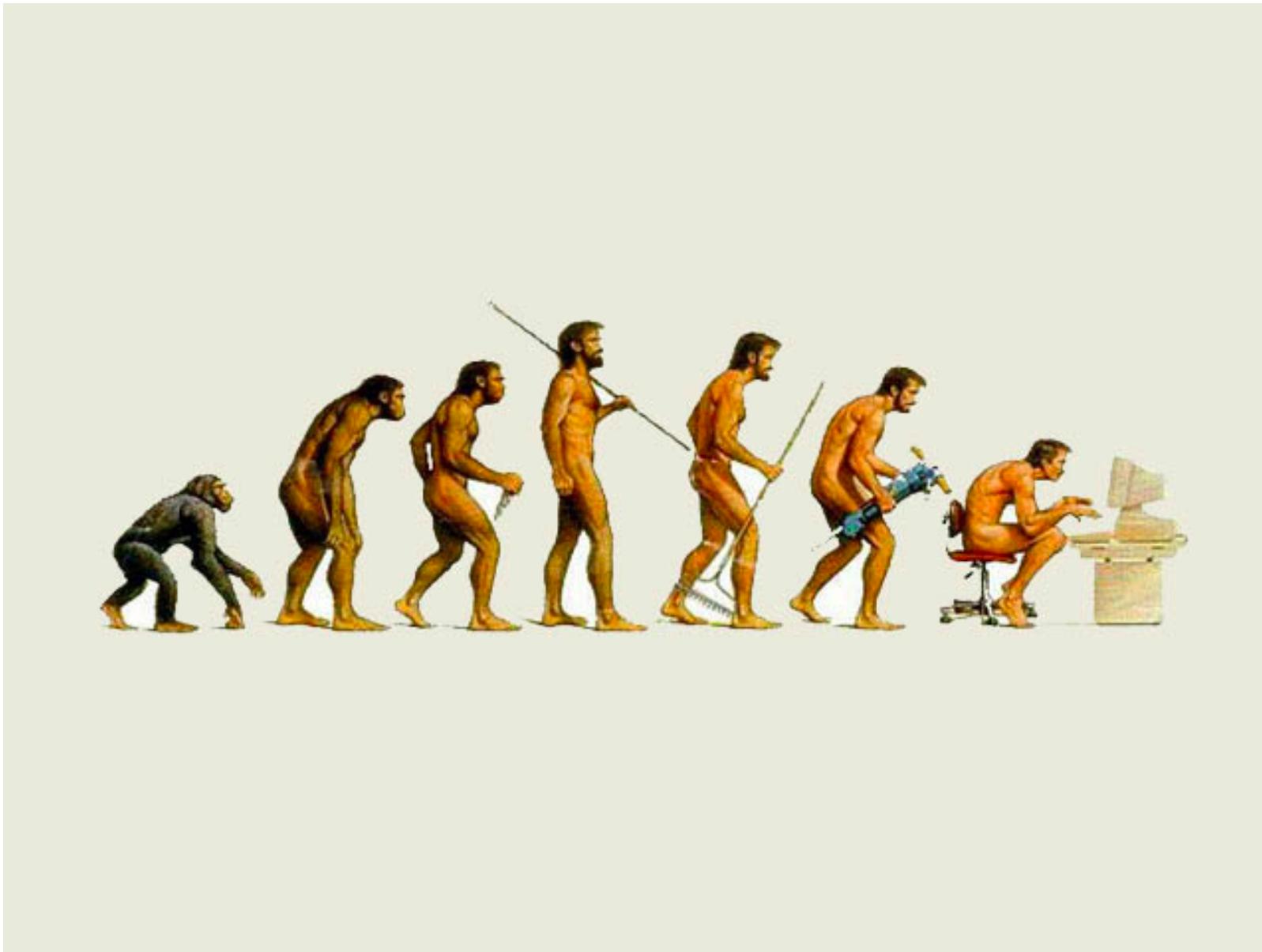


# The Evolution of Distributed Computing and Integration

Oxford University  
Software Engineering Programme  
Dec 2013



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



# Distributed Computing



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# Why is Distributed Computing important?

- Scale
- Evolution
- Integration
- Cross-boundary



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# DCE

- Standardized remote procedure call model proposed in the 90s by the Open Software Foundation (OSF)
- Didn't succeed widely, except as part of Microsoft's DCOM



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

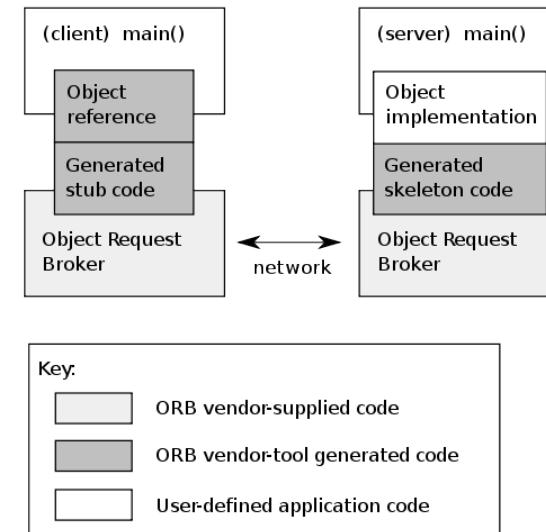
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# CORBA

## Common Object Request Broker Architecture

- An attempt to move DCE like distributed computing into the OO world
- Object Management Group (OMG)
- Still in some use today
- Stagnated/Deprecated



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# SOAP

- XML approach to replace CORBA/DCE, etc
- Theoretically built to be transport independent and message oriented
- 97% of the time used over HTTP in RPC model
- We will cover in much more detail later



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# REST

- Representational State Transfer
  - A model that influenced and was based on HTTP 1.1
- Web approach to distributed computing and integration
- We will cover in much more detail later



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Next generation RPC

- Java RMI (Remote Method Invocation)
- Google Protocol Buffers
- Apache Thrift
- Apache Avro
- Kryo etc
- Twitter's Finagle

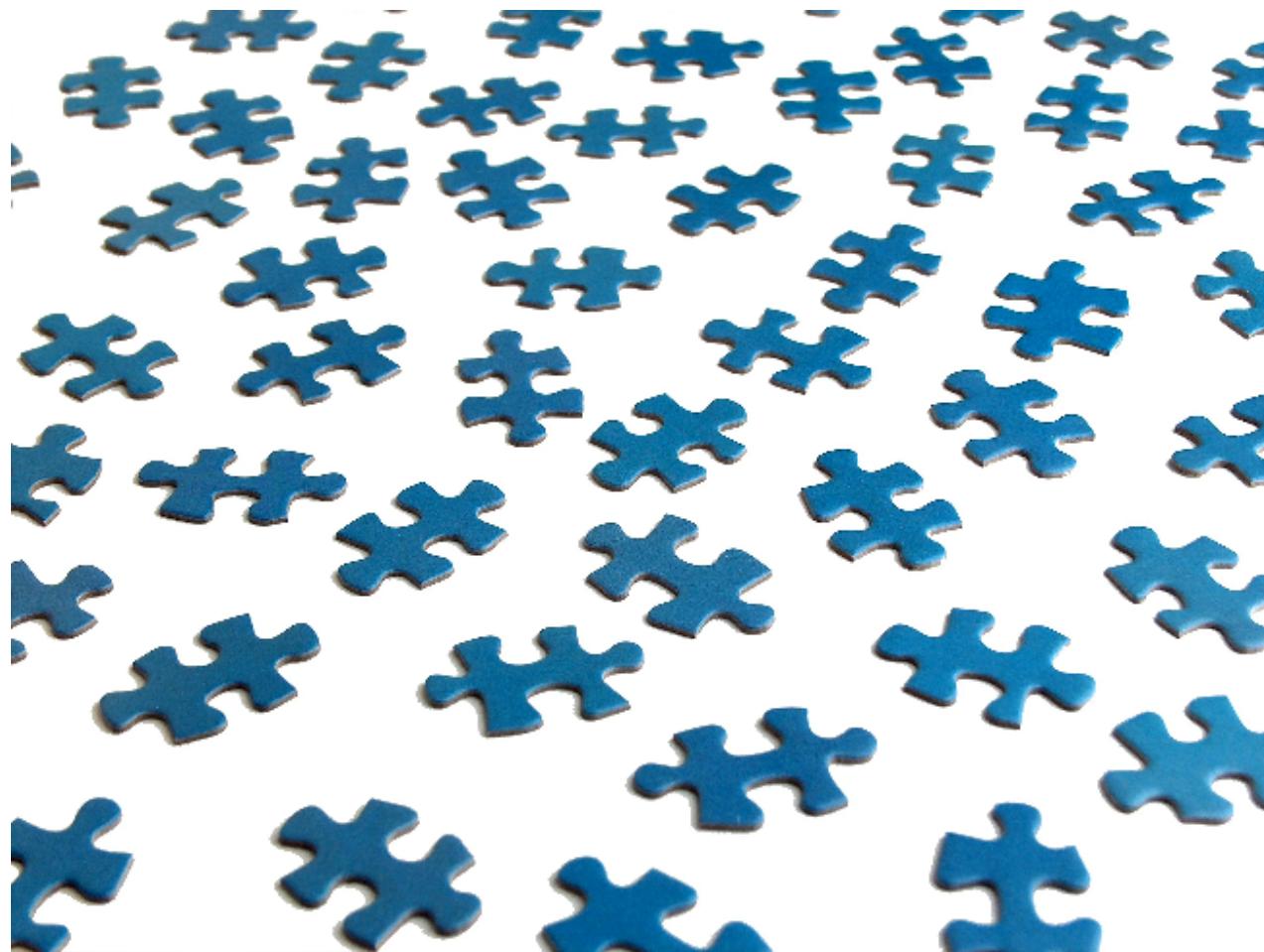


© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Integration



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Integration

- It is unhelpful to consider Distributed Computing and SOA without considering Integration
- Integration is the problem of creating applications that incorporate existing as well as new logic
- Almost no system is standalone and does not need to evolve!
- Even embedded systems are now moving to incorporate Internet and Messaging



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Integration points

- Data
  - Integration in the database layer
- Application
  - Inter-application integration
- Web tier
  - Portal based integration
- On-the-glass
  - Cut and paste



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Tightly coupled

- Tightly coupled systems have significant problems:
  - Errors, delays and downtime spread through the system
  - The resilience of the whole system is based on the weakest part
  - Cost of upgrading or migrating spreads
  - Hard to evaluate the useful parts from the dead weight

© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



# Loose coupling



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s). Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# “6 Degrees of Separation”

- Location
- Access
- Programming Language
- Stack/Vendor
- Time
- Scalability

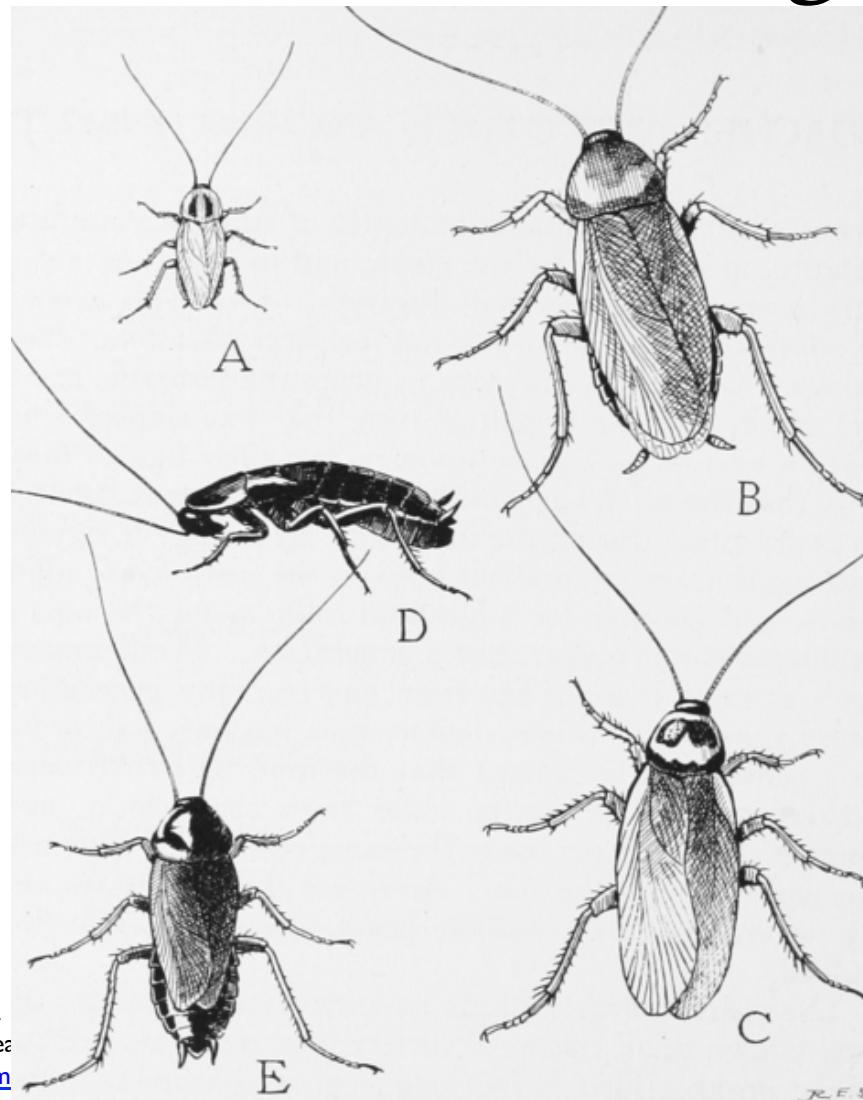


© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

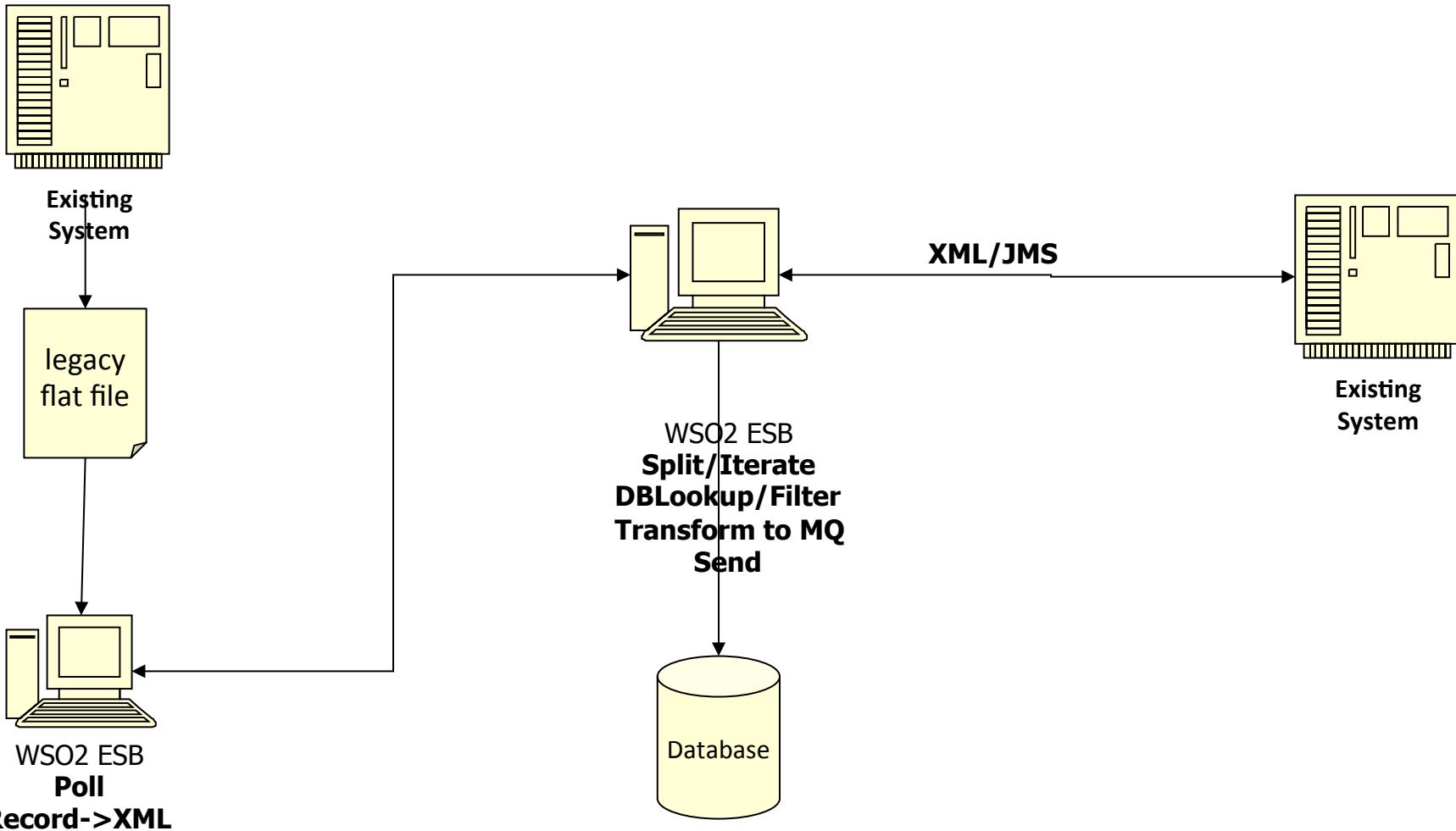
# Batch file transfer - the cockroach of integration



© Paul Fremantle 2012.  
Licensed under the Creative  
Commons Attribution-ShareAlike  
License. See <http://creativecommons.org/licenses/by-sa/3.0/>

the author(s).

# File transfer lives on



Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).



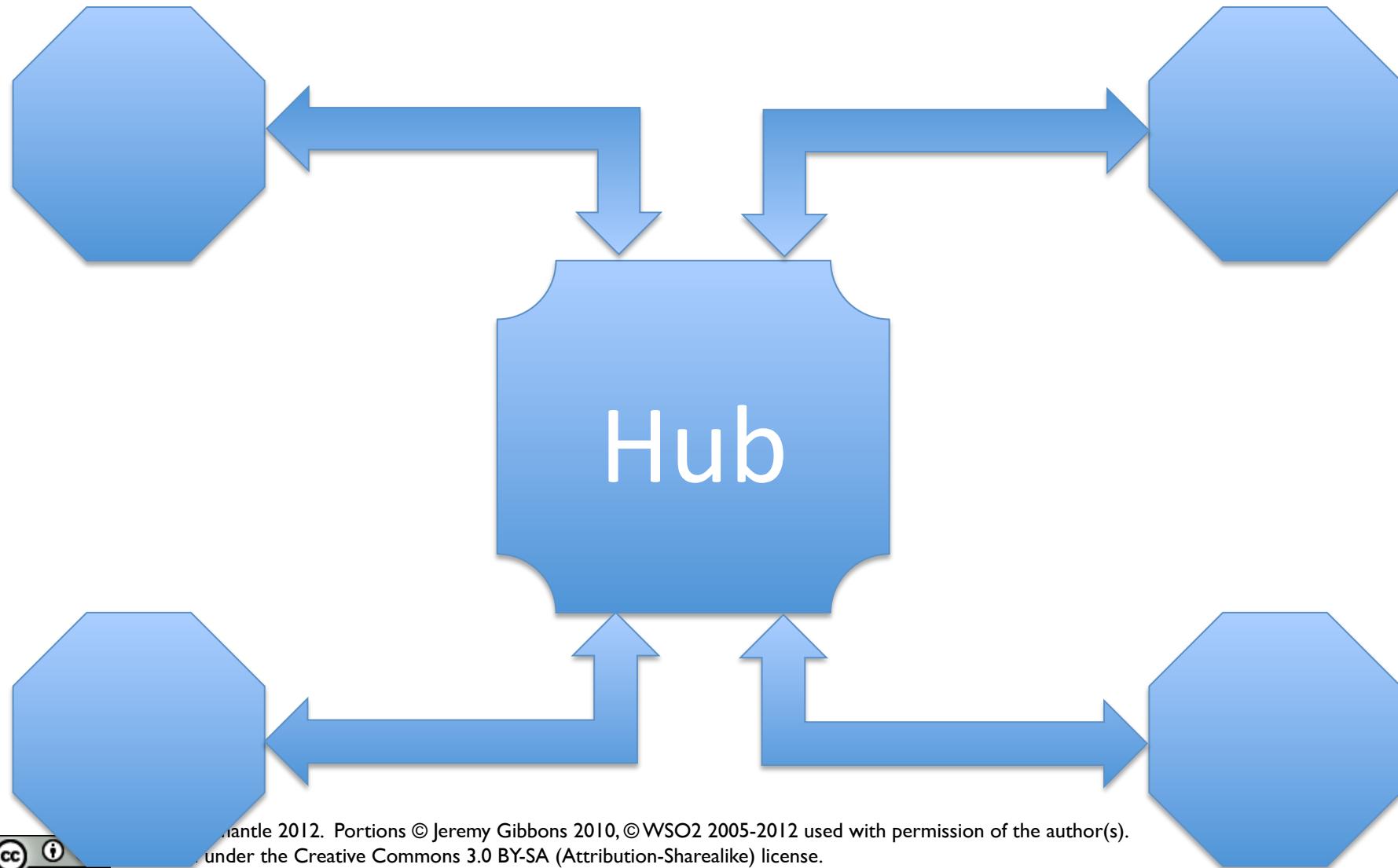
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

NEW YORK

LONDON

# EAI



# EAI hub

- Many integration models still uses the “hub model” today (even with an ESB)
  - Most vendors renamed their hub to ESB
- Why?
  - Well understood pattern
  - Easy to manage
- Why not?
  - Too many meetings with the “EAI Hub Team”
  - Too centrally managed (MQSI experiences)

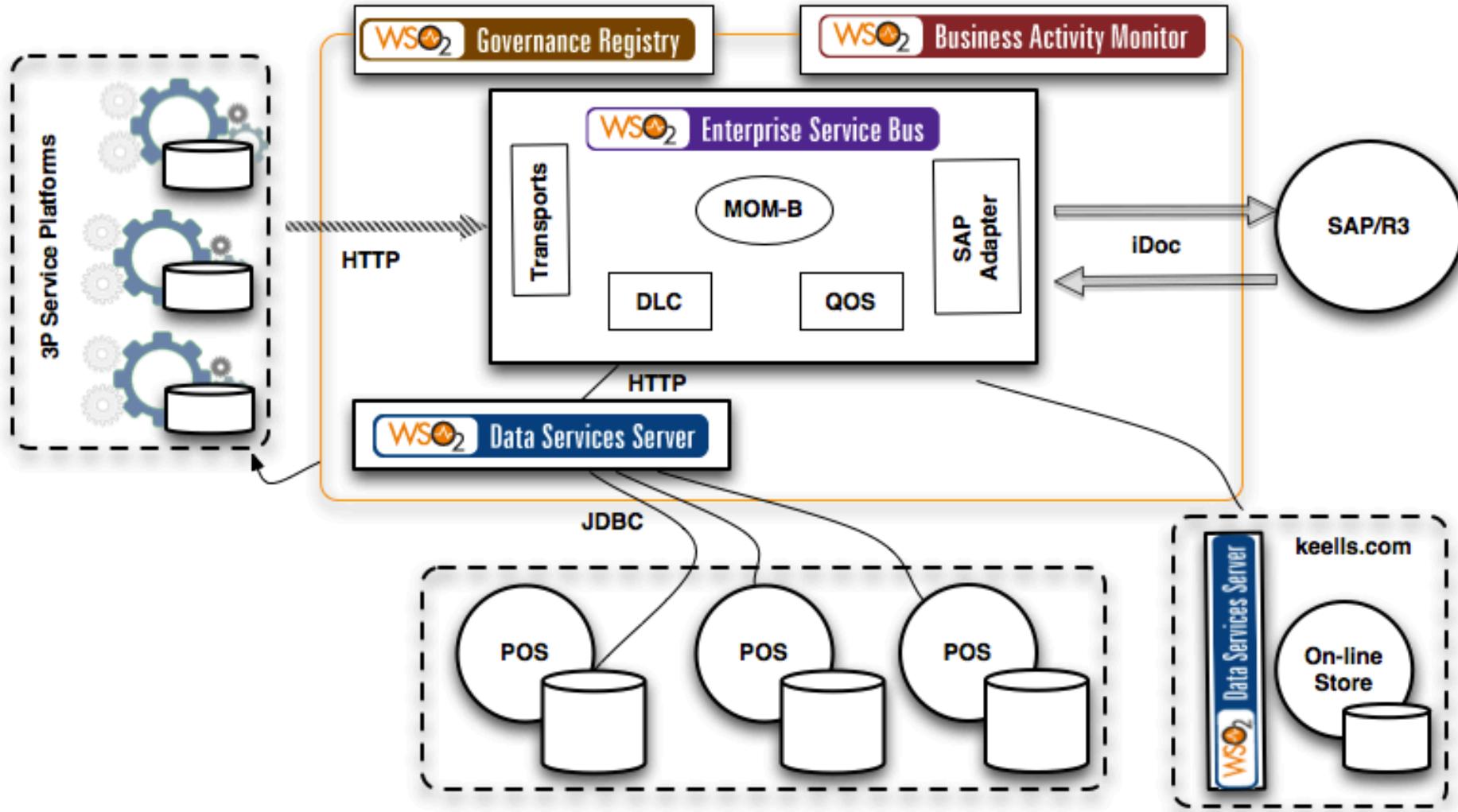


© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

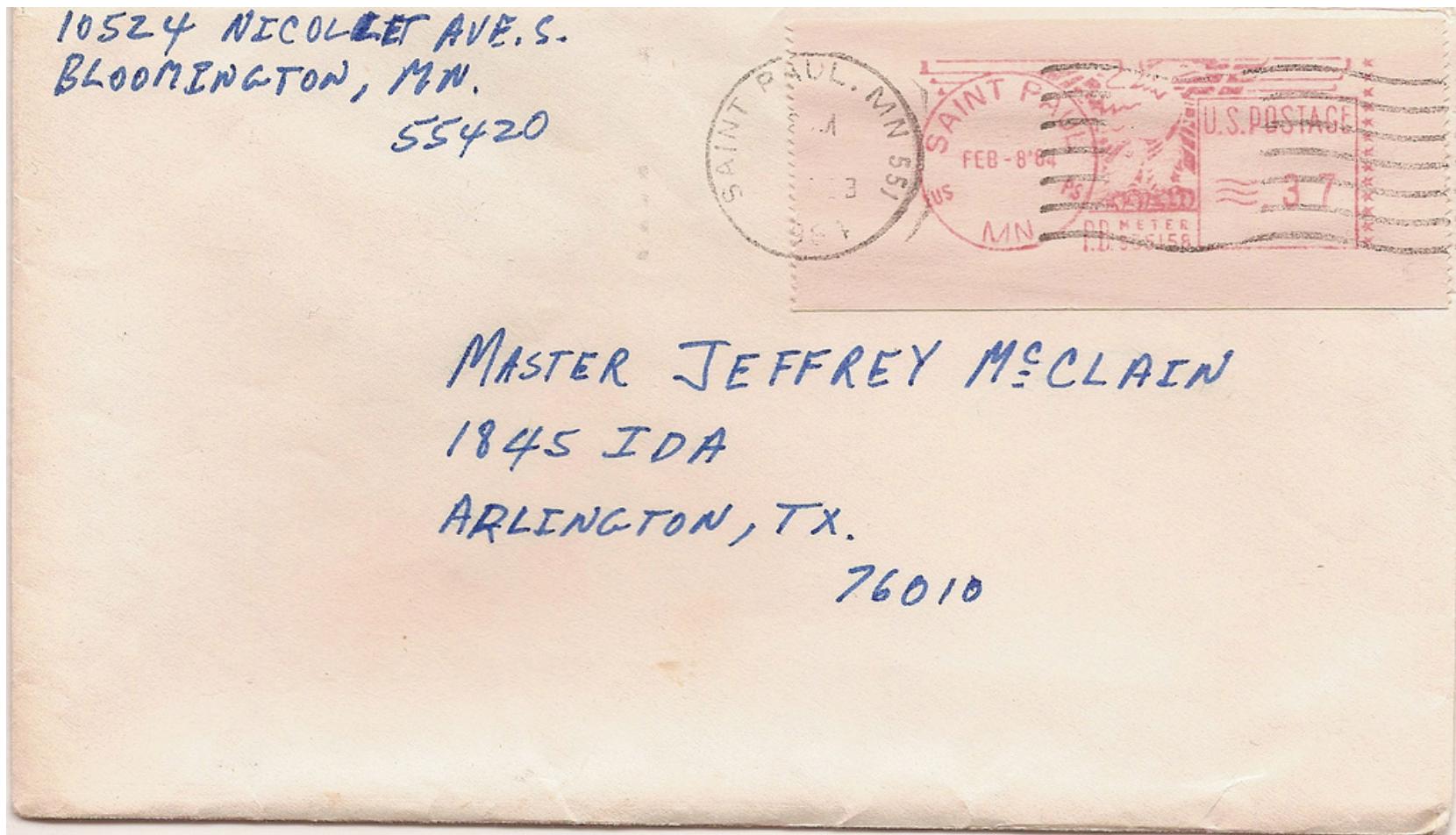
See <http://creativecommons.org/licenses/by-sa/3.0/>

# Hub approach with an ESB



Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# “Message Oriented Middleware”



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSC 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



<http://www.flickr.com/photos/robeast/>

# MOM model

- Decouple message producers from consumers
  - Decoupled in addressing *and* in time
- Not inherently decoupled in message format
  - Though in many cases that too
- One-way asynchronous messages
  - But request-reply possible using “Reply Queues”
- Usually used with reliable delivery

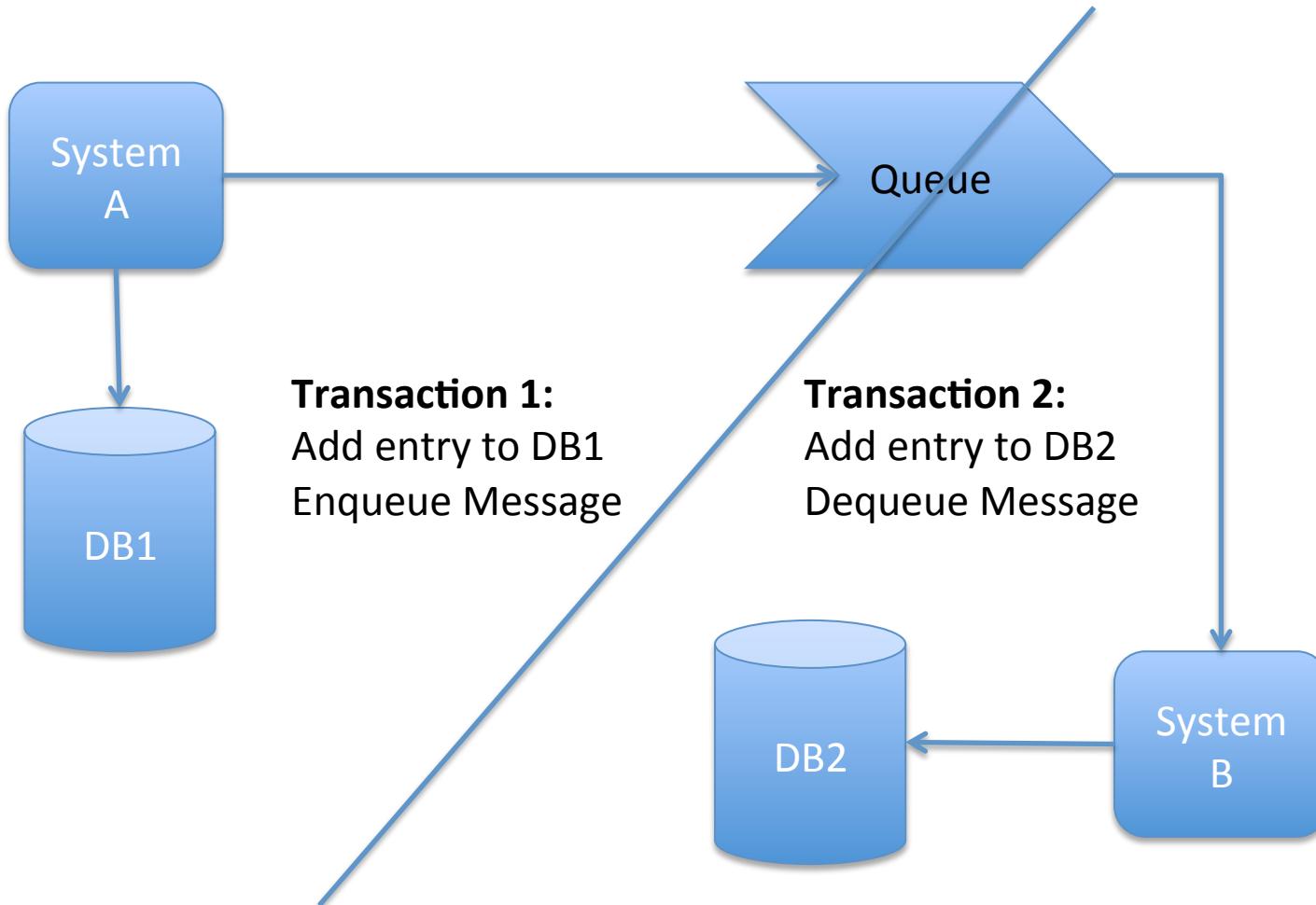


© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Queued Transaction Processing



# AMQP



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# AMQP

- Emerged out of JP Morgan
  - Wanted an open source/open standard alternative to proprietary (e.g. MQSeries)
  - Standardized at OASIS
  - Multiple Open Source implementations
    - RabbitMQ, Apache Qpid, WSO2 Message Broker, Apache Apollo, Apache ActiveMQ, etc
  - Used widely e.g. in OpenStack



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# MQTT

- Very lightweight messaging pub/sub system
- Built for SCADA, M2M, IoT
- Being standardised at OASIS MQTT TC
  - Close to being done
  - No wire changes from 3.1
- Big effort in the Eclipse Paho / M2M project

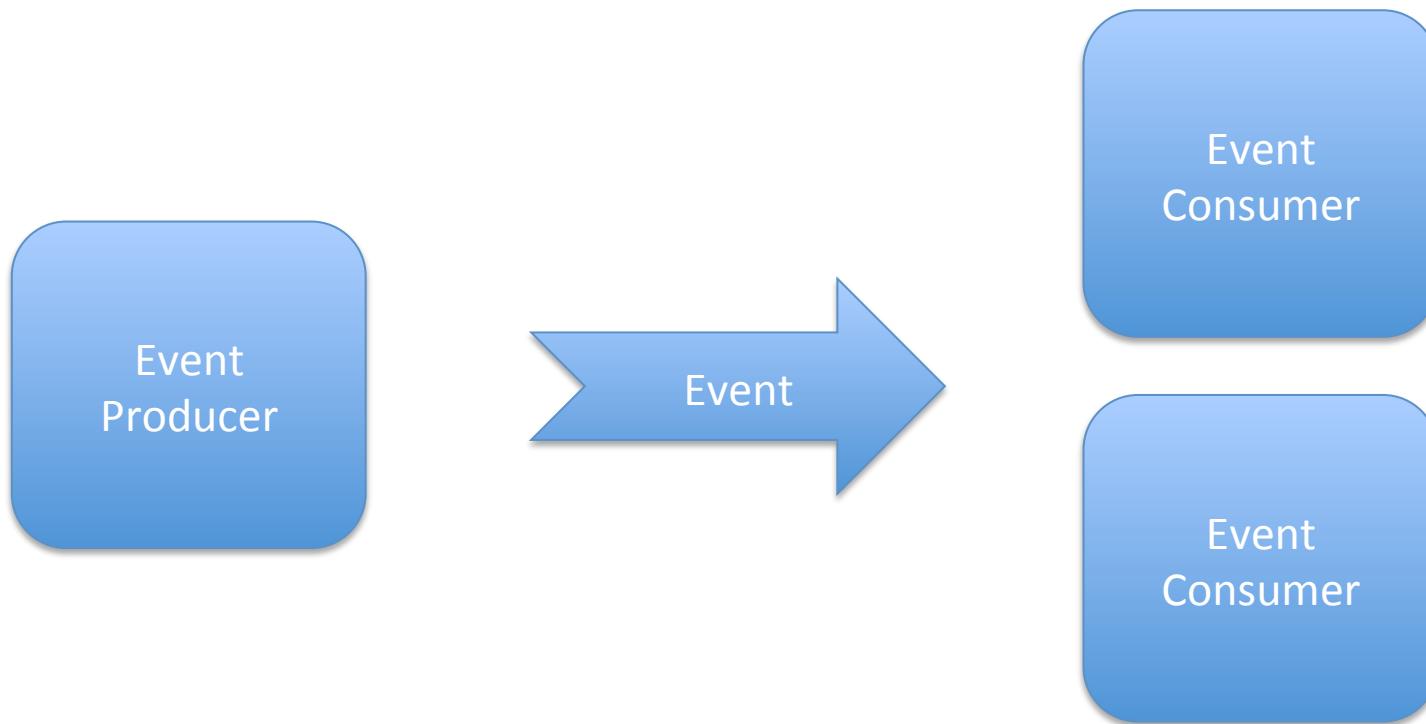


© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Event Driven Architecture



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# “Event Driven Architecture”

- Actually how Apache (and WSO2) work(s)
  - Mailing lists = topics
- Can be layered with reliable delivery
- Used a lot in high-volume logging, trading environments, fraud detection, etc
- *Requires a very different mindset*



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# EDA

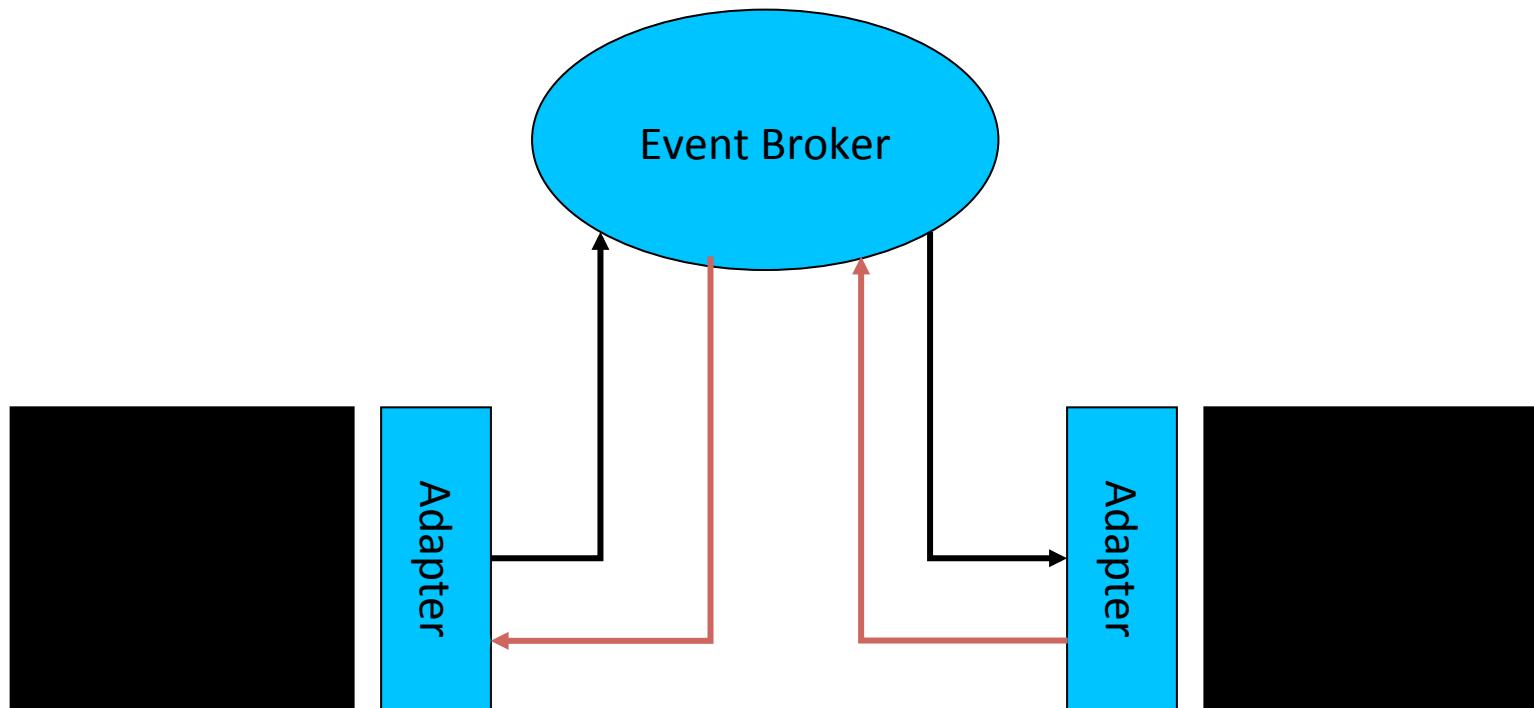


© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Feedback loops

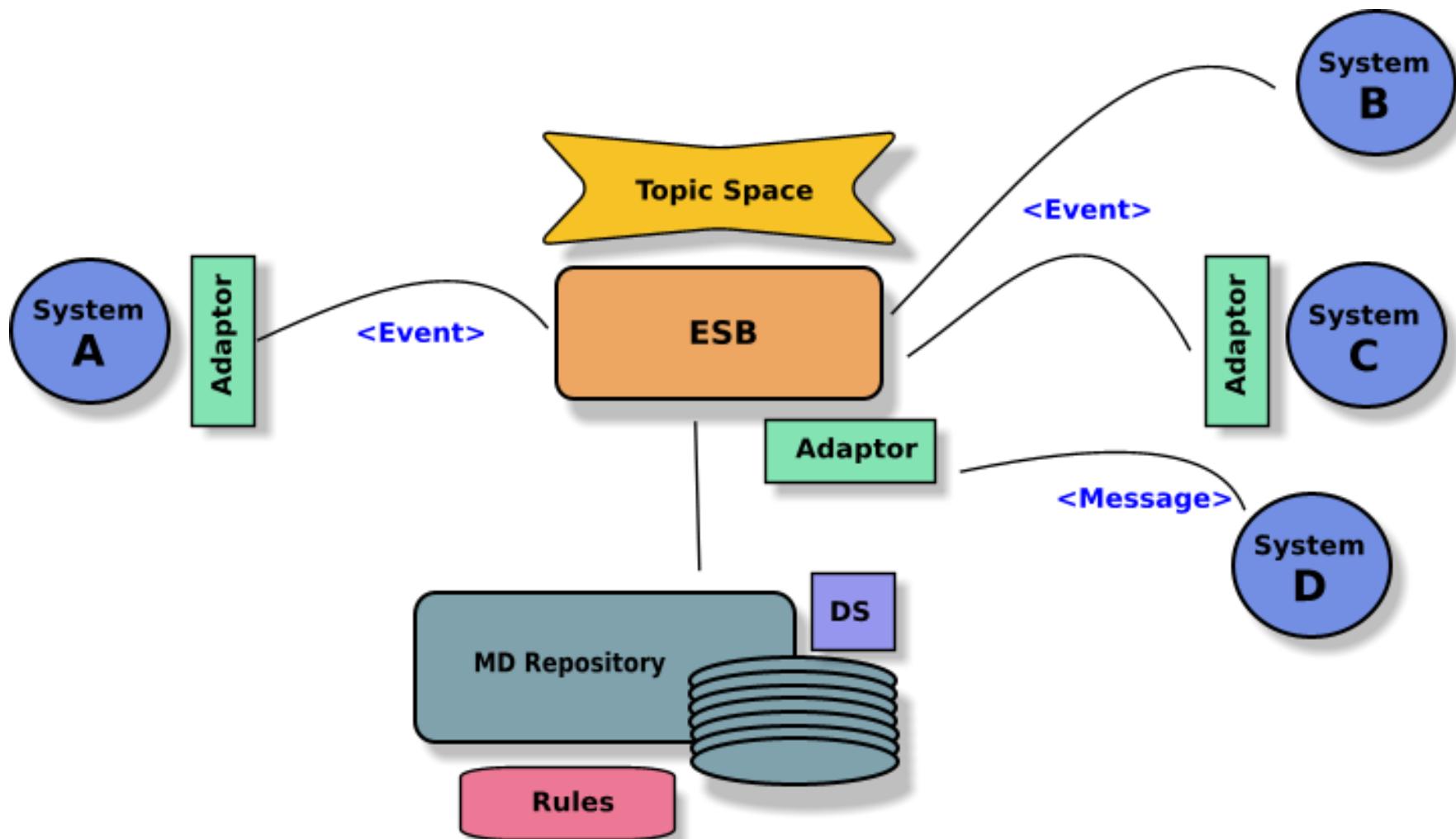


<http://pzf.fremantle.org/2008/09/interesting-problem-in-event-driven.html>



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# Solution



Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# Why did SOA evolve?

- Directly came out of XML
  - Understanding the schema and structure of messages
  - Especially within the “fabric” not just at the endpoints
- What’s different?
  - Metadata
  - Policies
  - Security



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Service



<http://www.flickr.com/photos/yjv/>

© Paul Fernandes 2012. Portions © Jeremy Gibbons 2010, © WSQZ 2009-2012 used with permission of the author(s).



Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# SOA failures

One consumer per service



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# SOA failures

“Just buy an ESB from me”



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# SOA failures

## Vendor Driven Architecture

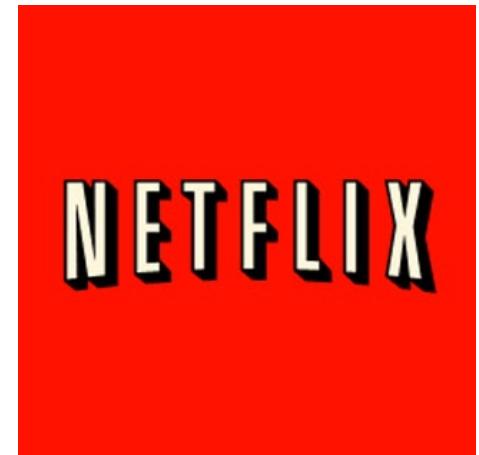
© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



# “soa” successes



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

eBay Open Source

https://www.ebayopensource.org/index.php/Turmeric/HomePage

WSO2, Inc. Mail - Int Offline WSO2 Mail WSO2, Inc. - Calend... Fremantle Family Ma Shorten with bit.ly Gmail - Inbox (720) HP Photosmart C450

{} eBay Open Source

Search

Home Projects Downloads Source Issues Documentation Forums Builds Login/Register

Turmeric

Print

Fork me on GitHub

[Home](#)

[Get Involved](#)

[Downloads](#)

[Prior Releases](#)

[License](#)

[Nightly Builds](#)

[FAQ](#)

Documentation

[1.1.0](#)

[Overview](#)

[Architecture](#)

[Installation](#)

[Tutorials](#)

 Turmeric

## What is Turmeric?

Turmeric is a comprehensive, policy-driven SOA platform that you can use to develop, deploy, secure, run and monitor SOA services and consumers. It is a Java based platform, follows the standards (SOAP, XML, JSON, XACML, etc.), and supports WSDL (SOAP style - Doc Lit wrapped mode and REST style). It supports a variety of protocols and data formats. Eclipse plugins help with the development of services and consumers. Other important features include:

- Various Quality of Service (QoS) features such as authentication, authorization, and rate limiting, which you control by defining respective policies.
- Monitoring capabilities.



# Why ESB/SOA model isn't just EAI

- Policy based
  - XACML, Throttling Policy, etc
  - eBay's Internal Service Router
- Independent management
  - Loose coupling of configuration
  - Hot deploy / re-deploy / continuous delivery
- Governance
  - Lifecycle and Dependency management
  - Analysis and reporting on the meta-model
- Non-blocking asynchronous routing
- Distributed architecture



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

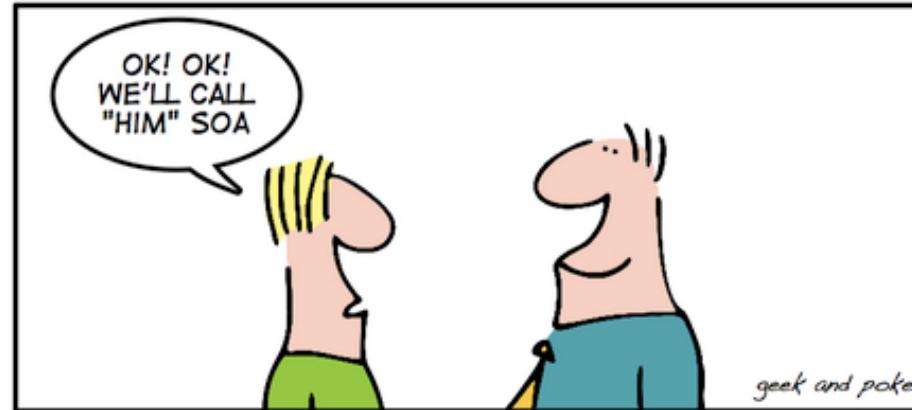
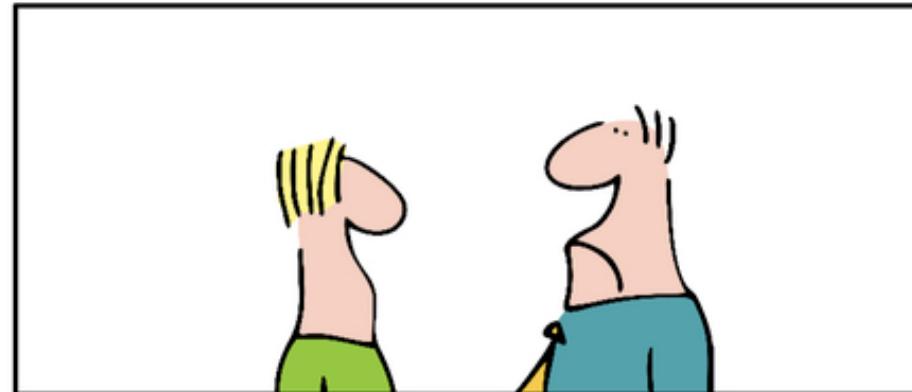
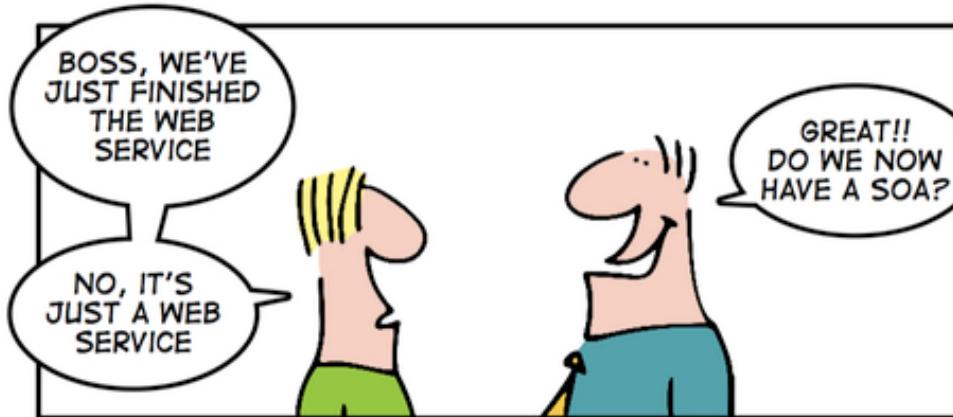
See <http://creativecommons.org/licenses/by-sa/3.0/>

# Web Services != SOA

- There are SOA's that don't use Web Services
  - We will see some in the case studies...
- There are systems that use Web Services that are not Service Oriented
  - e.g. Taking an existing text/TCP model and tunnelling over SOAP



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

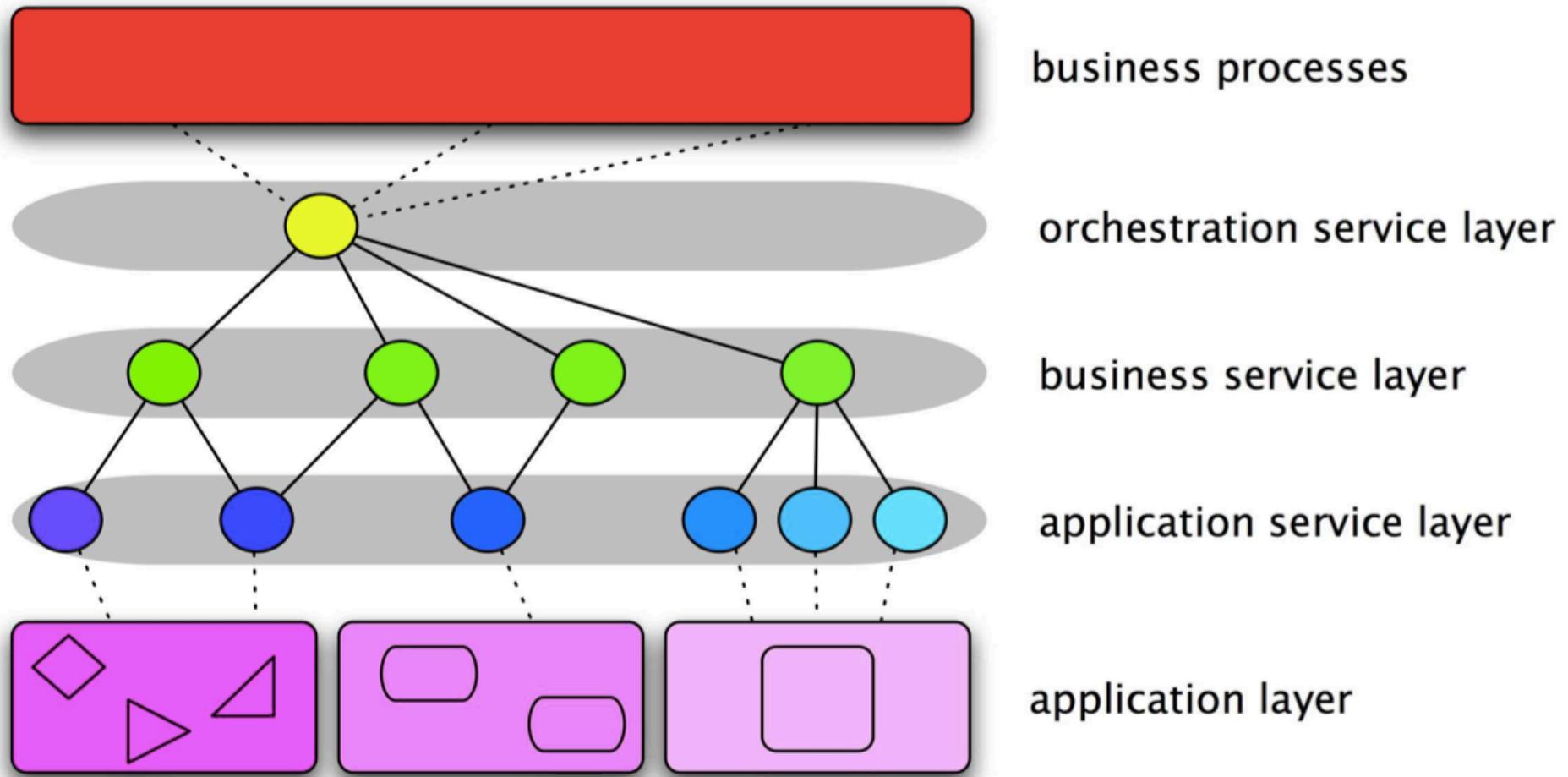


© Paul Fremantle 2008  
Licensed under the  
See <http://creativecommons.org/licenses/by-sa/2.0/>

## HOW TO GET A SOA

or(s).

# SOA



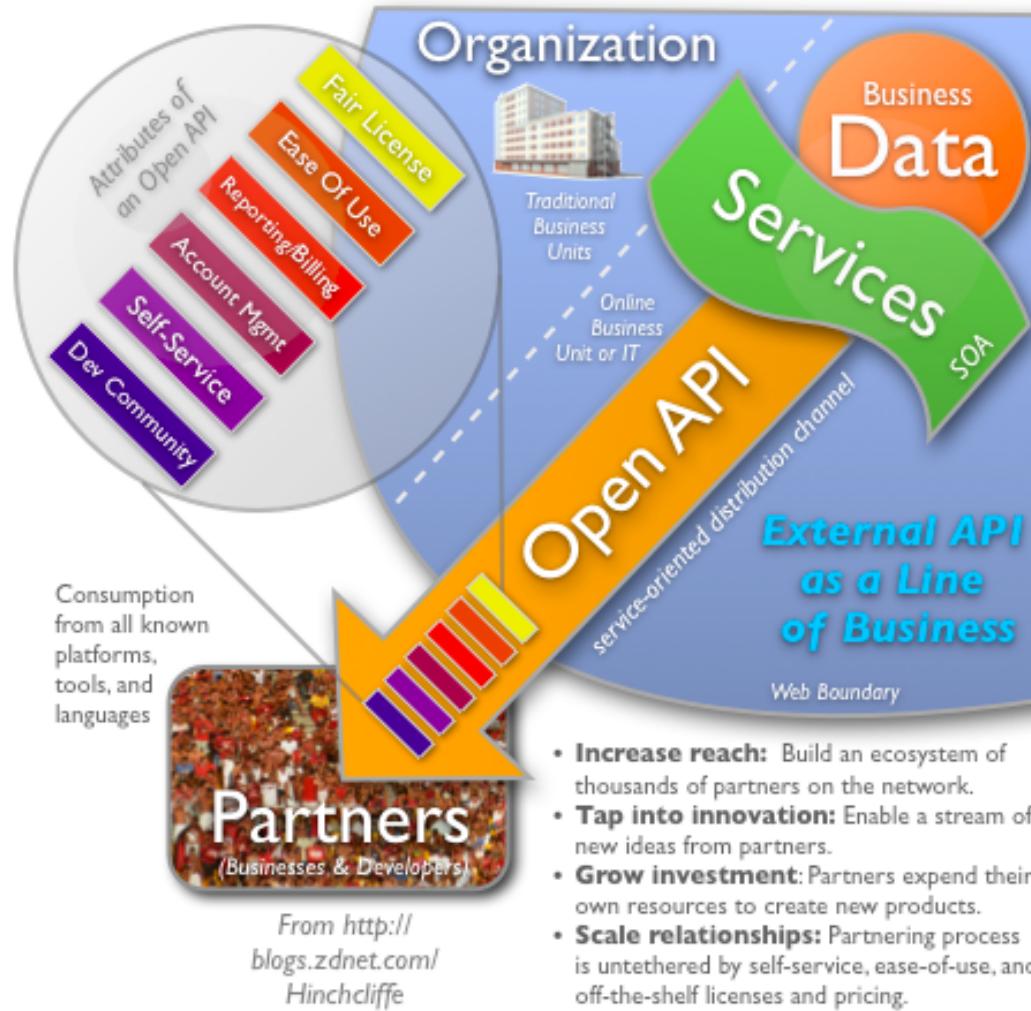
© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



# Running your SOA like a Web startup



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



<http://www.zdnet.com/blog/hinchcliffe/running-your-soa-like-a-web-startup/525>

# API

- **An API** is a business capability delivered over the Internet to internal or external consumers
  - Network accessible function
  - Available using standard web protocols
  - With well-defined interfaces
  - Designed for access by third-parties



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# What is different from an API and a Service?

- Publishing your API in a Portal
- Expecting people to use it without them having to meet with you
- Making it easy to consume (JSON? Ready built clients in Github?)
- Governance
  - Caring who uses it
  - Letting them know when you version it
  - Meeting an SLA



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Key API technologies

- json / rest
- OAuth / OAuth2 keys
- SLA management
- API portal / API Store
  - Catalogue, subscription/purchase
  - Monetization
  - Forum, Ratings, Social
- Analytics



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

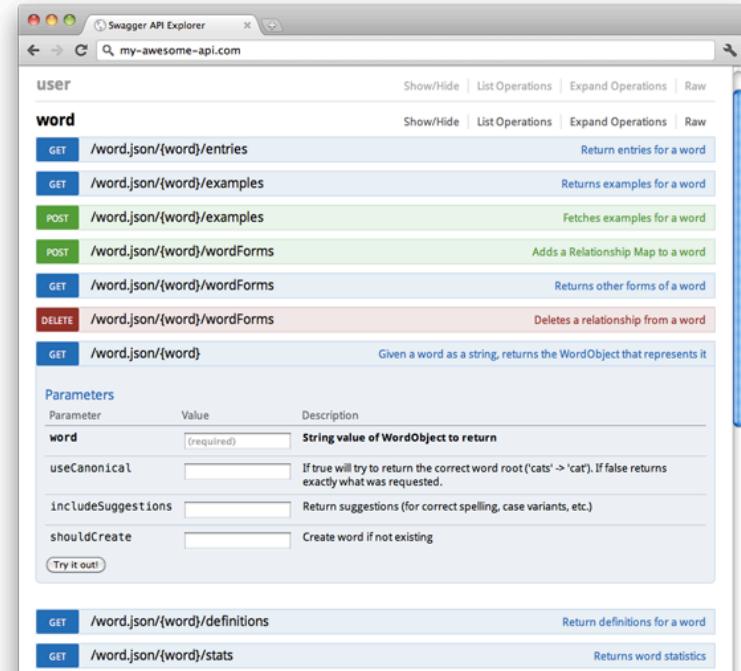
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

# REST description



The screenshot shows the Swagger API Explorer interface. On the left, there's a detailed view of a 'USER' resource, specifically for the 'word' endpoint. It lists several operations:

- GET /word.json/{word}/entries**: Returns entries for a word.
- GET /word.json/{word}/examples**: Returns examples for a word.
- POST /word.json/{word}/examples**: Fetches examples for a word.
- POST /word.json/{word}/wordForms**: Adds a Relationship Map to a word.
- GET /word.json/{word}/wordForms**: Returns other forms of a word.
- DELETE /word.json/{word}/wordForms**: Deletes a relationship from a word.
- GET /word.json/{word}**: Given a word as a string, returns the WordObject that represents it.

Below the operations, there's a table for parameters:

Parameter	Value	Description
word	(required)	String value of WordObject to return
useCanonical		If true will try to return the correct word root ('cats' -> 'cat'). If false returns exactly what was requested.
includeSuggestions		Return suggestions (for correct spelling, case variants, etc.)
shouldCreate		Create word if not existing

At the bottom of the main panel, there are two more operations:

- GET /word.json/{word}/definitions**: Returns definitions for a word.
- GET /word.json/{word}/stats**: Returns word statistics.

**Document your API with Style**

Swagger is a specification and complete framework implementation for describing, producing, consuming, and visualizing RESTful web services. The overarching goal of Swagger is to enable client and documentation systems to update at the same pace as the server. The documentation of methods, parameters and models are tightly integrated into the server code, allowing APIs to always stay in sync. With Swagger, deploying, managing, and using powerful APIs has never been easier.

**Who is responsible for Swagger?**

Both the specification and framework implementation are initiatives from Wordnik. Swagger was developed for Wordnik's own use during the development of [developer.wordnik.com](#) and the underlying system. Swagger development began in early 2010—the framework being released is currently used by Wordnik's APIs, which power both internal and external API clients.

**Why is Swagger useful?**

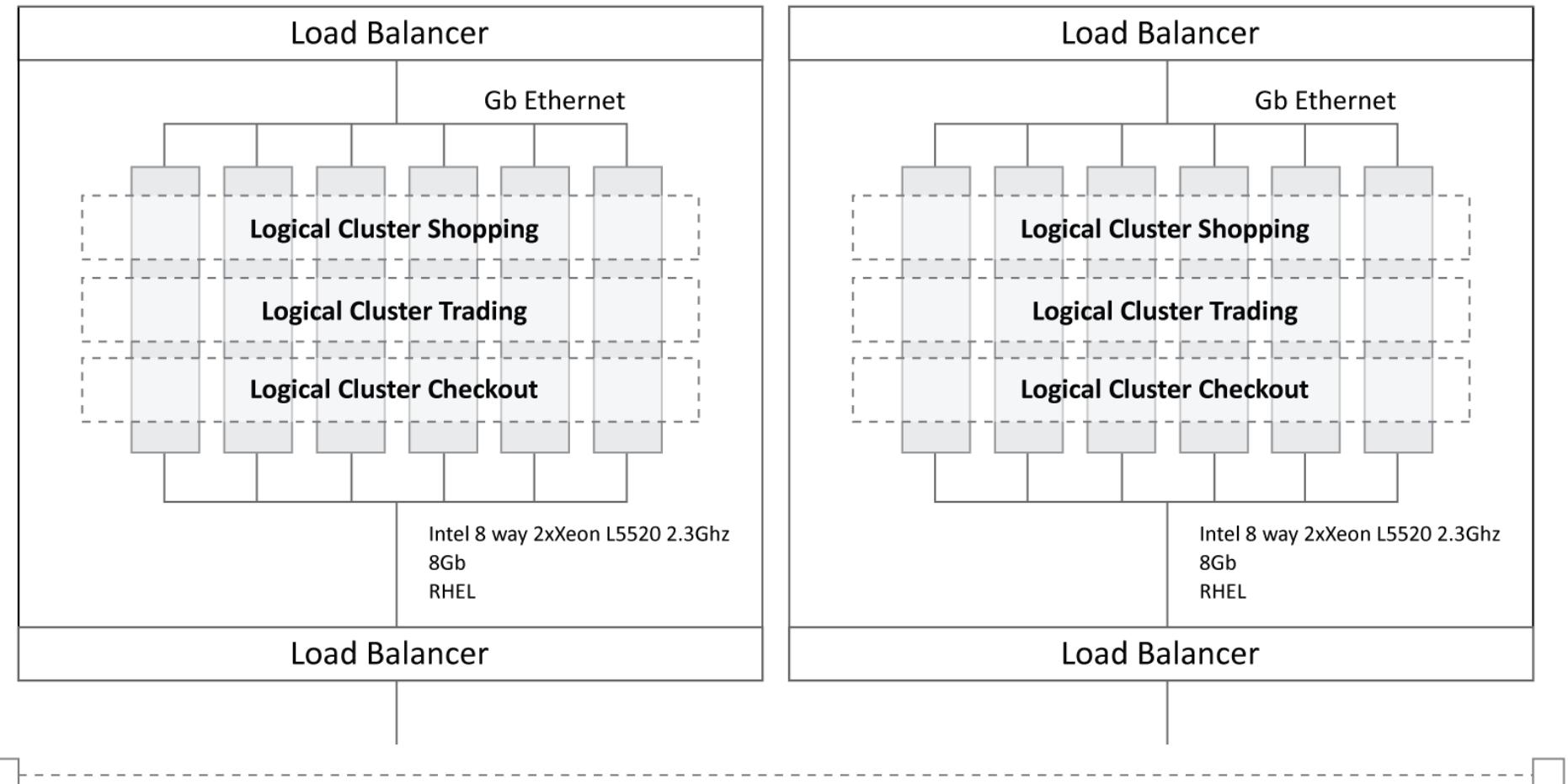
© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



# High volume integration @ eBay



100's of service instances

Tomcat + Axis2

# Change in focus

- Security, tokens, access control/entitlement
- Throttling, caching
- Latency and CPU usage
- Monitoring, BAM and CEP



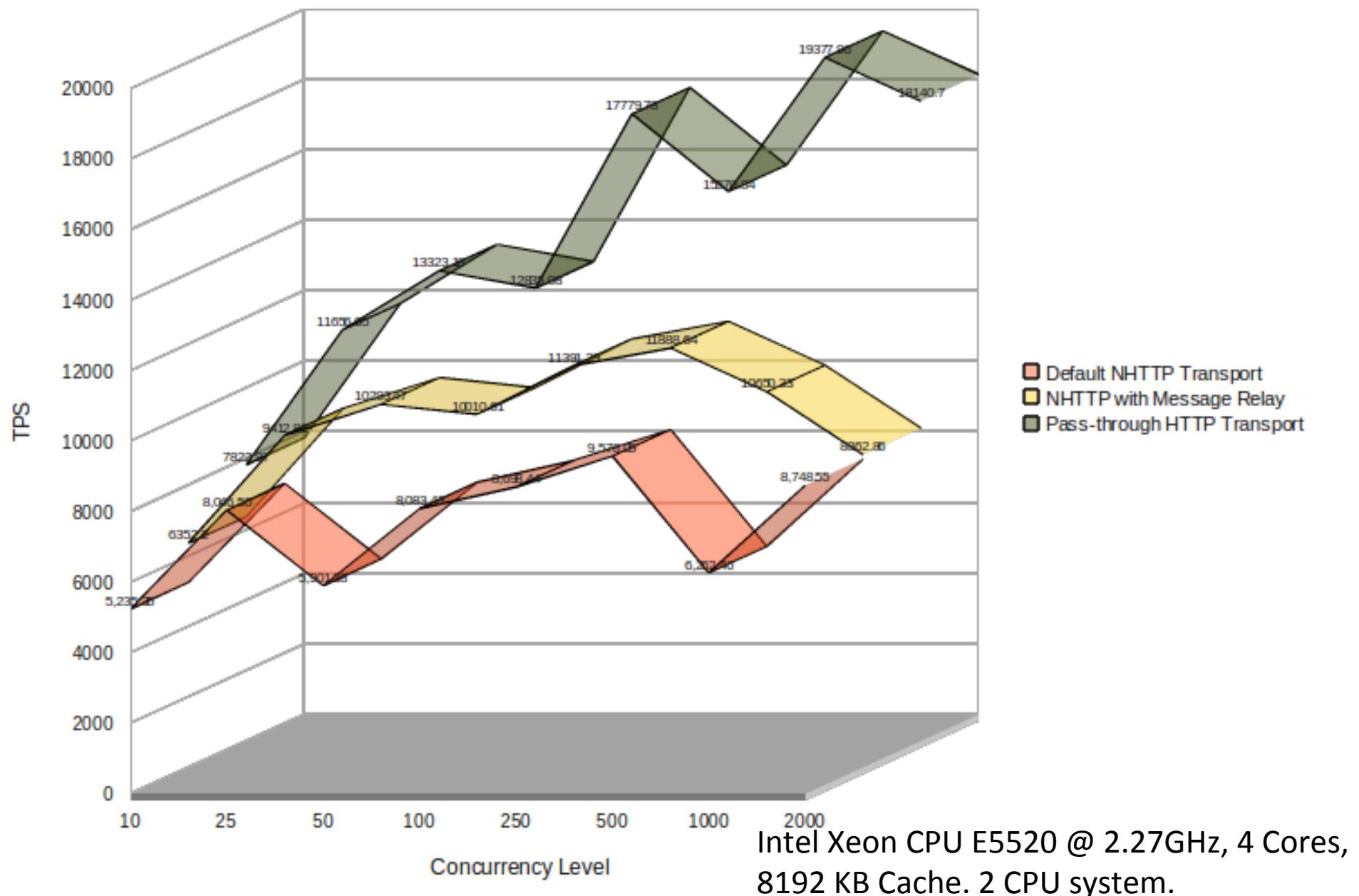
© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

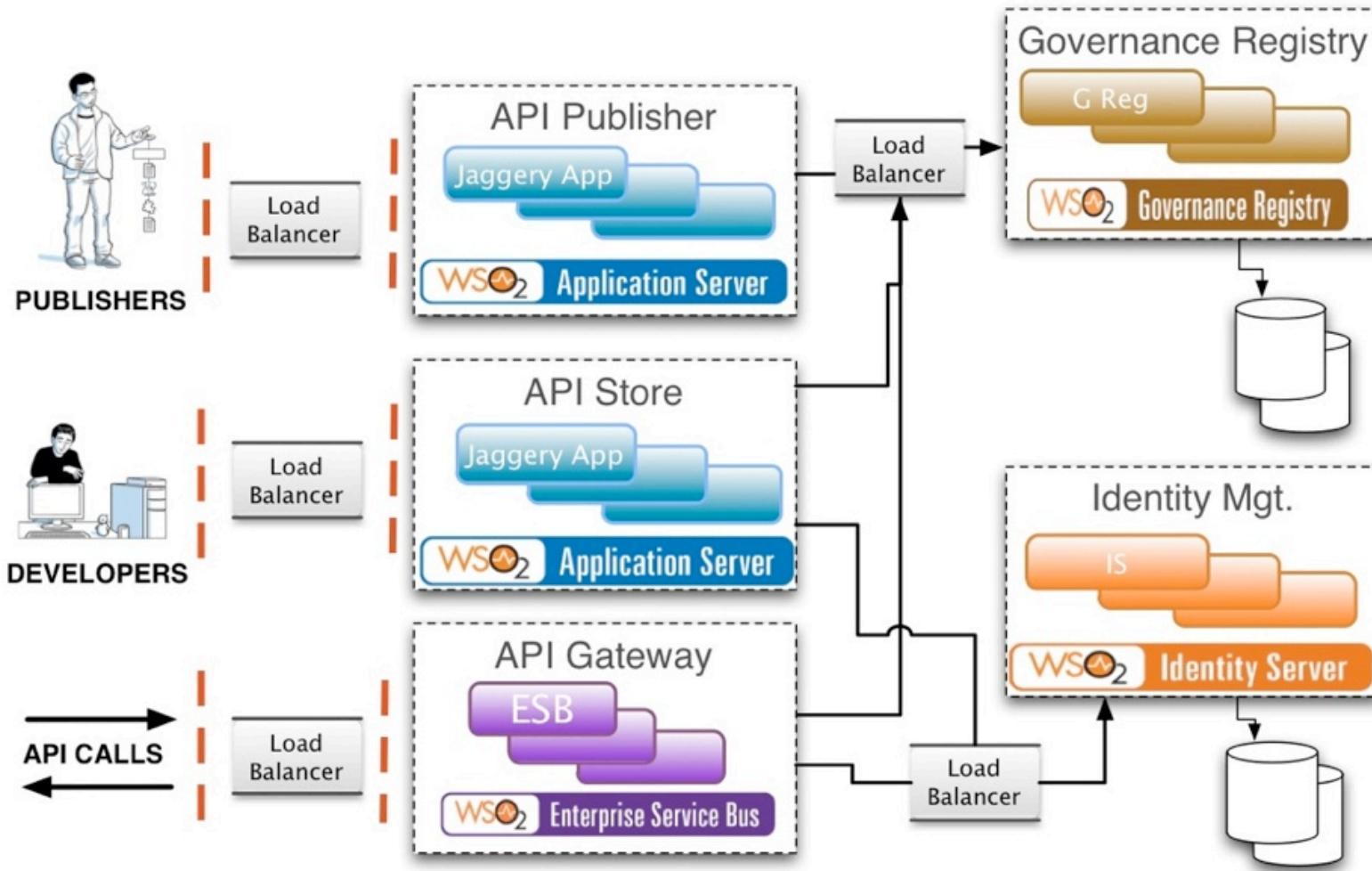
See <http://creativecommons.org/licenses/by-sa/3.0/>

## Pass-through Proxying

Message Size 0.5K



# API management



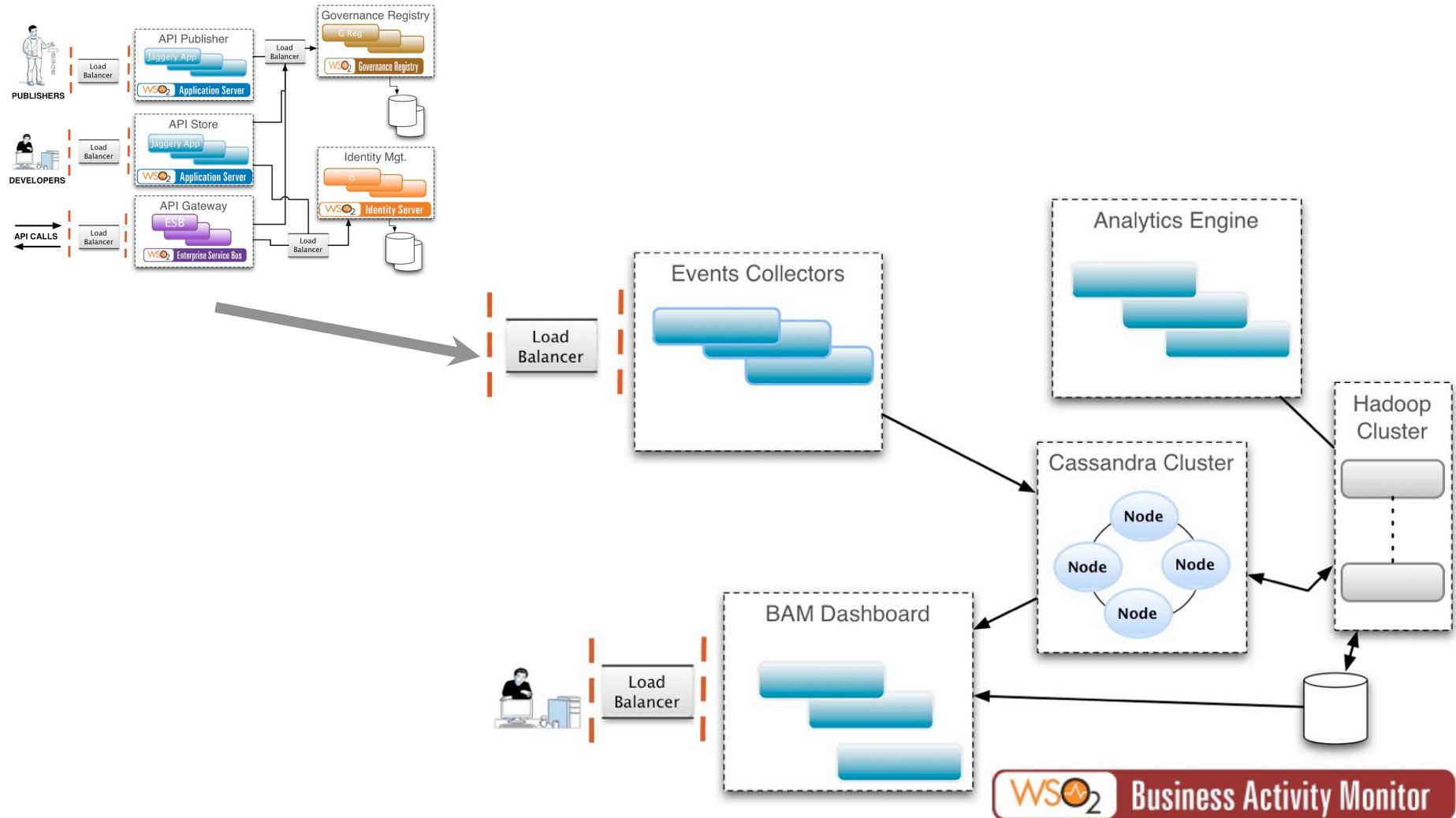
© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



# Scalable analytics



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2009-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



# Cloud integration



salesforce.com



Google™ Docs

ESB-as-a-Service



# Cloud integration

- APIs are the right approach
  - Use a “cloud gateway” to bridge into internal systems
- “Push-me pull-you” pattern
  - Use an active ESB in the cloud
- Analytics
  - See what is happening



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# What's next?

- Still a long way from canonical models
- Successful systems are using “soa” and “rest” at scale
  - Architecture is more important than dogma
- Governance sounds boring but is key
- Applying monetization approaches and “API Economy” models
- Analytics and feedback loops



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>

# Summary

- Integration has evolved in some interesting ways
  - Async messaging, EDA, APIs, High Volume
- Evolution isn't monotonic
- Doing APIs right is about the mindset as much as the technology



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).

Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.

See <http://creativecommons.org/licenses/by-sa/3.0/>



© Paul Fremantle 2012. Portions © Jeremy Gibbons 2010, © WSO2 2005-2012 used with permission of the author(s).  
Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>