



[www.centricconsulting.com](http://www.centricconsulting.com)

# Event Driven Architecture

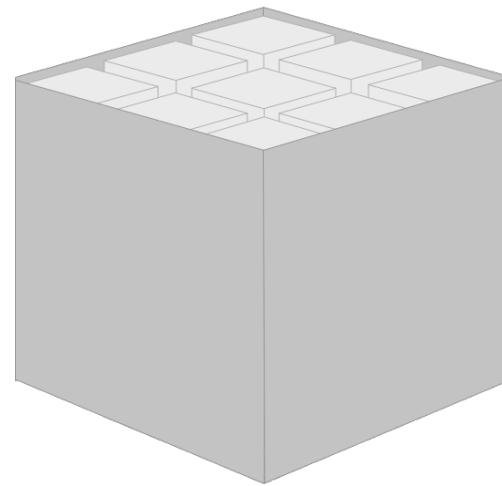
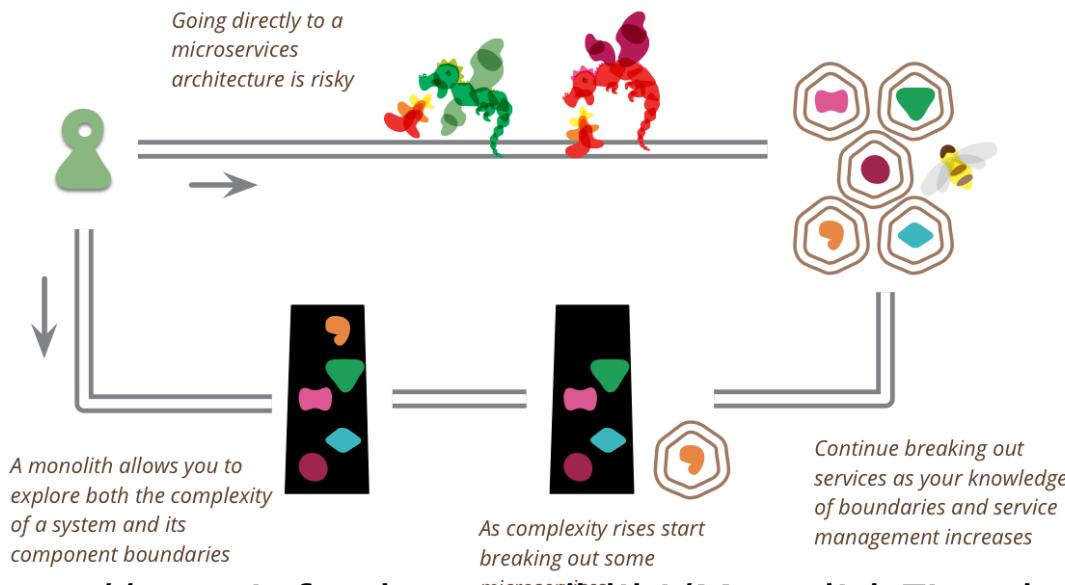
A primer...

@ShawnWallace

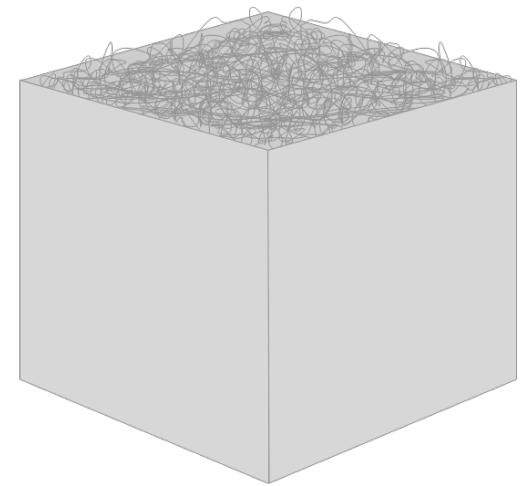
# The Great Debate...

<http://martinfowler.com/articles/dont-start-monolith.html>

## Monolith First?



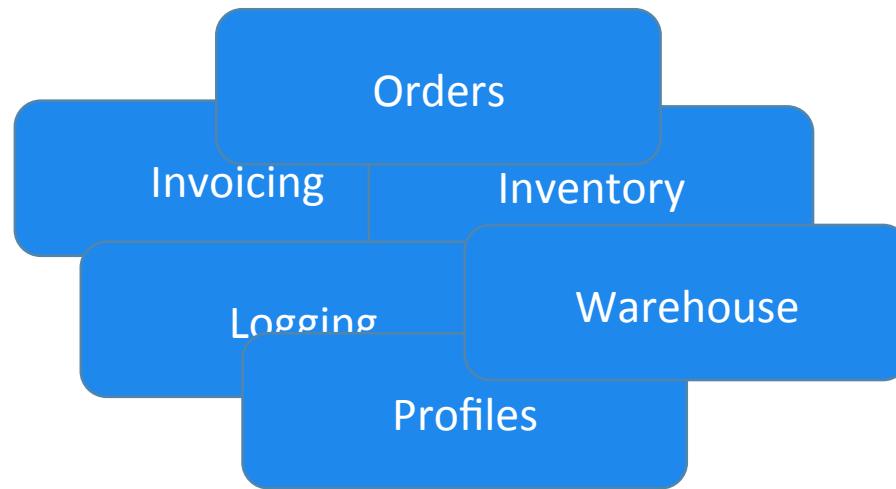
Hope



Reality

## Micro services First?

# Monolithic



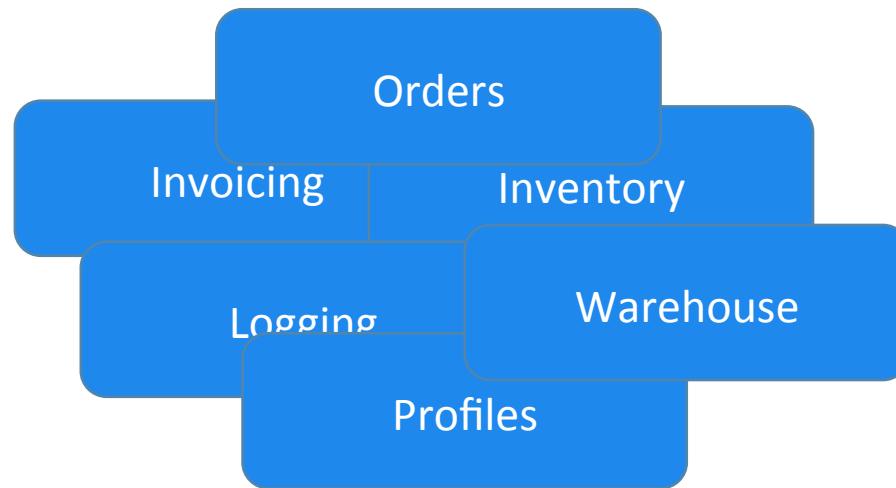
# A problem with ‘Monolith First’

A product of our agile approach to our projects...smaller features finished earlier.

The goal is to deploy more often...this is hard.

One solution is to have smaller applications.

# Monolithic



# SOA

Orders

Invoicing

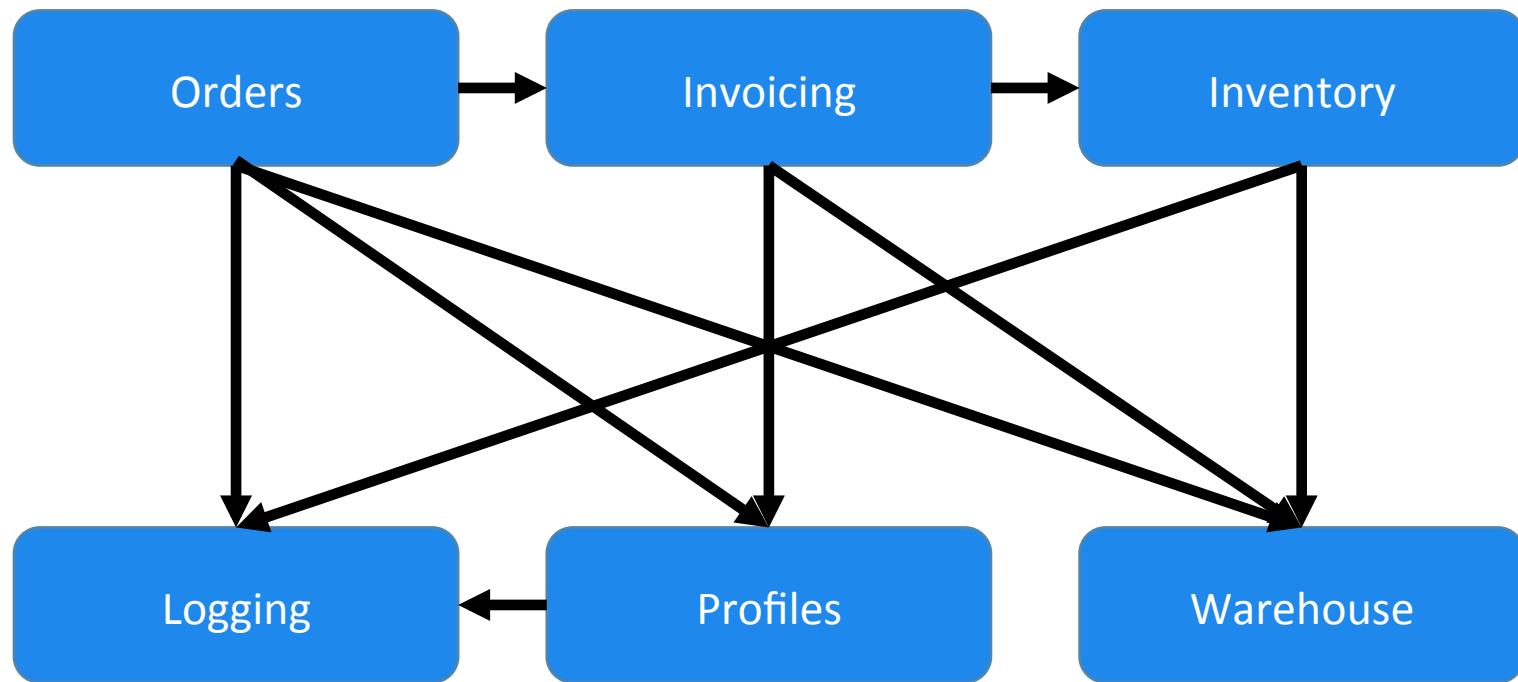
Inventory

Logging

Profiles

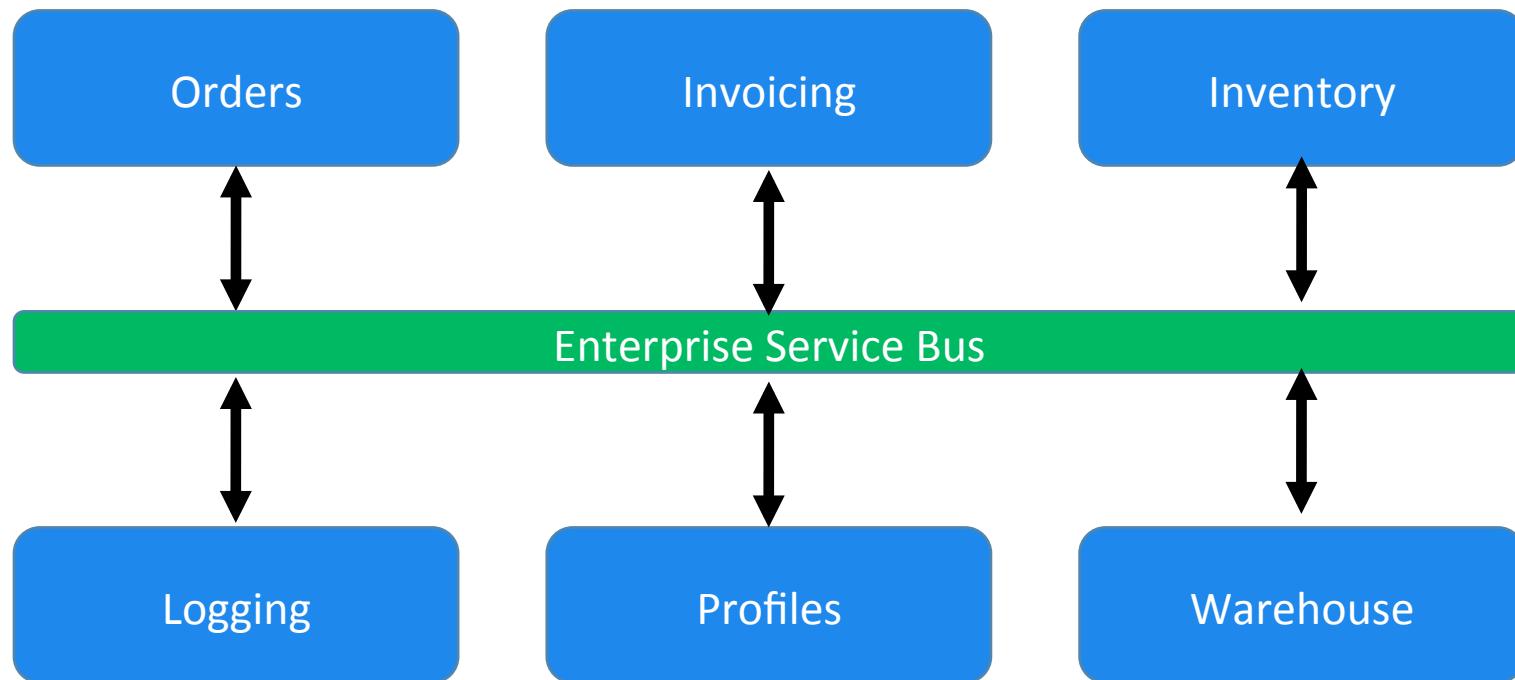
Warehouse

# SOA



Tightly Coupled

# SOA with Enterprise Service Bus



STILL Tightly Coupled

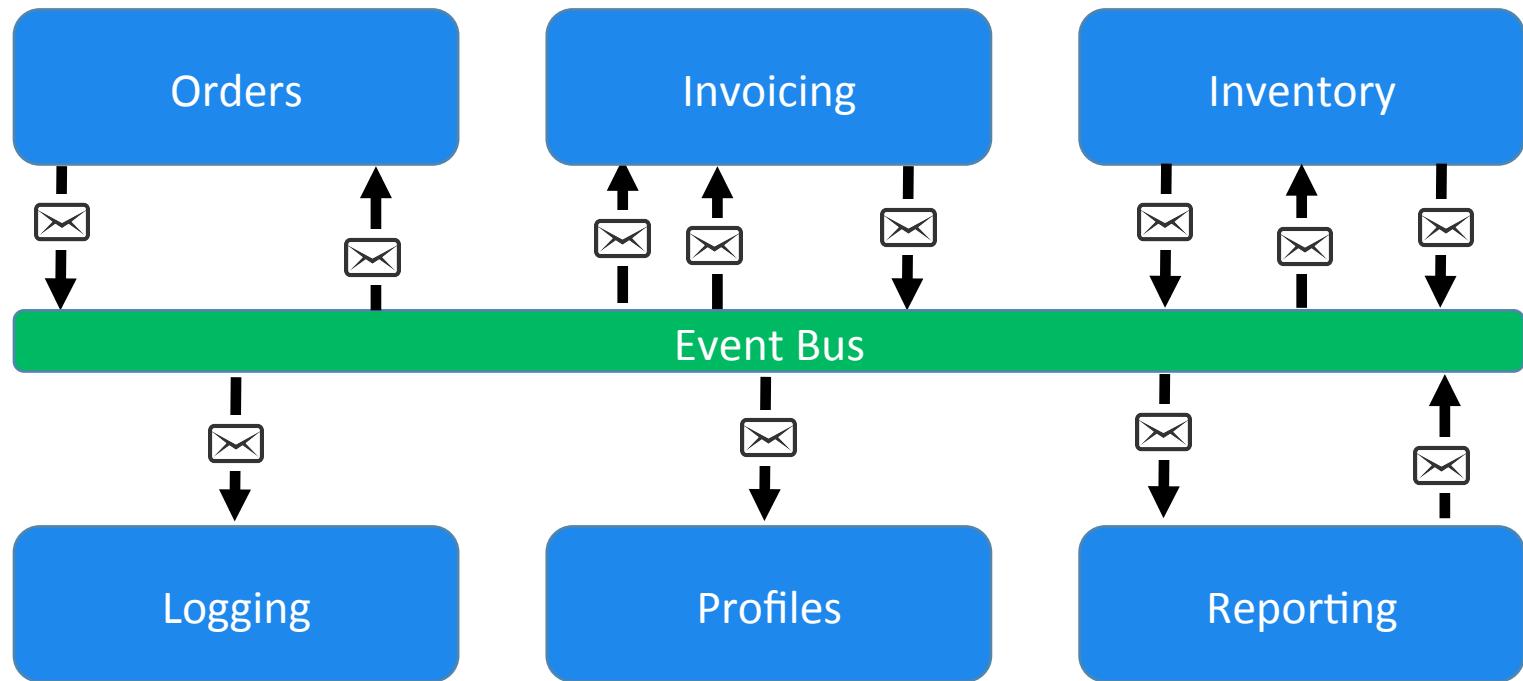
# Event Driven Architecture

- An Event is represents something that happens in a domain
  - Customer Submits Order
  - Customer Billed
  - Payment Received
  - Order Ready for Shipment
  - Order Shipped
- While Events and their payload are designed at the enterprise system level, their implementations are left to the specific subsystems.

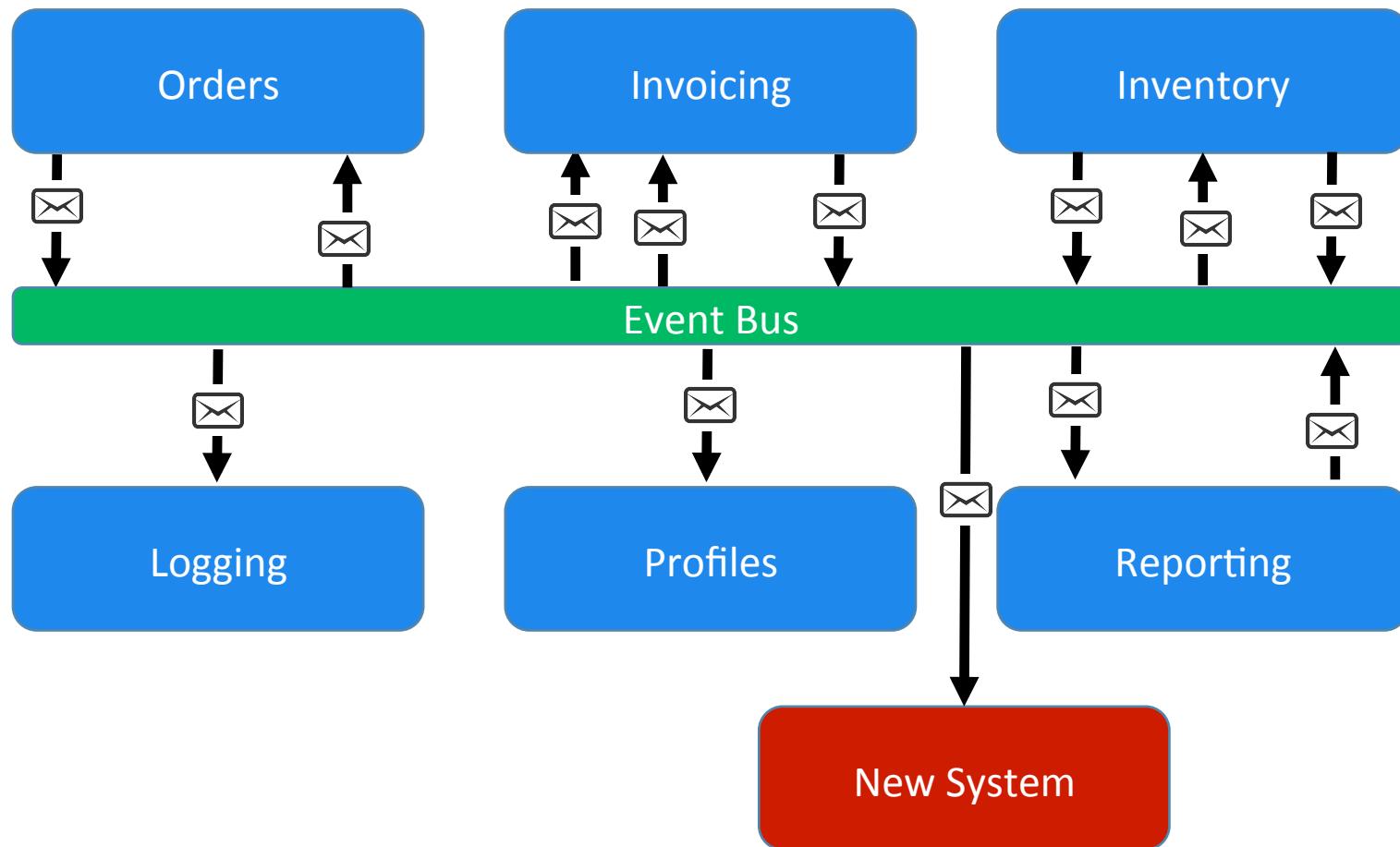
# Event Driven Architecture

- No point-to-point integrations
- Loosely coupled, highly scalable systems
- Loosely coupled, TEAMS
- Easier to test
- Easier to change
- Topology agnostic

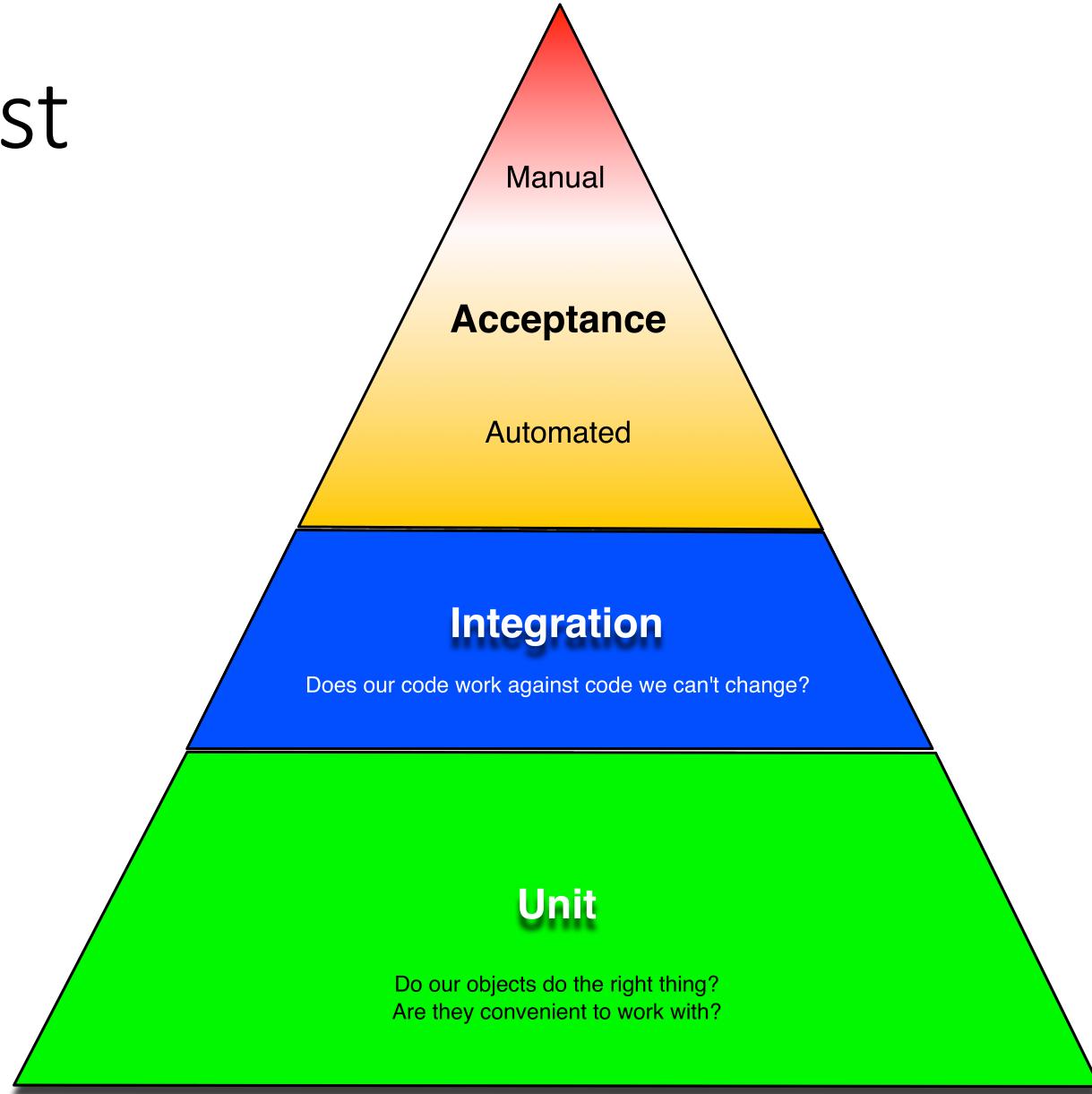
# Loosely Coupled



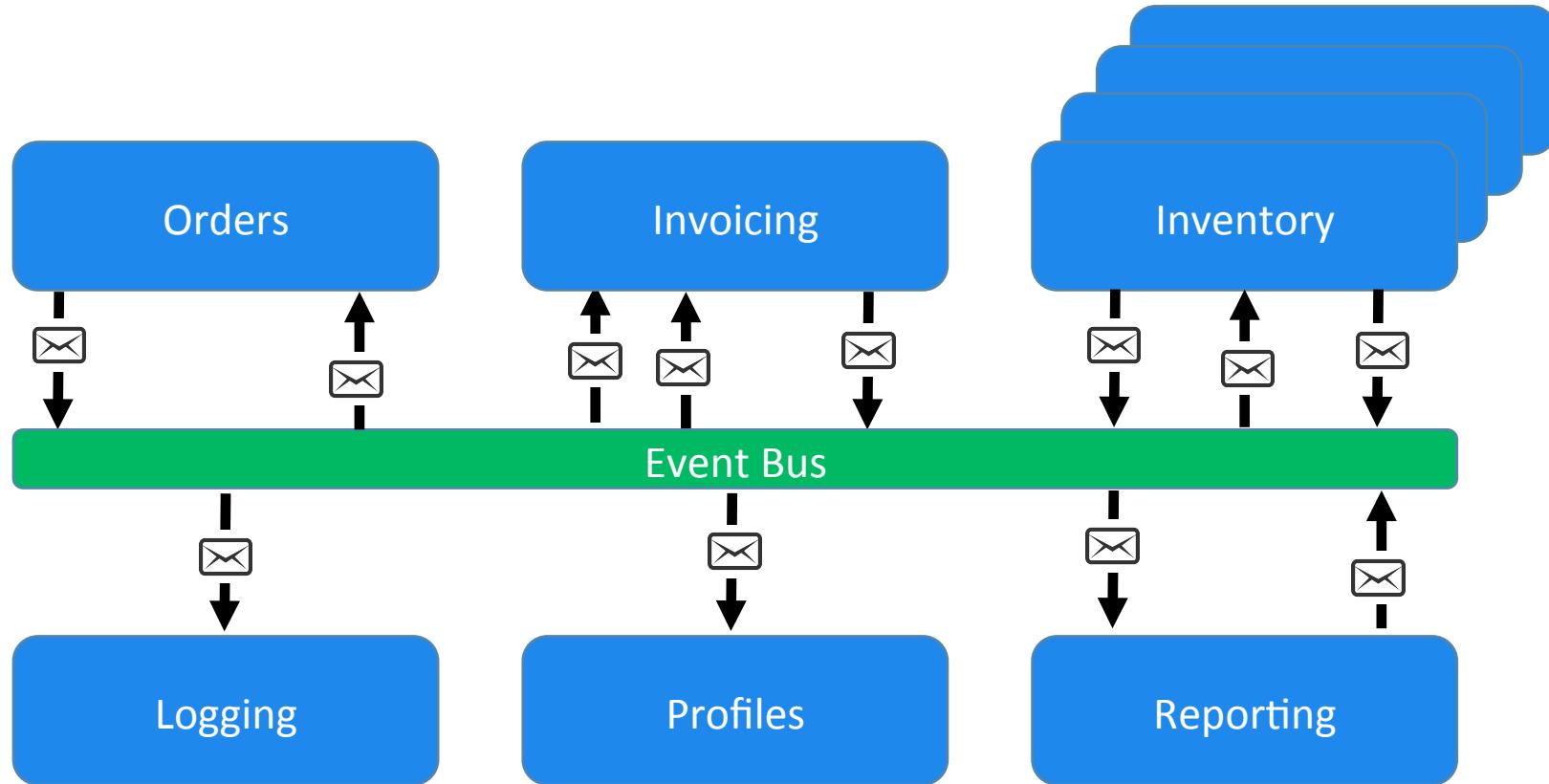
# Easy to Integrate New Systems



# Easy to Test

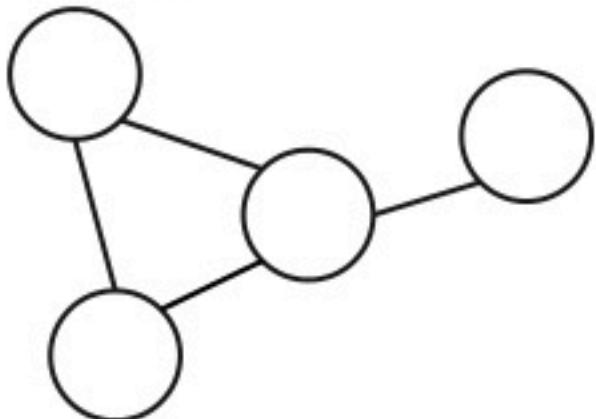


# Scales

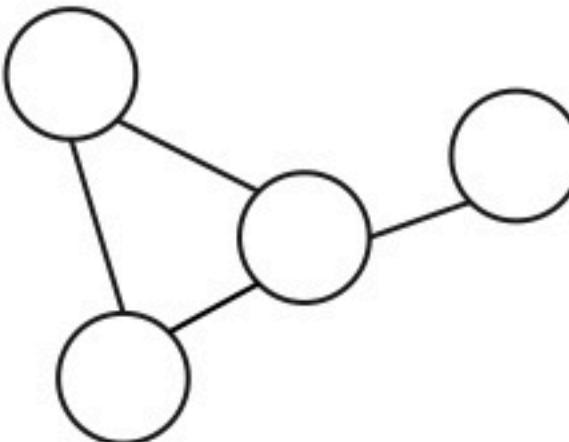


# conway's law

new system:



organization:

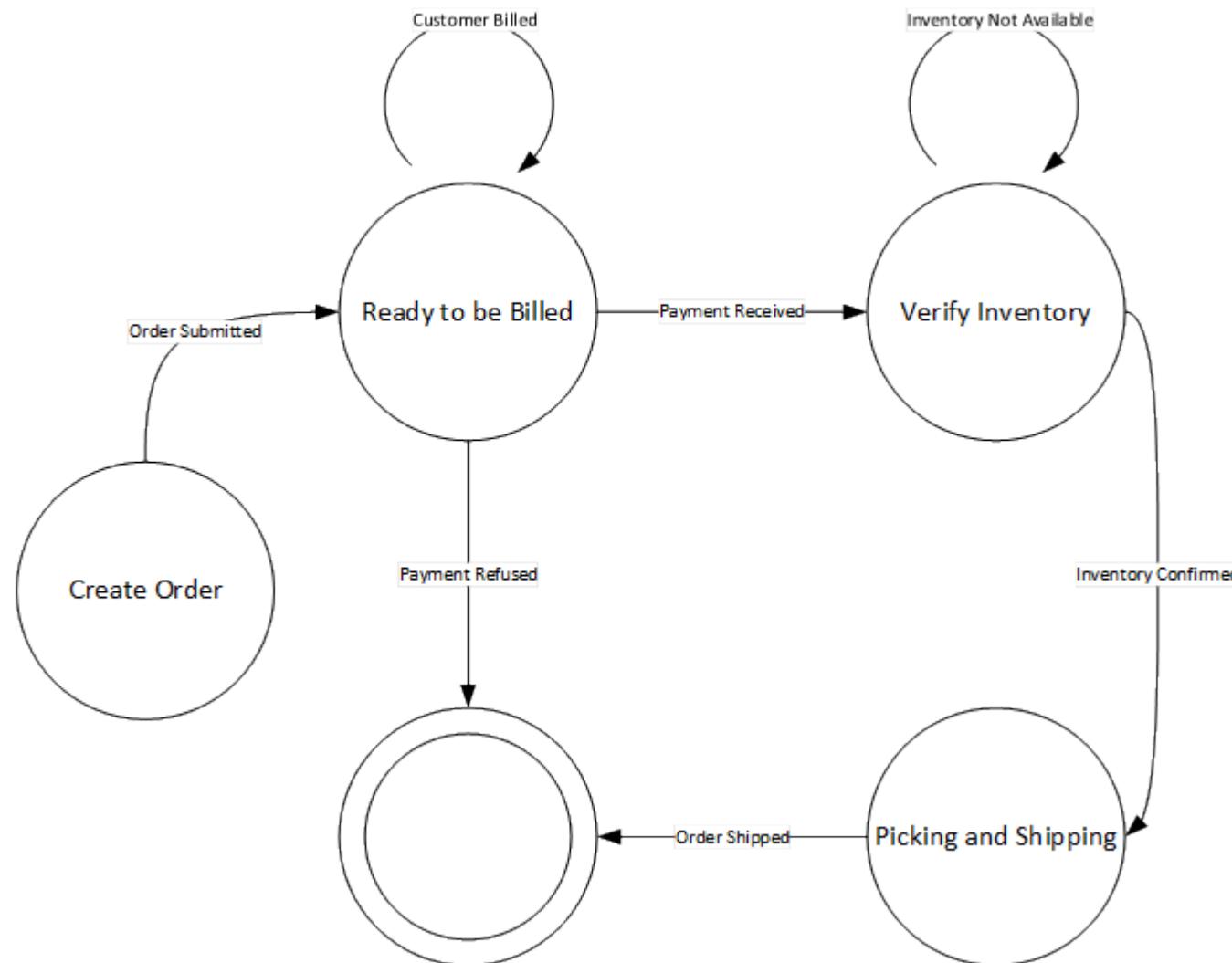


***The basic thesis of this article is that organizations which design systems (in the broad sense used here) are constrained to produce designs which are copies of the communication structures of these organizations.***

--- Melvin Conway

((CENTRIC))

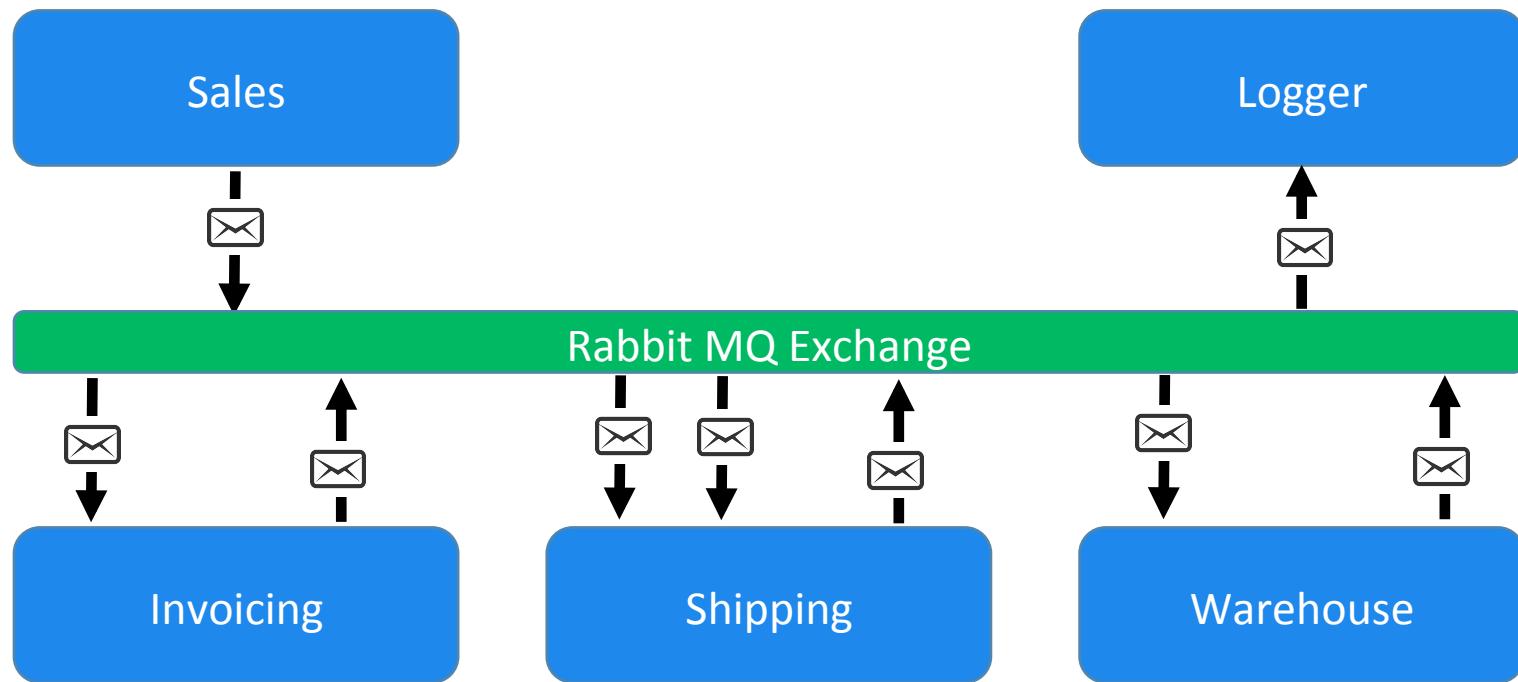
# Easily Modeled as a finite state diagram



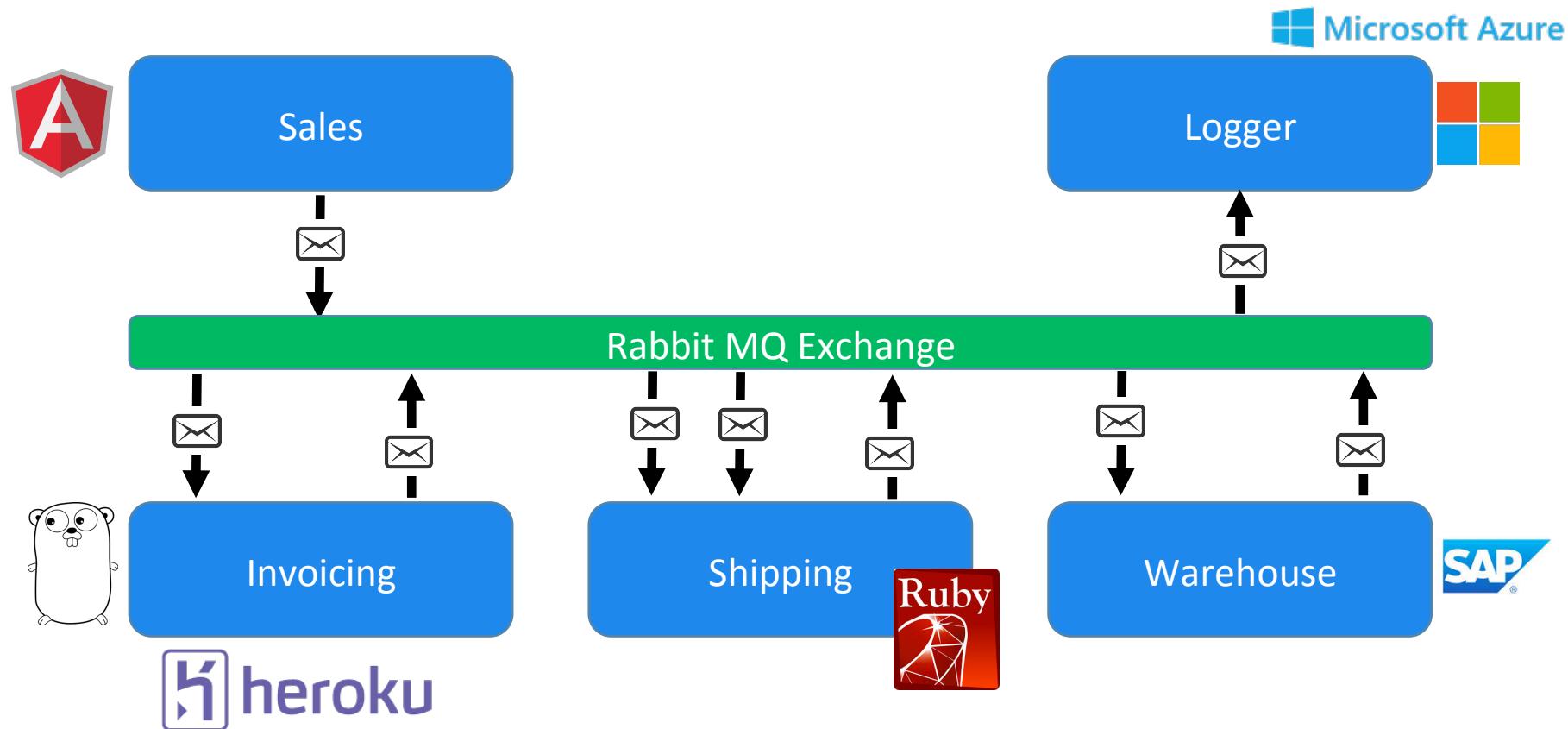
# Other benefits

- Ease of 3<sup>rd</sup> party integration
- Existing systems can be ‘wrapped’
- Can be deliberate about scaling
- Fault tolerant
- Event messages can be logged and ‘replayed’
- Can test subsystems in isolation

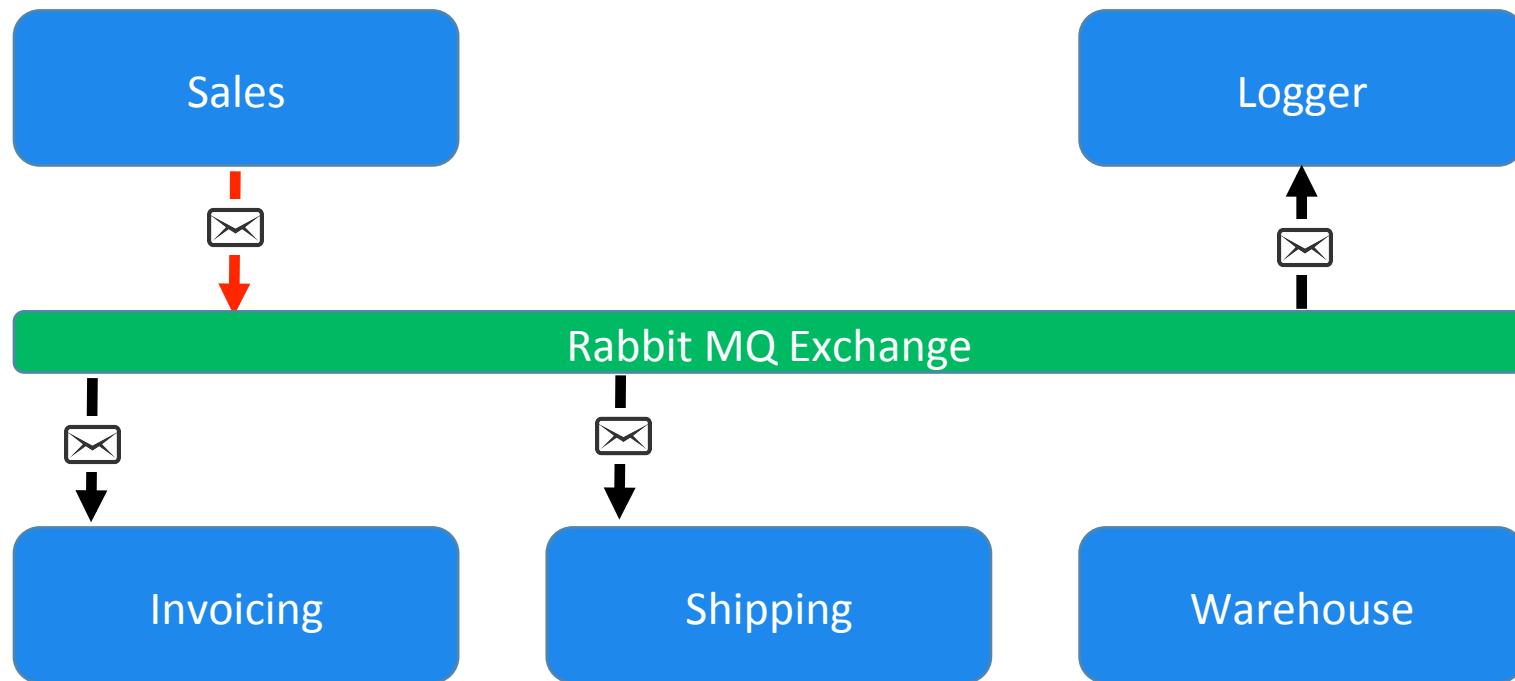
# A Retail System



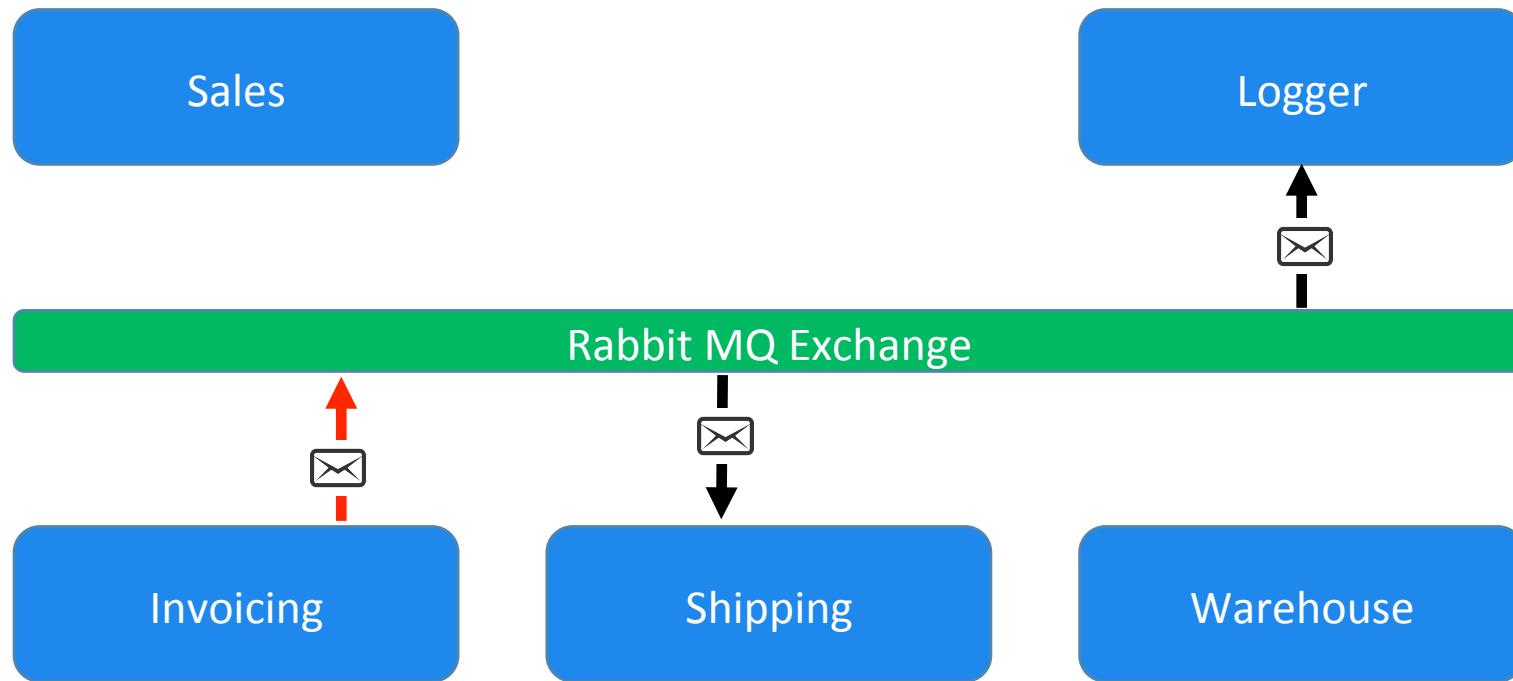
# A Retail System – Platform Agnostic



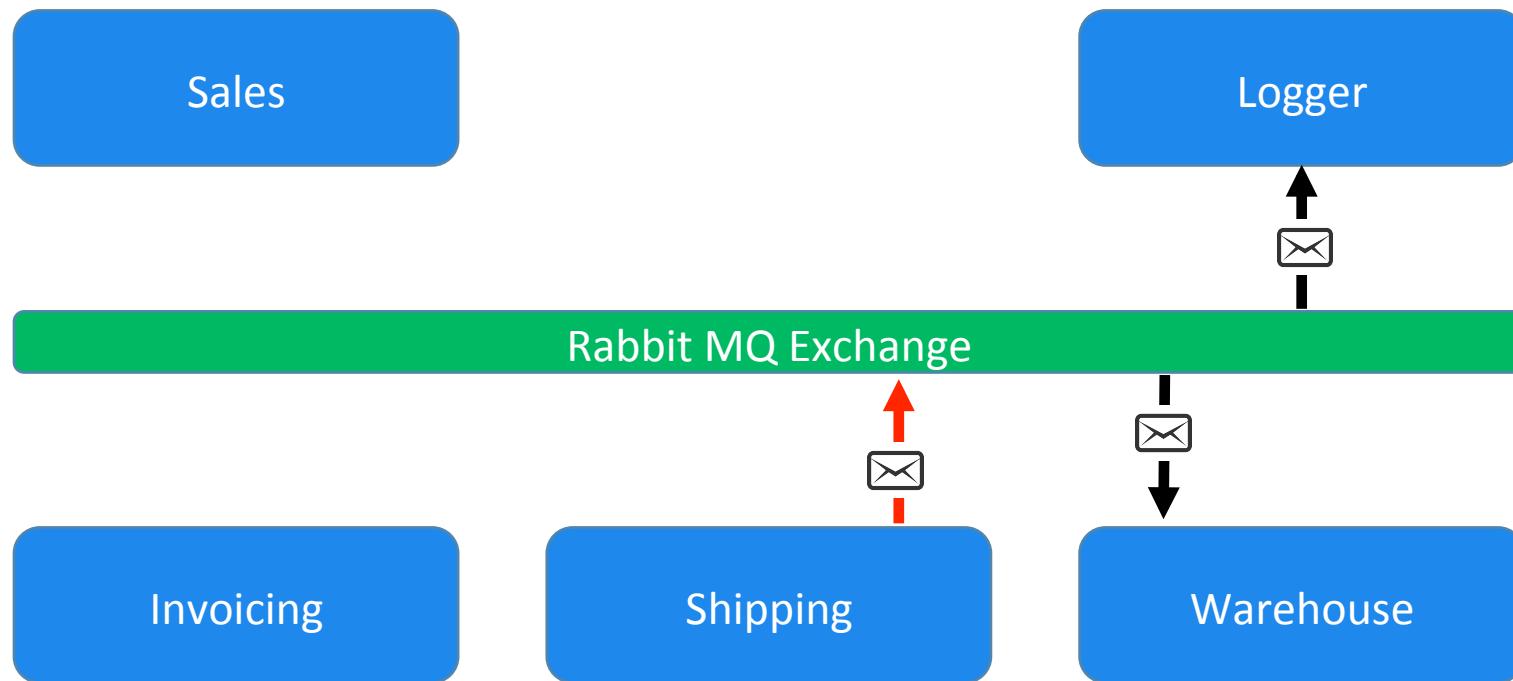
# Order Accepted Event is generated



# Customer Billed Event



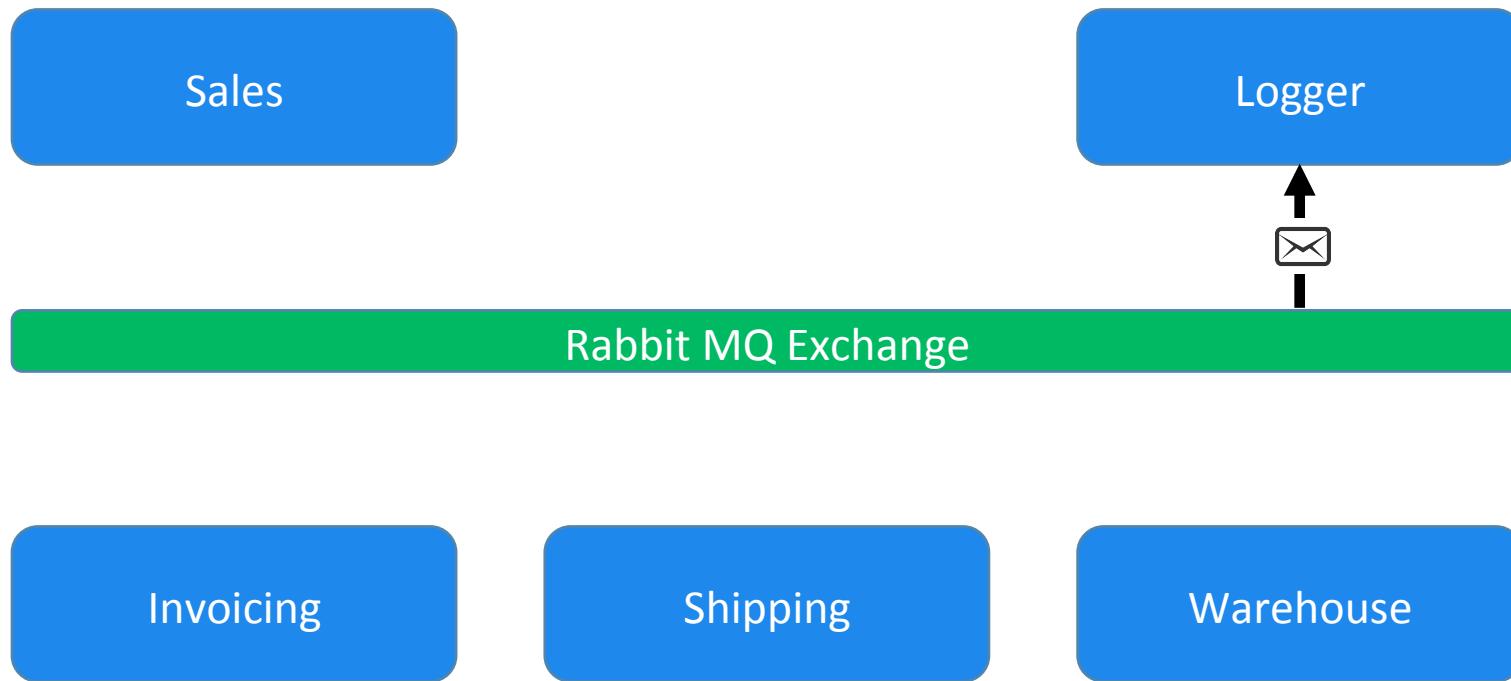
# Order Ready for Shipment Event



# Order Shipped



# Logger Consumes EVERY Event



# Logger Consumes EVERY Event

Results				
Id	OrderId	RoutingKey	Content	WhenReceived
873	2D4BE6EB-6274-4074-8690-C014D3FCA645	order_accepted	{"CustomerId": "834de3b4-fe10-43c6-82a3-6443749c5...", "OrderId": "2d4be6eb-6274-4074-8690-c014d3fca645"}	2015-08-10 15:22:16.013
990	2D4BE6EB-6274-4074-8690-C014D3FCA645	customer_billed	{"OrderId": "2d4be6eb-6274-4074-8690-c014d3fca645"}	2015-08-10 15:24:57.343
1169	2D4BE6EB-6274-4074-8690-C014D3FCA645	order_ready_to_ship	{"OrderId": "2d4be6eb-6274-4074-8690-c014d3fca645"}	2015-08-14 16:04:00.827
1371	2D4BE6EB-6274-4074-8690-C014D3FCA645	order_shipped	{"OrderId": "2d4be6eb-6274-4074-8690-c014d3fca645"}	2015-08-14 16:07:05.793



**DEMO**

# For Further Reading

- THIS presentation (and code) on GitHub

[https://github.com/shawnewallace/choreographed\\_process](https://github.com/shawnewallace/choreographed_process)

- **Stephan Norberg EDA**

<http://www.infoq.com/presentations/Domain-Event-Driven-Architecture>

- **“Programming Without a Call Stack”**

<http://www.enterpriseintegrationpatterns.com/docs/EDA.pdf>

- **“Event-Driven Architecture Overview”**

<http://www.omg.org/soa/Uploaded%20Docs/EDA/bda2-2-06cc.pdf>

# Shawn Wallace

Work:

[shawn.wallace@centricconsulting.com](mailto:shawn.wallace@centricconsulting.com)

Twitter: @ShawnWallace

Blog: blog.shawnewallace.com

<http://www.about.me/shawnwallace>



[www.centricconsulting.com](http://www.centricconsulting.com)



- Event-driven architecture is a versatile approach to designing complex software systems. These systems tend to be easier to model for an organization as EDA embraces Conway's Law and can often be modeled as a finite state machine. Event-driven architecture is not a new concept, but as more organizations embrace micro services, this approach to system design has become appropriate in more situations. In this talk we'll: introduce event-driven architecture, talk about key components, look at a sample implementation of a retail system.