



The Sector Specialists

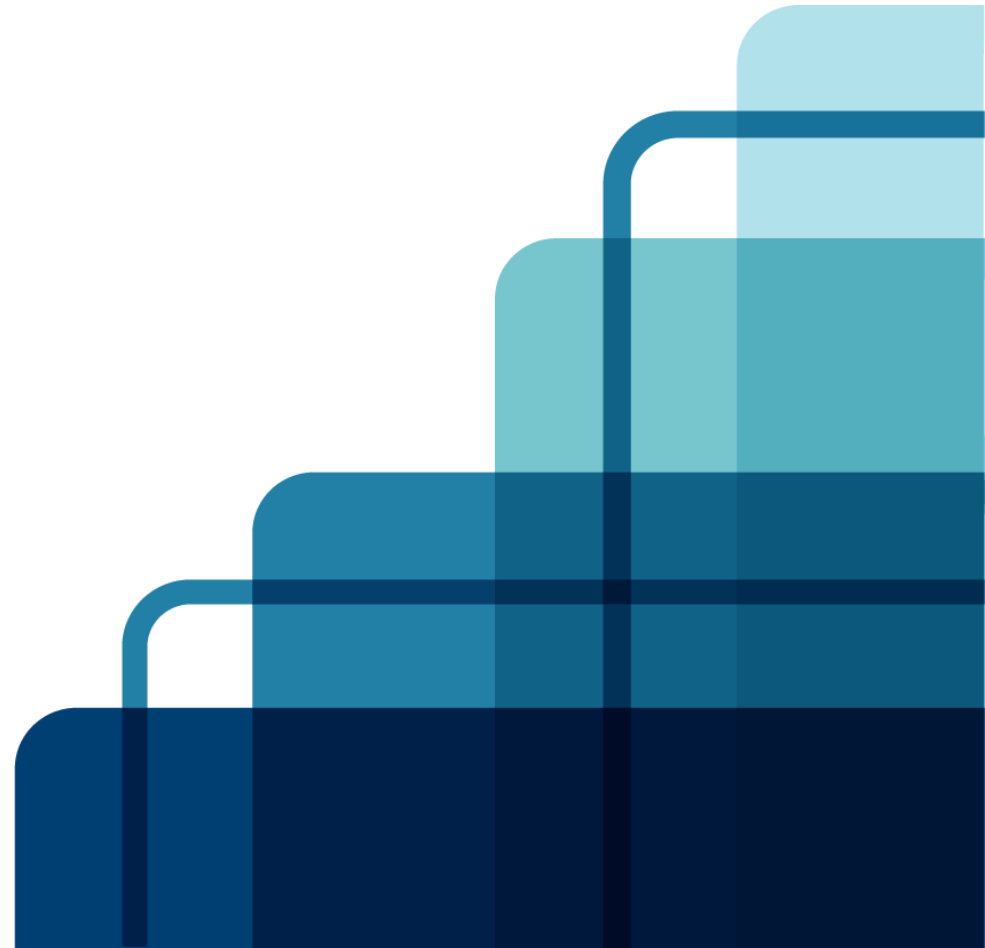
Messaging in .NET

Prepared by: Maciej Paszta

Submitted on: 27-09-2014

Version: 1.0

Confidential



Messaging in .NET

Request-reply communication

Topic exchange

Message passing

Other features

AMPQ

Documentation

Basic message flow

Direct exchange

Fanout exchange

Request-reply communication

- Basic concept in communication supported by most development environments
- Point-to-Point communication
- Clear separation between a client and a server
- Every request requires a reply
- Synchronous processing
- Problems
 - ▶ notifications
 - ▶ multiple replies to a single message
 - ▶ load balancing
 - ▶ failover

Message passing

- Concept similar to traditional or electronic mail
- Asynchronous communication
- Middle-tier software required for message passing - broker
- Various routing (addressing) schemes
- Several communication patterns
 - ▶ request-reply (RPC)
 - ▶ one-to-one
 - ▶ one-to-many
- Client/Server becomes Publisher/Consumer
- **Not only network communication!**

Messaging solutions

- Software:
 - ▶ MSMQ
 - ▶ Apache ActiveMQ
 - ▶ Windows Azure Service Bus
 - ▶ IBM WebSphereMQ
 - ▶ RabbitMQ
- Hardware
 - ▶ Solace
 - ▶ Tervela
 - ▶ Apigee
- What to choose?
- Do all solutions provide the same concepts, terminology and features?

AMPQ

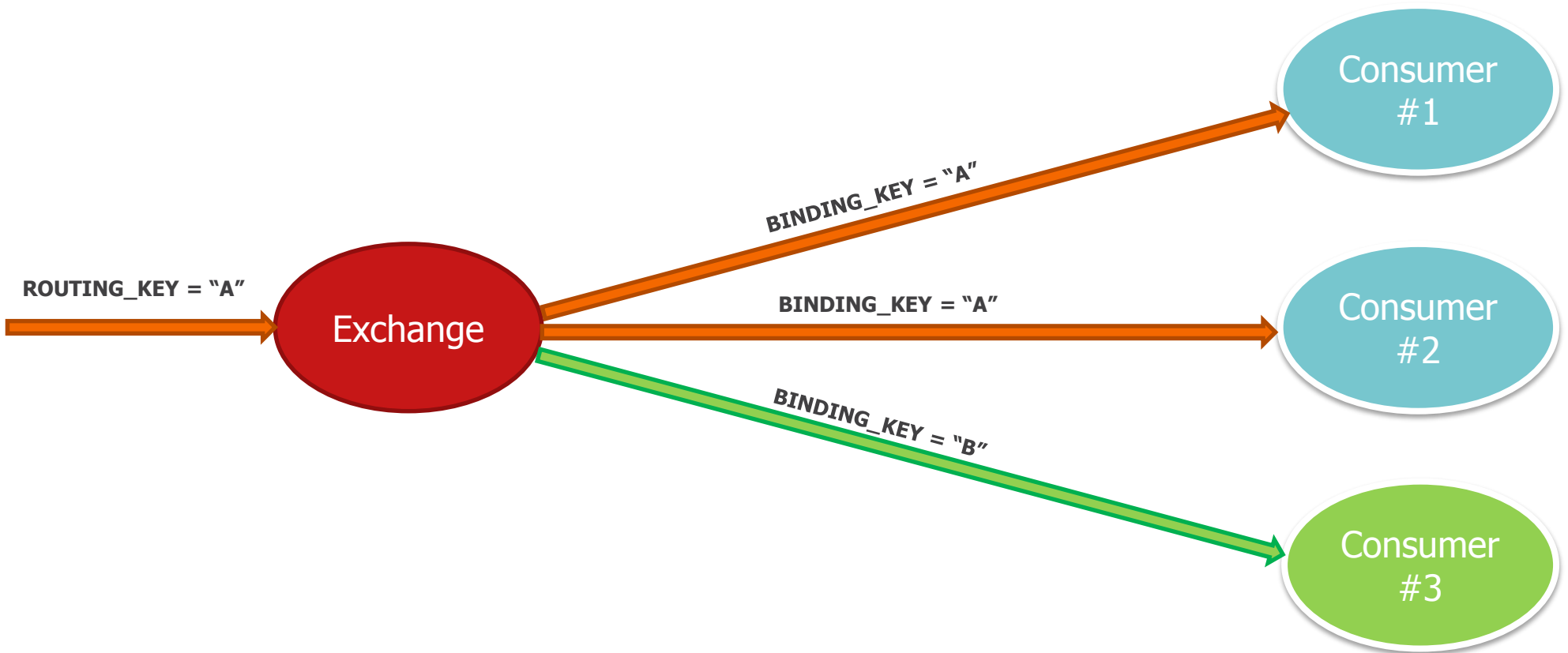
- Advanced Message Queueing Protocol
- Wire format
- Relies on TCP/IP for packet exchange
- Standard naming of communication components
- Mandates the behavior of the message provider and the client
- Provides:
 - ▶ Flow control
 - ▶ Message-delivery guarantees
- Includes features such as:
 - ▶ queueing
 - ▶ routing (point-to-point, publish-subscribe)
 - ▶ reliability
 - ▶ security (SASL/TSL)
 - ▶ load balancing (consumer not queue based)
- Originated from JPMorgan Chase

Basic message flow



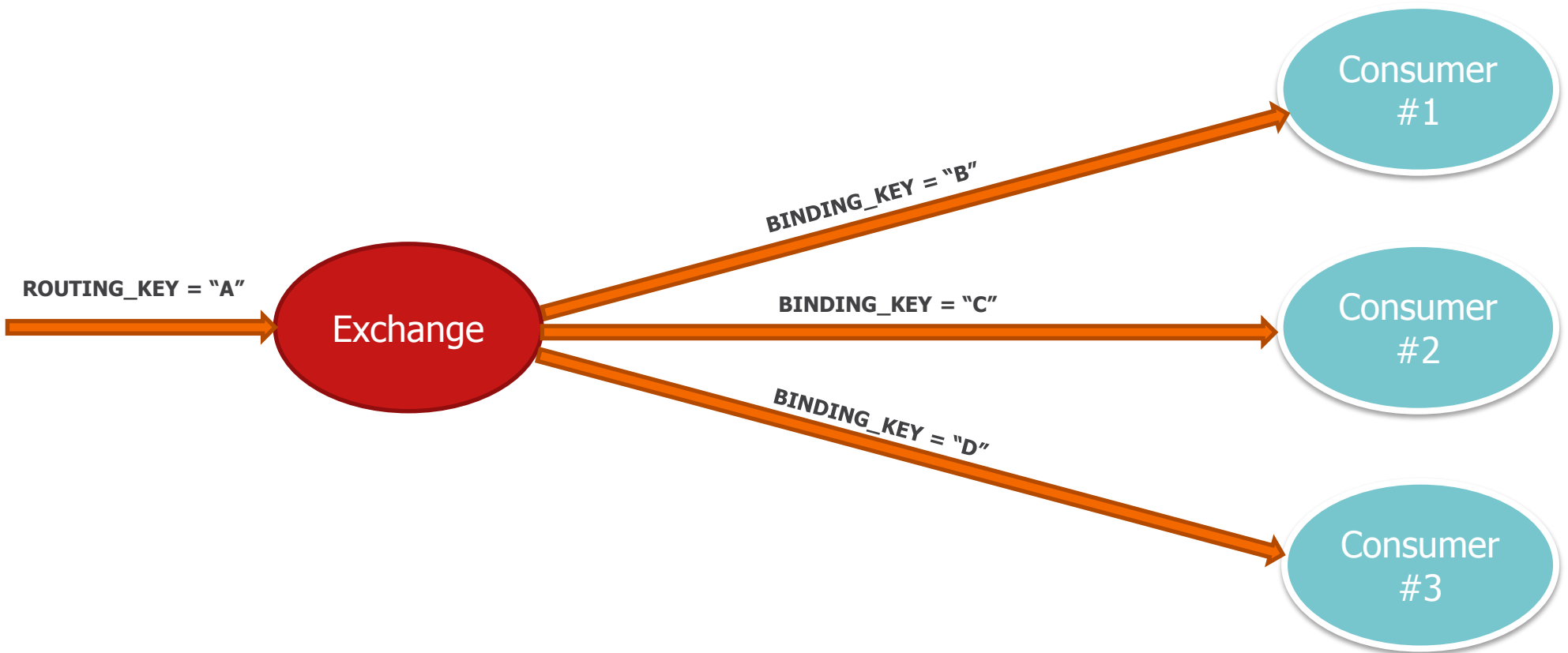
Direct exchange

- Message will be delivered to queues where routing key matches exactly the binding key



Fanout exchange

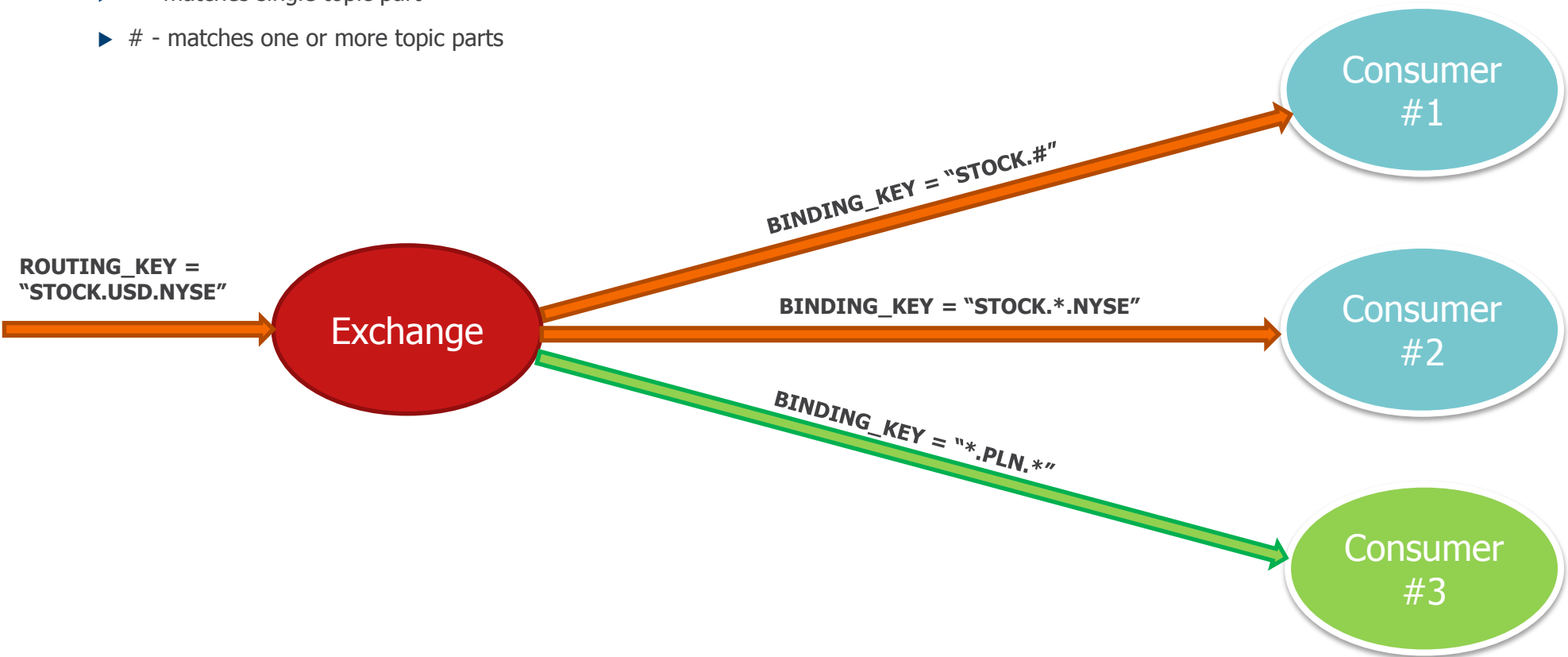
- Message will be delivered to all queues, regardless of the routing key



Topic exchange

- Message will be delivered to queues with matching (based on a pattern) binding keys:

- ▶ Parts of topic delimited by "."
- ▶ * - matches single topic part
- ▶ # - matches one or more topic parts



Other features

- Load balancing (round robin) across all consumers bound to the same queue
 - ▶ Message dispatching can be controller with ACK and prefetch
- Message acknowledgments (until no ACK is received, message is saved)
- Ability to create broker clusters
 - ▶ Failover capabilities
 - ▶ Load balance across consumer connected to different brokers
 - ▶ Increase scalability

Documentation

- Advanced Message Queueing Protocol: <http://www.amqp.org/>
- RabbitMQ: <http://www.rabbitmq.com/>
- Apache ActiveMQ: <http://activemq.apache.org/>
- MSDN: <http://msdn.microsoft.com/>

Thank You!