- Introduction AMQP
- (Advanced Message Queuing Protocol)
- Protocol Working
- AMQP Features
- Message Delivery Guarantees
- AMQP frame types
- Components of AMQP
- AMQP exchanges types
- AMQP Features
- Applications





AMQP





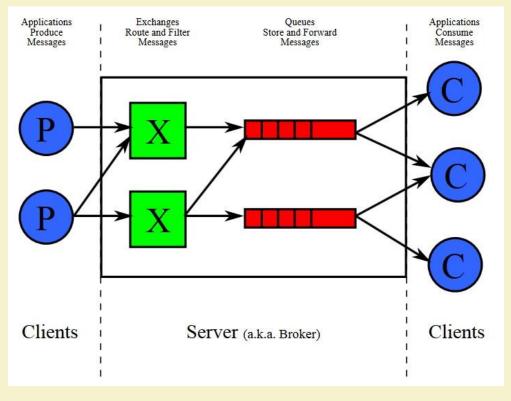
Introduction

- ✓ Advanced Message Queuing Protocol.
- ✓ Open standard for passing business messages between applications or organizations.
- ✓ Connects between systems and business processes.
- ✓ It is a binary application layer protocol.
- ✓ Basic unit of data is a *frame*.
- ✓ ISO standard: ISO/IEC 19464

Source: "Advanced Message Queuing Protocol", Wikipedia (Online)







Source: "The-amqp-model-for-wikipedia.svg", Wikimedia Commons (Online)





AMQP Features

Organizations **Technologies** Time Space

Connects across





Features







Message Delivery Guarantees

- ✓ At-most-once
 - each message is delivered once or never
- √ At-least-once
 - each message is certain to be delivered, but may do so multiple times
- ✓ Exactly-once
 - message will always certainly arrive and do so only once

Reference: "OASIS AMQP version 1.0, sections 2.6.12-2.6.13". OASIS AMQP Technical Committee





AMQP Frame Types

- ✓ Nine AMQP frame types are defined that are used to initiate, control and tear down the transfer of messages between two peers:
 - Open (connection open)
 - Begin (session open)
 - Attach (initiate new link)
 - Transfer (for sending actual messages)
 - Flow (controls message flow rate)
 - Disposition (Informs the changes in state of transfer)
 - Detach (terminate the link)
 - End (session close)
 - Close (connection close)





Components

Exchange

- Part of Broker
- Receives messages and routes them to Queues

Queue

- Separate queues for separate business processes
- Consumers receive messages from queues

Bindings

 Rules for distributing messages (who can access what message, destination of the message)





AMQP Exchanges







AMQP Features

- ✓ Targeted QoS (Selectively offering QoS to links)
- ✓ Persistence (Message delivery guarantees)
- ✓ Delivery of messages to multiple consumers
- ✓ Possibility of ensuring multiple consumption
- ✓ Possibility of preventing multiple consumption
- ✓ High speed protocol





Applications

- ✓ Monitoring and global update sharing.
- ✓ Connecting different systems and processes to talk to each other.
- ✓ Allowing servers to respond to immediate requests quickly and delegate time consuming tasks for later processing.
- ✓ Distributing a message to multiple recipients for consumption.
- ✓ Enabling offline clients to fetch data at a later time.
- ✓ Introducing fully asynchronous functionality for systems.
- ✓ Increasing reliability and uptime of application deployments.

