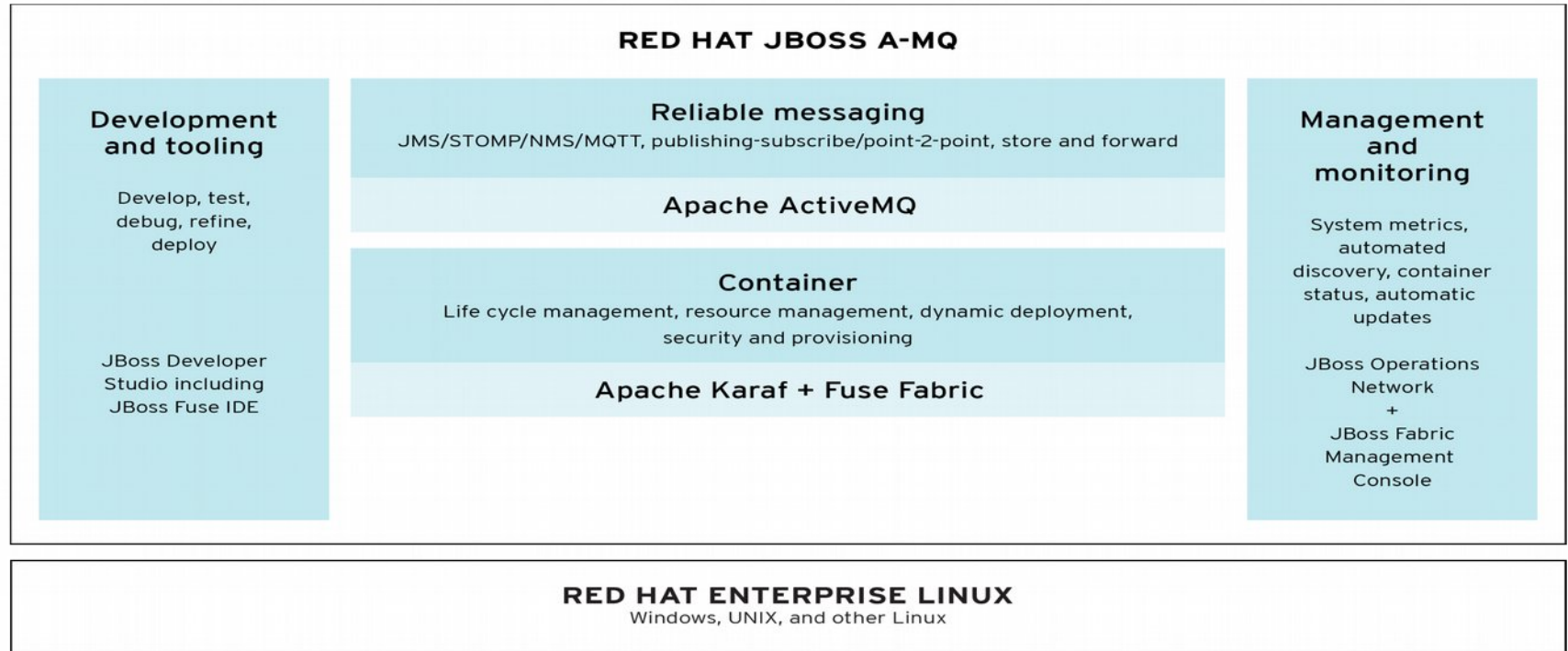
Scott Cranton

Jakub Korab

[PACKT] enterprise⁸⁸
PUBLISHING

Red Hat JBoss A-MQ

Small-footprint, high-performance, open source **messaging platform**



JB0010

Red Hat JBoss Integration Portfolio

Red Hat JBoss Fuse Service Works
adds Design and Runtime Governance

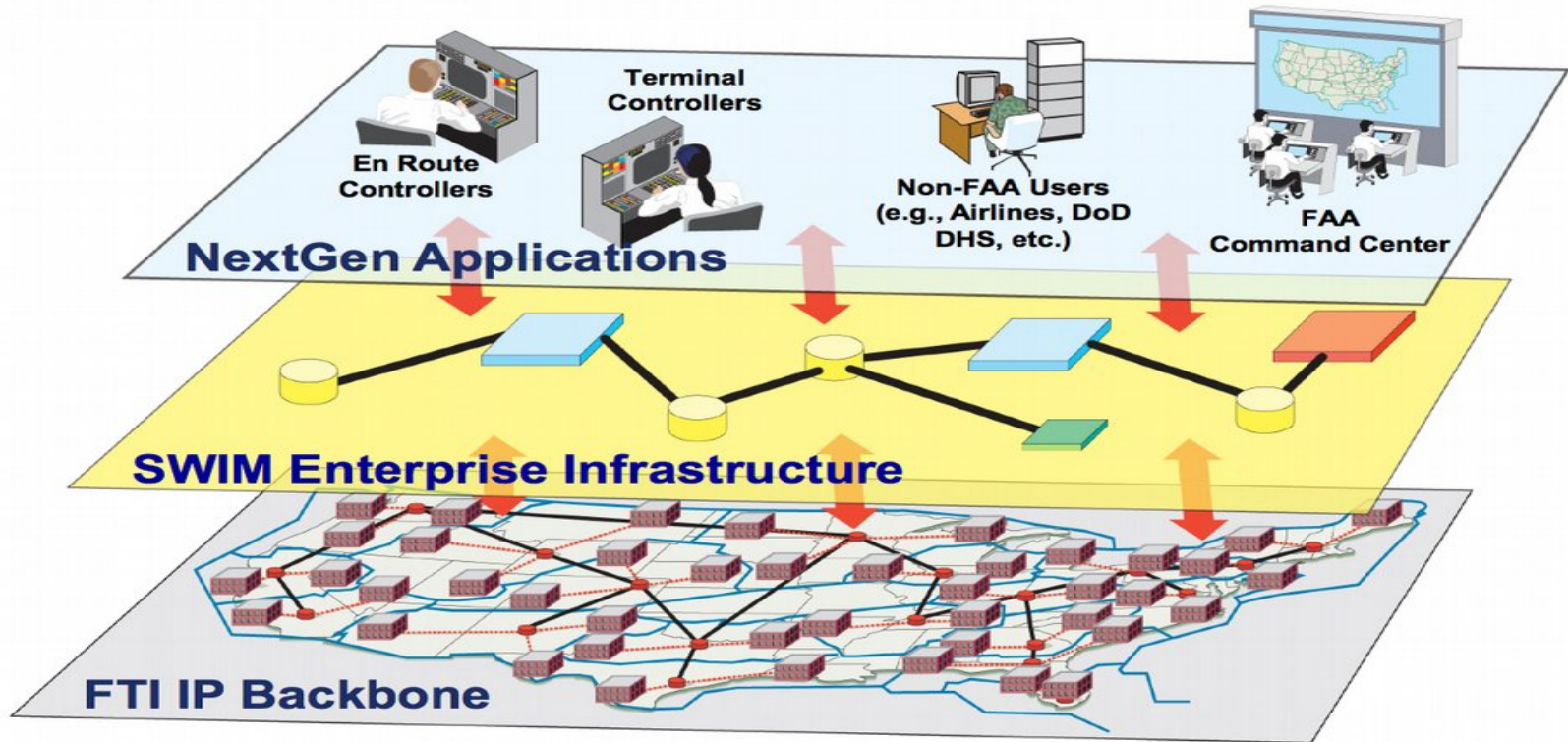
Red Hat JBoss Fuse
adds Protocol Mediation and Routing

Red Hat JBoss A-MQ
Reliable Messaging

Federal Aviation Administration (FAA) uses JBoss A-MQ and Fuse

- *Did you fly to DevNation? JBoss A-MQ helped you get here...*
- 35,000 controllers managing 7,000 takeoffs and landings an hour, and responsible for 50,000 aircraft in national airspace every day
- SWIM Program (System Wide Information Management) streamlining data exchange between FAA, industry, and airline partners; facilitating next generation applications...
- In production across 20 data centers nationally distributing data like: Corridor and terminal weather systems, flight data, control tower events, and runway visual range

State of the System



What is JBoss A-MQ?

Open

Scalable

Reliable

*assuming you already know it does reliable messaging

JBoss A-MQ :: Open

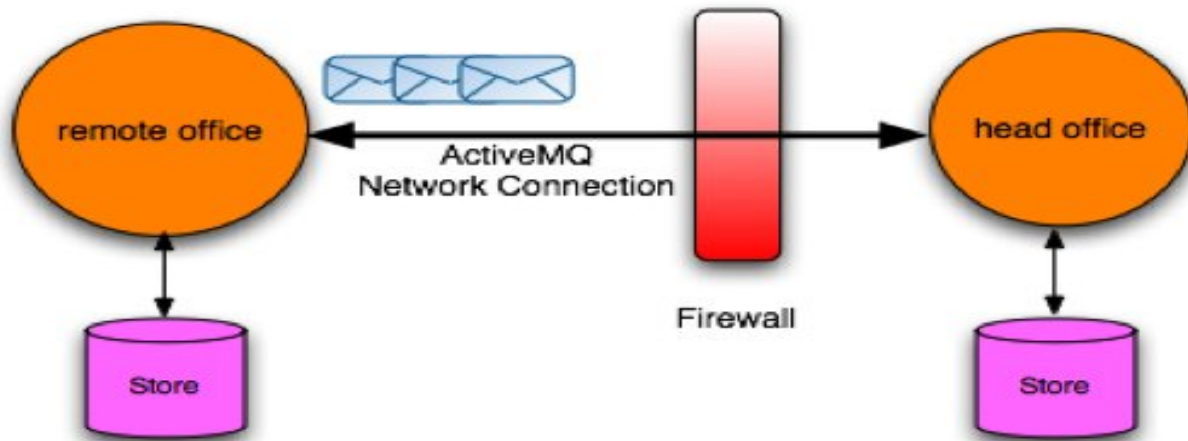
- Open Source
 - Apache Software License 2.0 licensed
 - Based on very popular Apache ActiveMQ
- Open Protocol
 - AMQP 1.0, MQTT, STOMP, OpenWire, ...
- Open Language (Polyglot)
 - Client native languages: Java (JMS), C/C++, .NET
 - STOMP clients: Ruby, JavaScript, Perl, Python, PHP, ActionScript, ...

JBoss A-MQ :: Scalable

- Vertical Scaling
 - Limited primarily by disk and network throughput
- Horizontal Scaling
 - Network of Brokers (Clustering / Federation)
 - Fabric8 (<http://fabric8.io>) for scale out management
 - Central configuration management and provisioning
 - Client-side discovery, load balancing, and failover

JBoss A-MQ :: Scalable

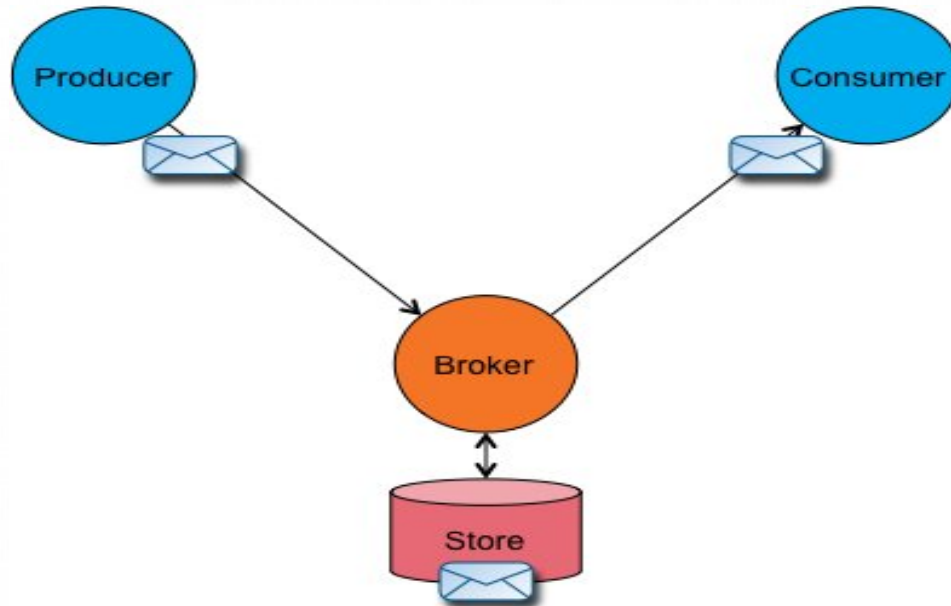
Network of Brokers



JBoss A-MQ :: Reliable

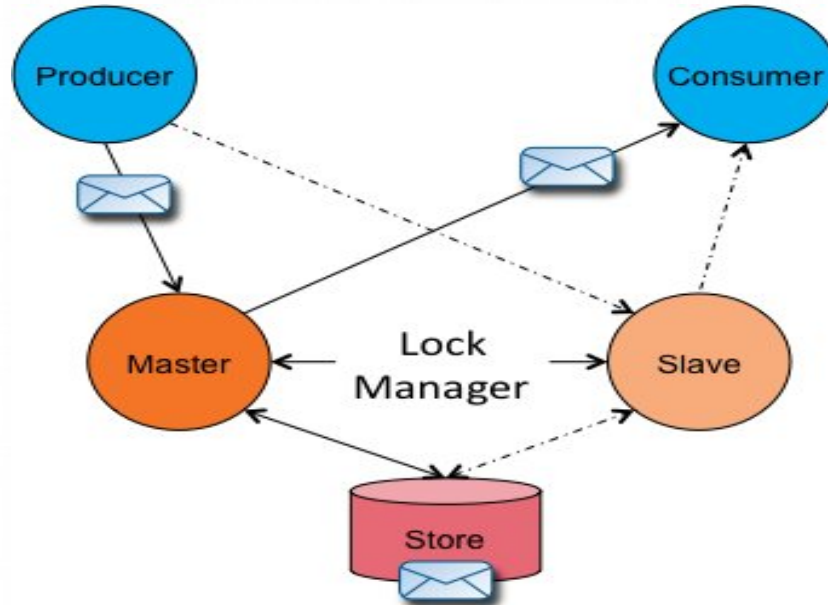
- Persistent Messaging (single node)
 - Store on File System or RDBMS
 - Survive restart, and process failure
- Master / Slave (Active / Passive)
 - Uses included or external Lock Manager
 - Shared Storage - SAN/GFS2 or NFS v4 or RDBMS
 - Replicated - Block or RDBMS replication
- Managed
 - Red Hat Cluster Suite
 - Fuse Fabric with Shared or Replicated storage

JBoss A-MQ :: Reliable

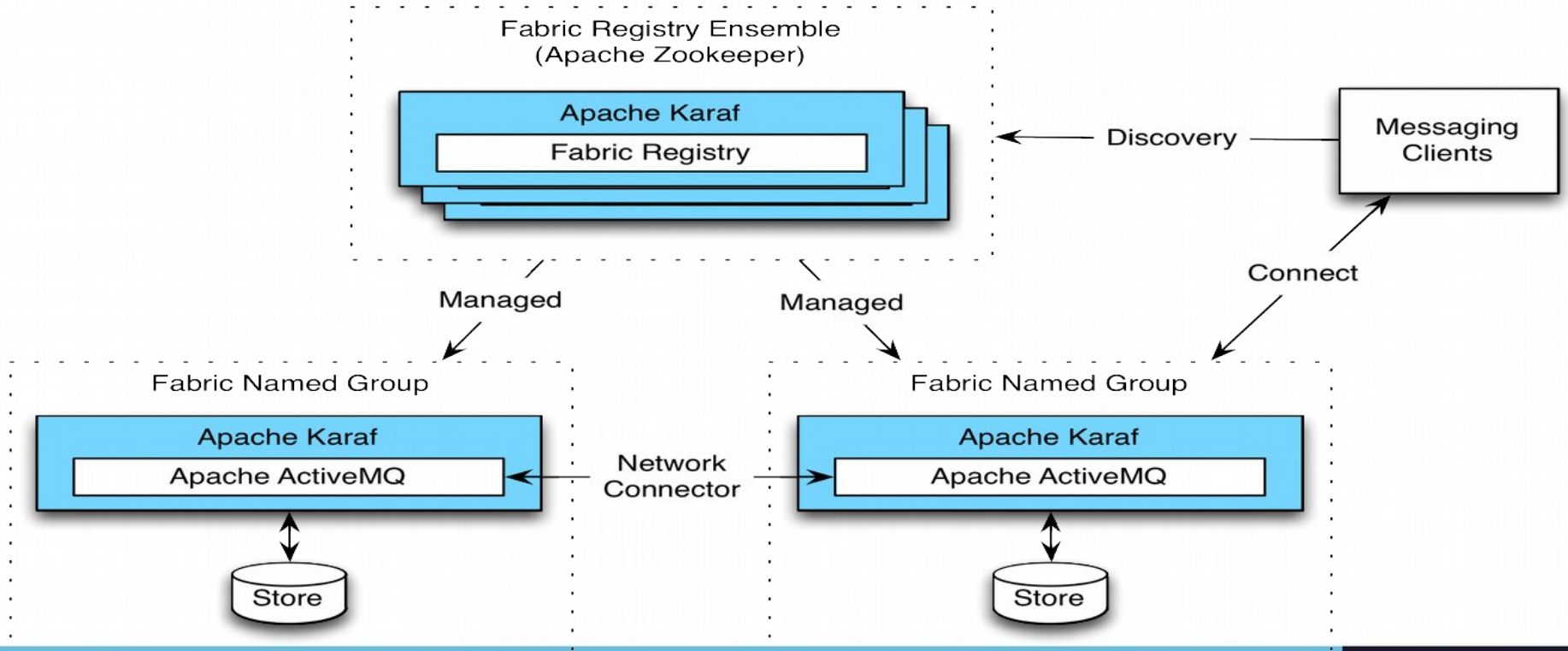


JBoss A-MQ :: Reliable

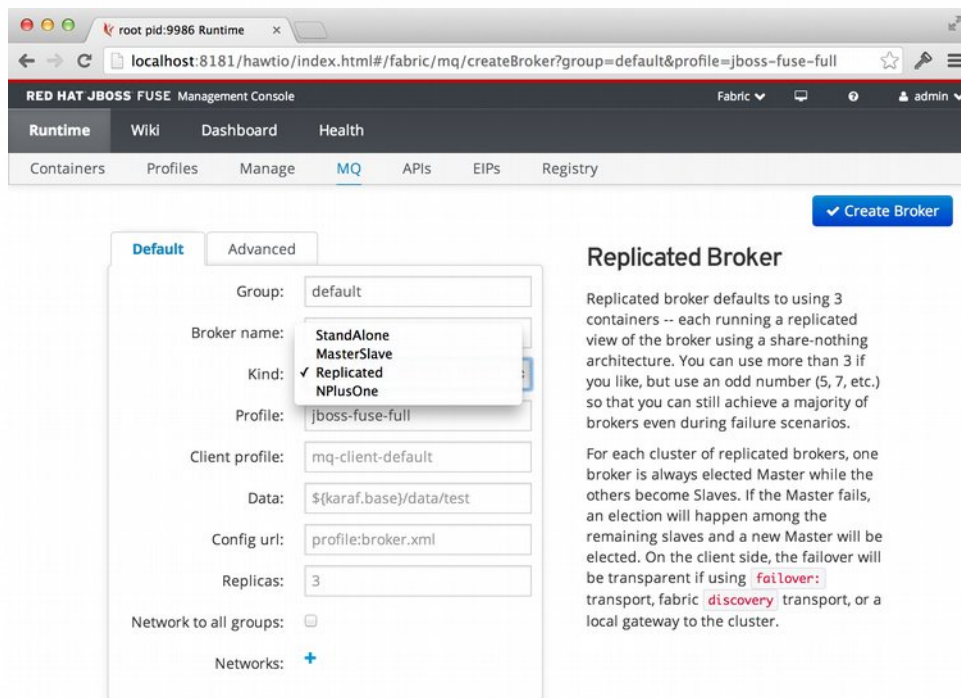
Master / Slave – Shared Storage



Fuse Fabric



Fuse Fabric Management Console



The screenshot shows the Fuse Fabric Management Console interface. The browser address bar displays `localhost:8181/hawtio/index.html#/fabric/mq/createBroker?group=default&profile=jboss-fuse-full`. The page title is "RED HAT JBOSS FUSE Management Console". The navigation menu includes "Runtime", "Wiki", "Dashboard", and "Health". Below this, there are tabs for "Containers", "Profiles", "Manage", "MQ", "APIs", "EIPs", and "Registry". A blue button labeled "Create Broker" is visible in the top right.

The "Default" tab is active, showing a form for creating a broker. The form fields are:

- Group: default
- Broker name: StandAlone, MasterSlave, Replicated (selected), NPlusOne
- Kind: Replicated (selected)
- Profile: jboss-fuse-full
- Client profile: mq-client-default
- Data: \${karaf.base}/data/test
- Config url: profile:broker.xml
- Replicas: 3
- Network to all groups:
- Networks: +

Replicated Broker

Replicated broker defaults to using 3 containers -- each running a replicated view of the broker using a share-nothing architecture. You can use more than 3 if you like, but use an odd number (5, 7, etc.) so that you can still achieve a majority of brokers even during failure scenarios.

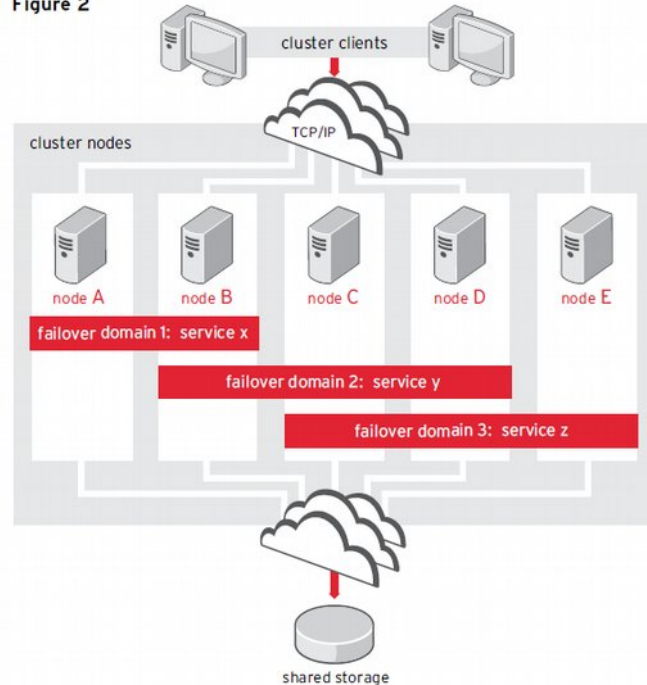
For each cluster of replicated brokers, one broker is always elected Master while the others become Slaves. If the Master fails, an election will happen among the remaining slaves and a new Master will be elected. On the client side, the failover will be transparent if using `failover:` transport, fabric `discovery` transport, or a local gateway to the cluster.

Scott McCarty

- Senior Cloud-Infrastructure Solutions Architect
- Systems Automation, Log Analysis
- Python, Bash, etc
- Clustering, Switching, Routing, Firewall, Load Balancing
- Background in large online properties, and research
- Blog: crunchtools.com

Red Hat High Availability / Clustering

Figure 2



Key Benefits

- General purpose clustering solution: processes, mount points, network addresses
- General purpose management interface
- Failover logic is stored in a technology that manages other high availability resources
- Greater availability of storage choices: EXT3/EXT4, BTRFS, GFS2, NFS
- Deeper availability of health checks: JMX, TCP, custom checks, looking glass services

Concepts

- Nodes: Separate operating system instances in the cluster
- Resources: IP address, process, storage mount
- Failover Domains: Groups of nodes unto which Service Groups can be assigned
- Fence Devices: Integrated Lights Out (ILO), Dell Remote Access Card (DRAC), IPMI

High Availability Components

- Cluster Manager: Ties it all together, calculates quorum, communicates with other cluster components
- Resources: Resource Manager controls starting/stopping of processes, storage mounts, IP addresses, etc
- Fencing: The act of ensuring that broken nodes are removed from the cluster
- Conga Web Interface

Management Interface

The screenshot shows a web browser window titled "Pacemaker/Corosync Configuration - Mozilla Firefox". The address bar shows the URL "https://hideo.tc.redhat.com:2224/manage". The browser's menu bar includes "File", "Edit", "View", "History", "Bookmarks", "Tools", and "Help". The browser's toolbar shows "Most Visited", "Zimbra", "RHN Satellite - Sign In", "My Account Snapshot", "Marriott Hotel Search...", "Search Home", and "Travel Solutions Portal".

The main content area features a dark header with the Red Hat logo, the text "HIGH AVAILABILITY MANAGEMENT", a "Select Cluster" dropdown menu, and a "hacluster" dropdown menu. Below the header, there is a "MANAGE CLUSTERS" section with links for "Remove", "Add Existing", and "Create New". A table lists the cluster "summit" with 3 nodes. To the right, an "INFORMATION ABOUT CLUSTERS" section displays details for the "summit" cluster, including its name and the list of nodes.

	NAME	NODES
<input type="checkbox"/>	summit	3

INFORMATION ABOUT CLUSTERS

Cluster: summit
Nodes: hideo.tc.redhat.com
lady3jane.tc.redhat.com
maelcum.tc.redhat.com

Links

- Technical Solution:
<http://crunchtools.com/resilient-messaging/>
- Cluster Knowledge Base Articles:
<https://access.redhat.com/knowledge/articles/47987>
- Best Practices:
<https://access.redhat.com/knowledge/articles/40051>
- Architecture Review Process:
<https://access.redhat.com/site/solutions/125153>
- Stretch Clustering:
<https://access.redhat.com/knowledge/articles/27136>