

Introducing MSMQ

Elton Stoneman
geekswithblogs.net/eltonstoneman
@EltonStoneman



pluralsight 
hardcore dev and IT training

Introducing MSMQ



Message queuing
built into **Windows**
and **Windows Server**

Works with **Active
Directory** for security
& discoverability

Core pattern
support &
advanced features

Goals

MSMQ technology overview

How it works, features it provides

Deployment & administration

Installation & configuration, administering queues

.NET client library

Usage, administration and operational features

Pattern support

Fire-and-forget, request-response, publish-subscribe

MSMQ

Microsoft Message Queueing

What is MSMQ?



Message Queueing
built into Windows,
as a Windows Service

5.0

Currently **version 5.0**
(Windows 7 and
Server 2008 R2)



Store-and-forward
queue technology

Public and Private Queues

Public

- Published in Active Directory
- Can query AD to find queues
- Integrates with Windows security

Private

- Not integrated with AD
- But still available for public use
- Security features not available

Private Queue Addresses

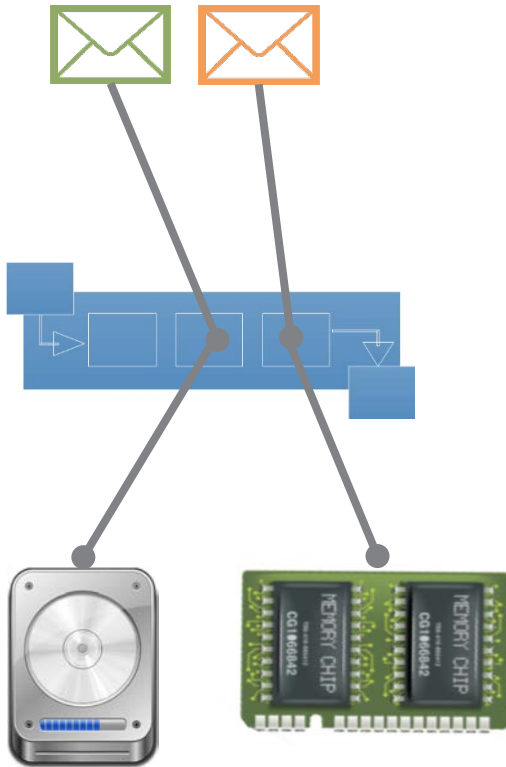
PATH

- *{machine}\private\$\{queueName}*
- *.\private\$\unsubscribe*
- *SC-MQ-01\private\$\unsubscribe*

DIRECT

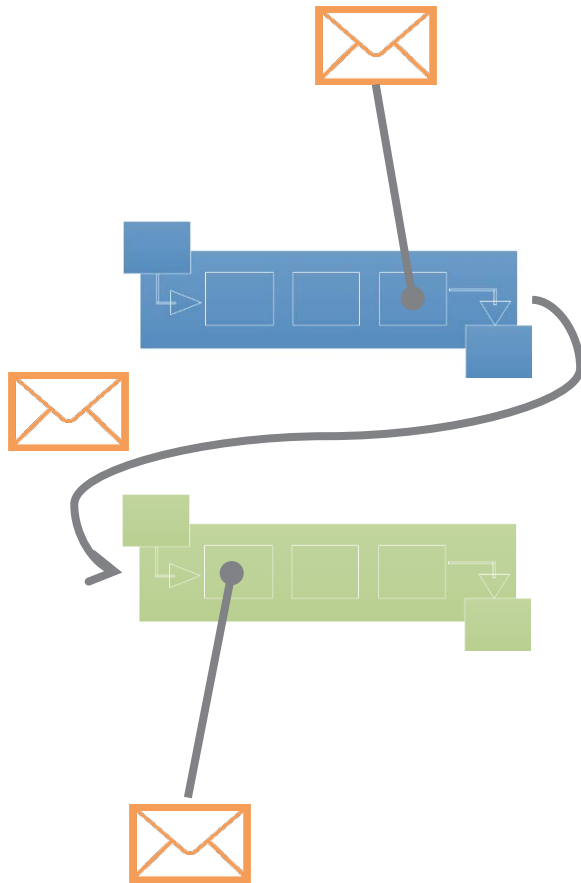
- *DIRECT={protocol}:{address}\private\$\{queueName}*
- *DIRECT=TCP:192.168.2.140\private\$\unsubscribe*
- *DIRECT=OS:SC-MQ-01\private\$\unsubscribe*

Durability & Reliability



Messages not persisted by default
Use the **recoverable** flag

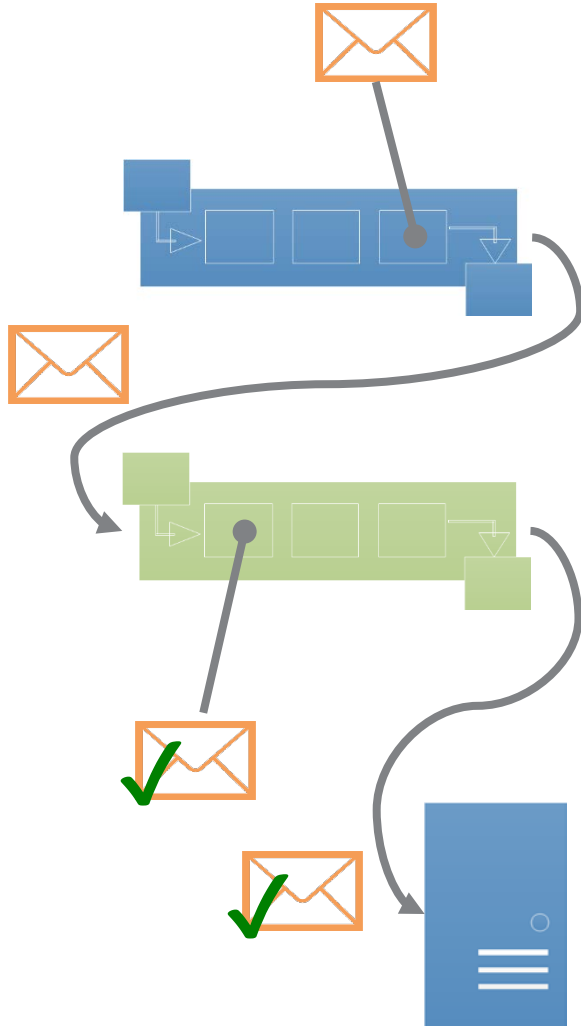
Durability & Reliability



Messages not persisted by default
Use the **recoverable** flag

Store and forward
Persist local; forward; persist remote

Durability & Reliability

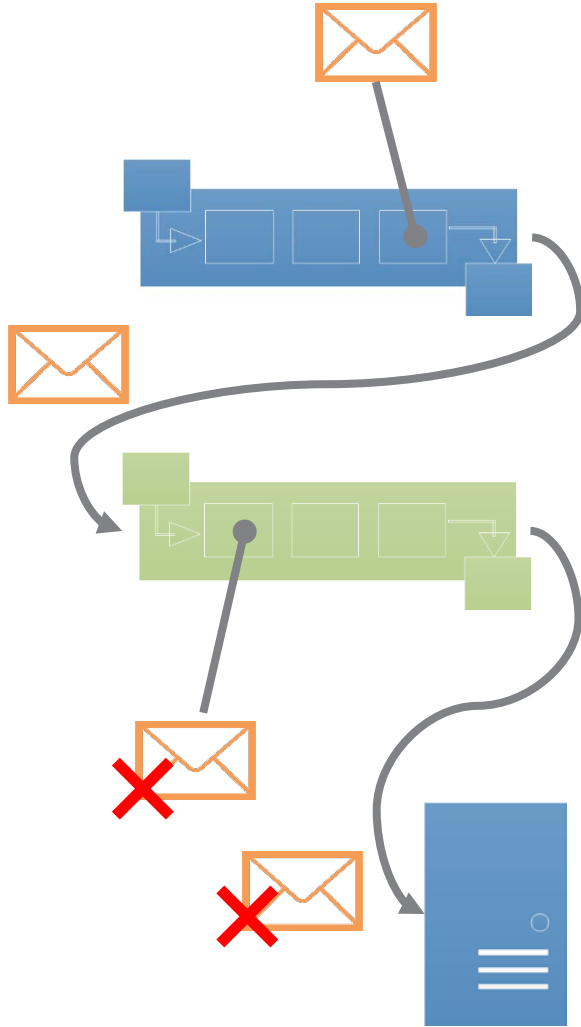


Messages not persisted by default
Use the **recoverable** flag

Store and forward
Persist local; forward; persist remote

Acknowledgement types
Positive – reaches destination or read

Durability & Reliability

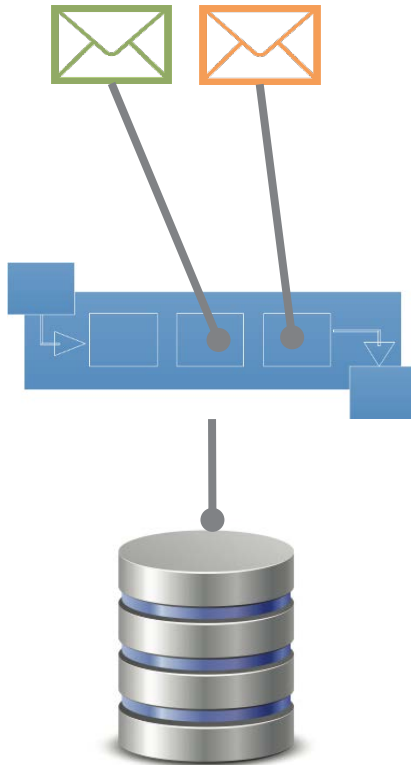


Messages not persisted by default
Use the **recoverable** flag

Store and forward
Persist local; forward; persist remote

Acknowledgement types
Positive – reaches destination or read
Negative – failed to send or be read

Durability & Reliability



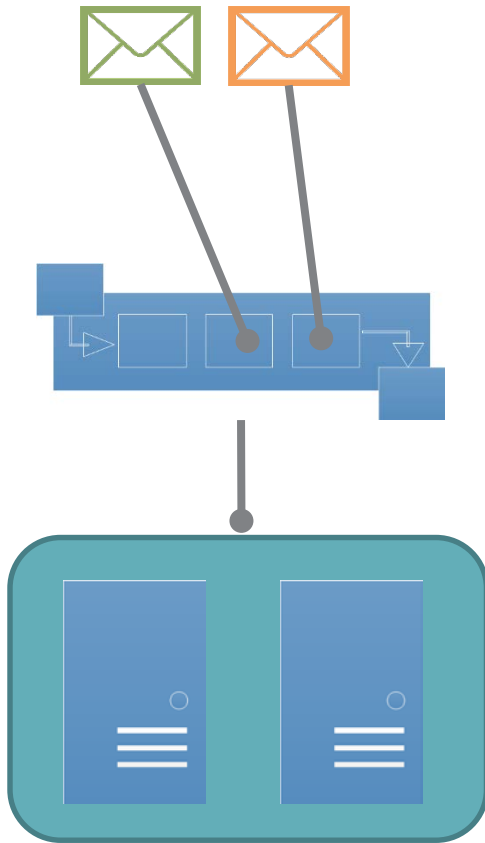
Recoverable messages saved to disk

As reliable as your disk/RAID/SAN

Standard disk storage

Not standard back up and restore

Durability & Reliability



Recoverable messages saved to disk

As reliable as your disk/RAID/SAN

Standard disk storage

Not standard back up and restore

Clustered MSMQ server instances

For redundancy

Resilient to machine failure

Demo 1: Deploying MSMQ

Feature

Install, configure
and verify MSMQ

Task

Install MSMQ on
Windows and
Windows Server

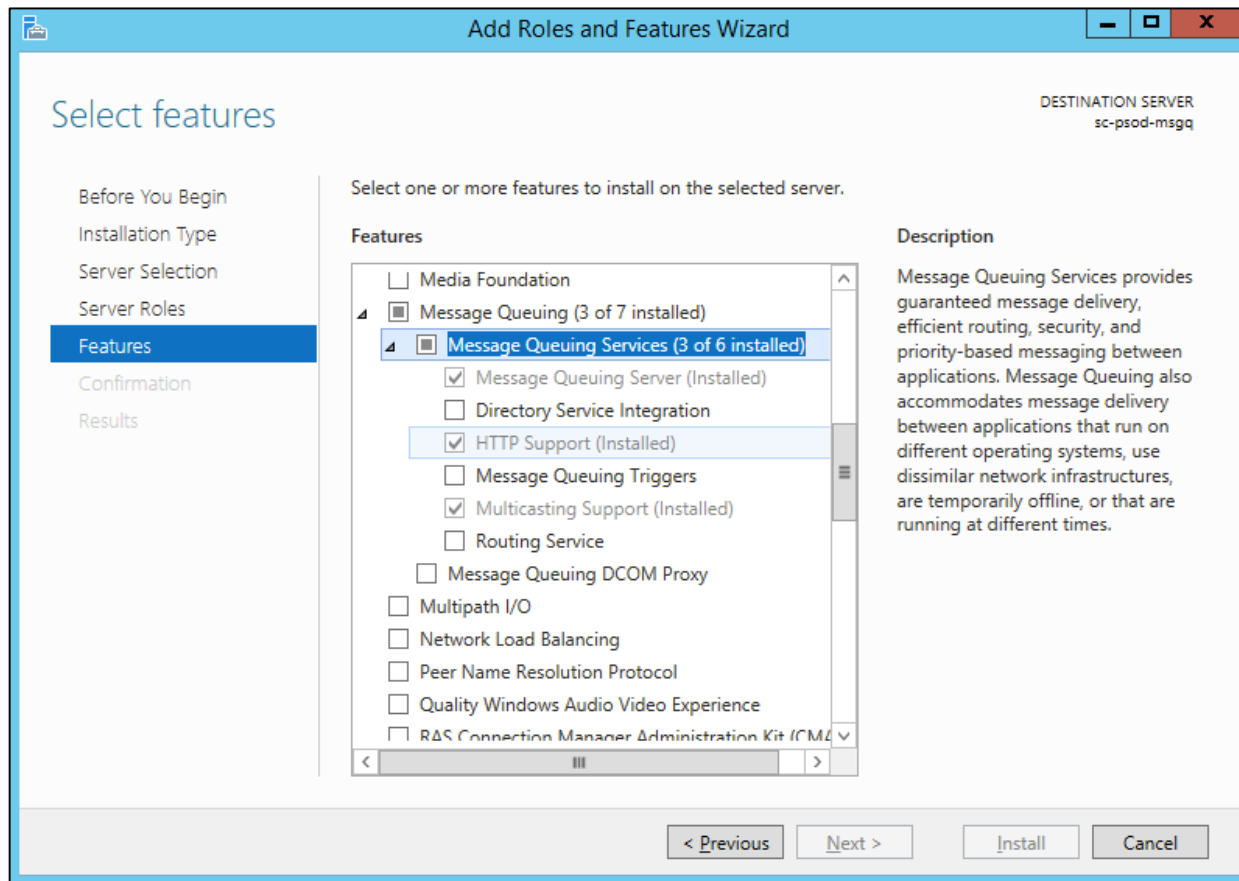
Task

Create queues
with different
properties

Demo 1: Deploying MSMQ

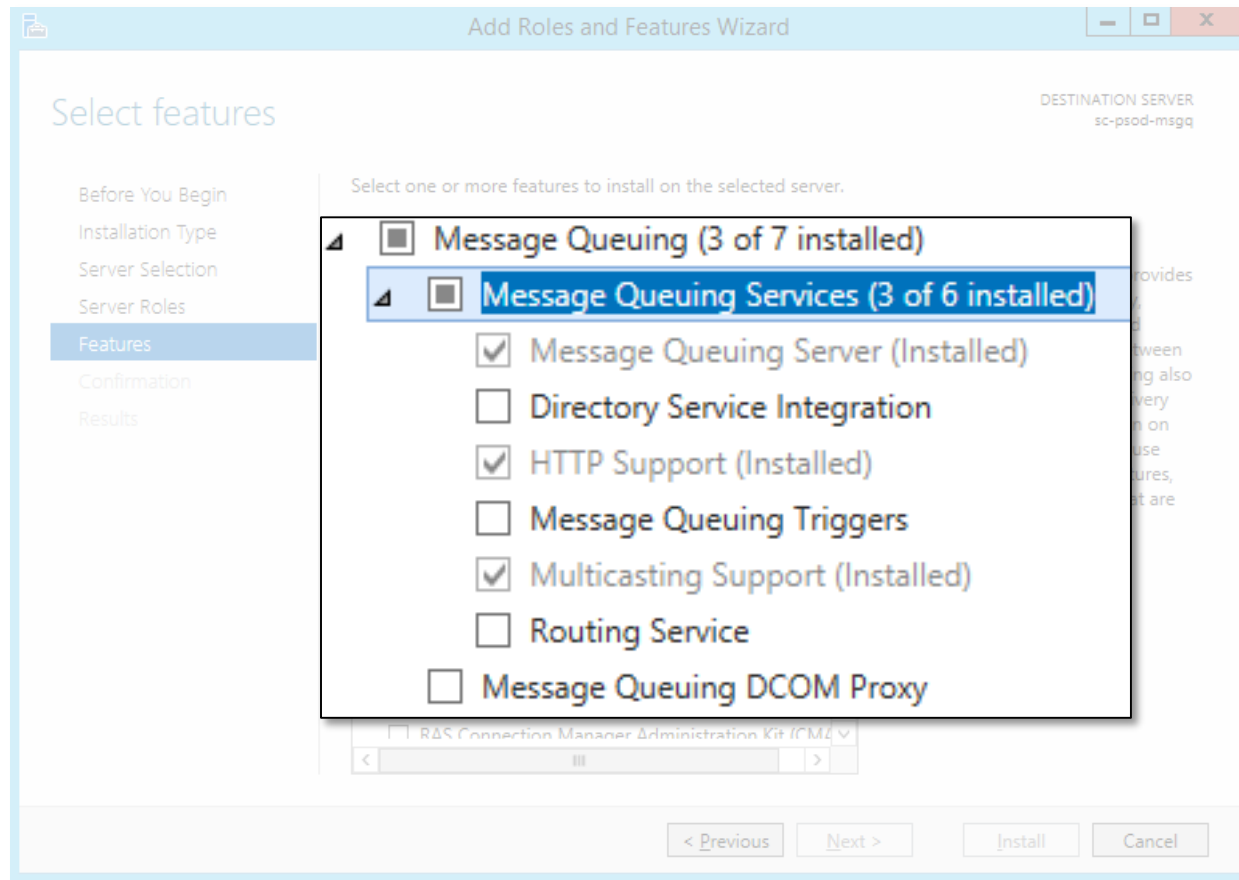
Demo 1: Deploying MSMQ

■ Installation as a feature – Windows Server



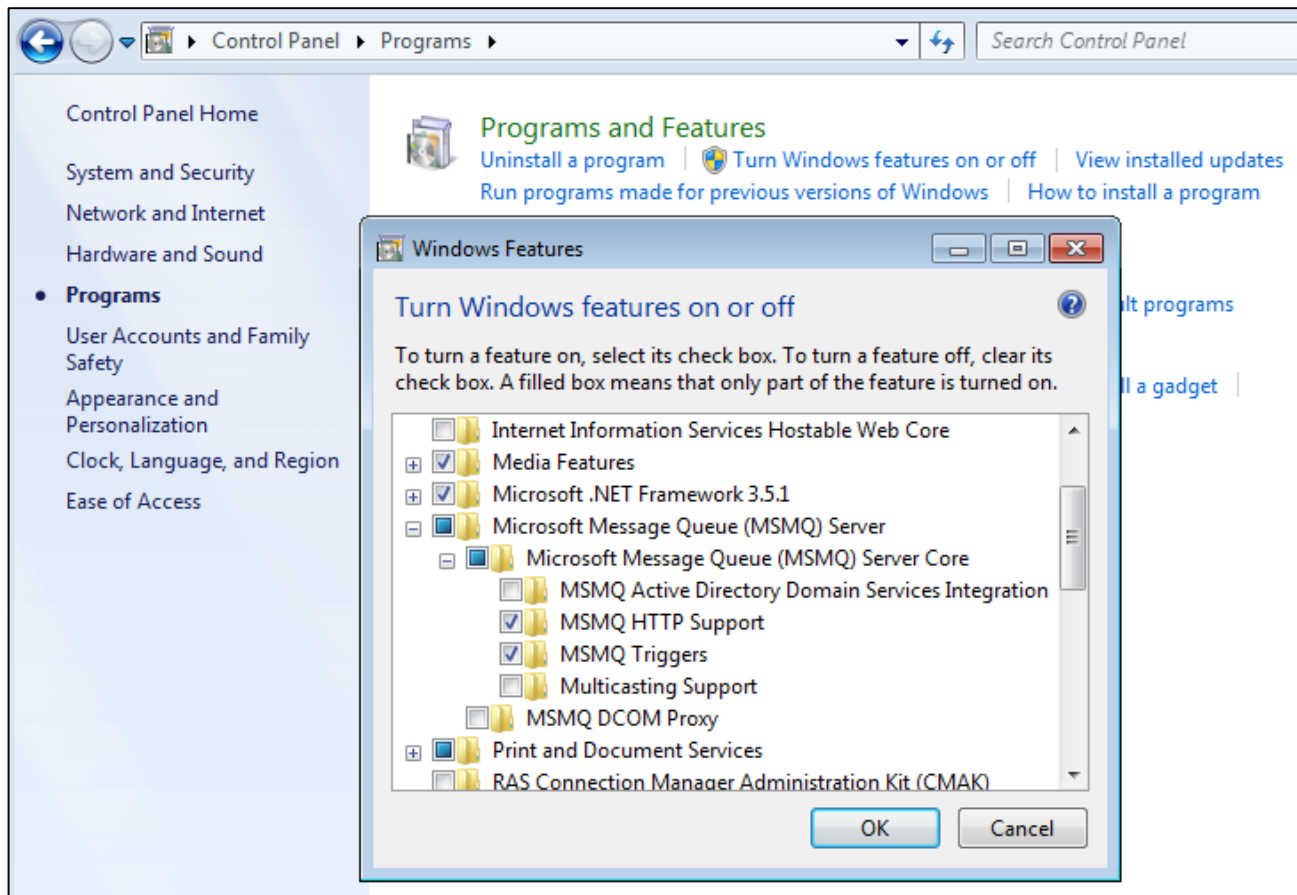
Demo 1: Deploying MSMQ

- Installation as a feature – Windows Server



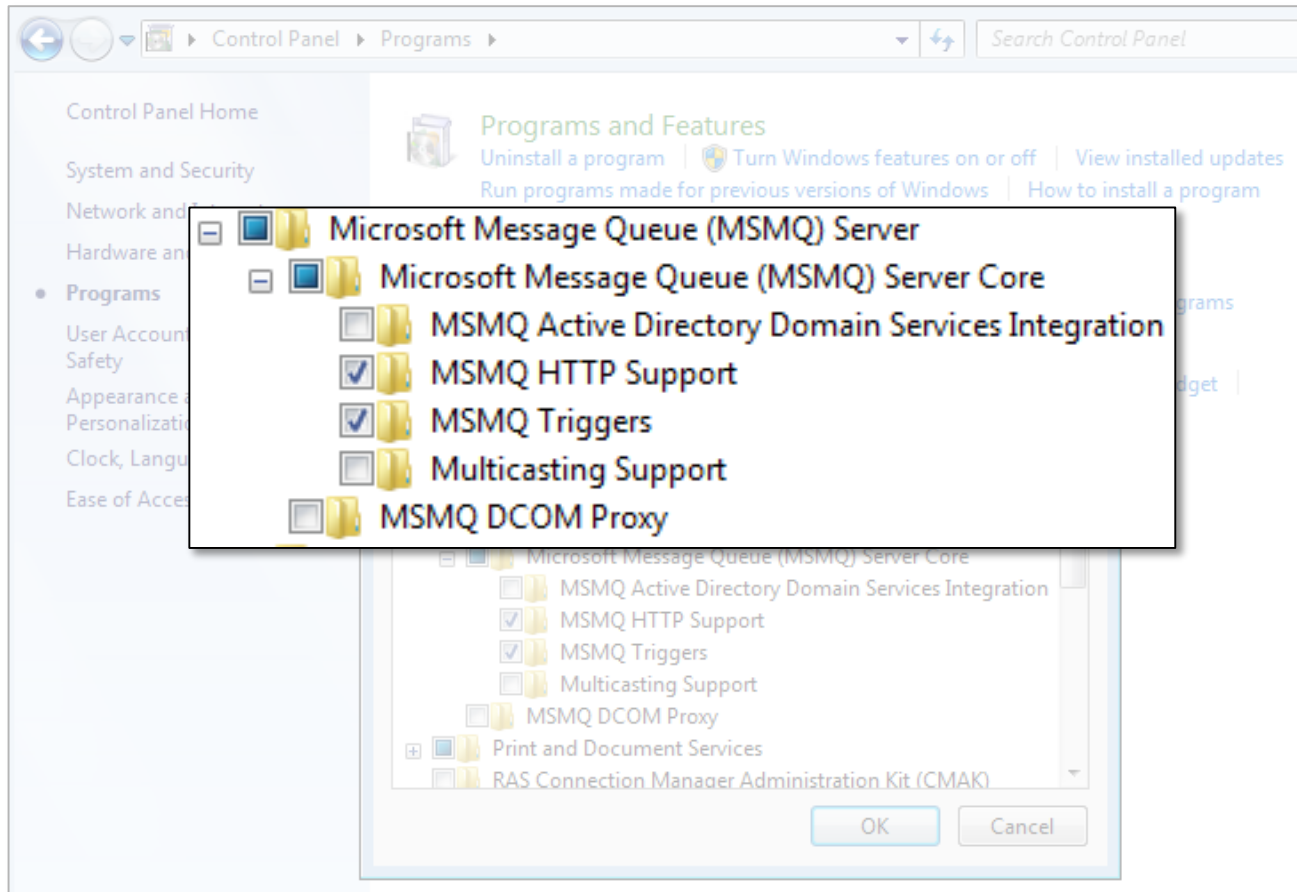
Demo 1: Deploying MSMQ

- Installation as a Windows Component – desktop Windows



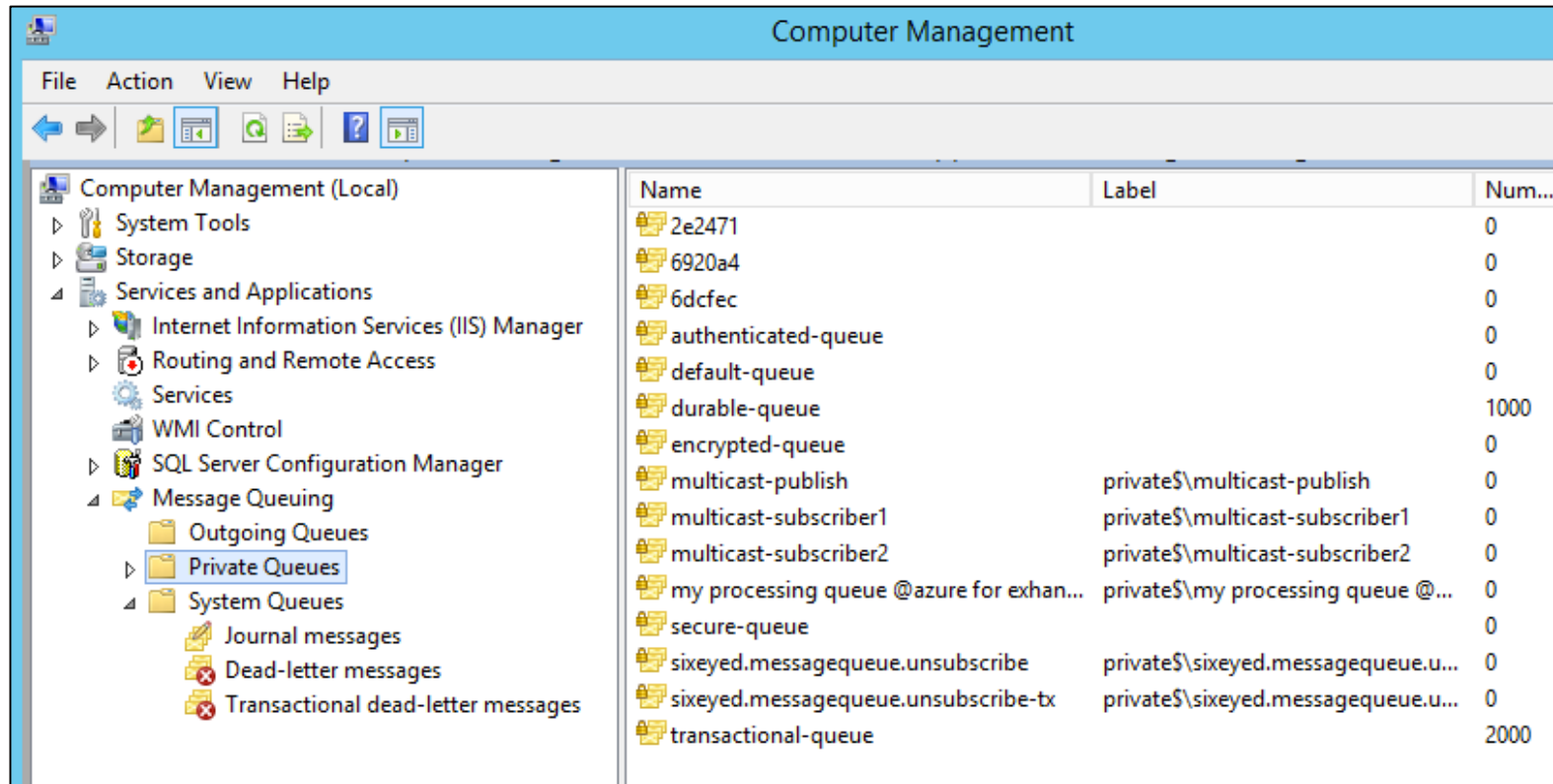
Demo 1: Deploying MSMQ

- Installation as a Windows Component – desktop Windows



Demo 1: Deploying MSMQ

- Administration with Computer Management



Demo 1: Deploying MSMQ

- **Administration with Computer Management**
 - Create queues
 - Specify name
 - Transactional or non-transactional
 - Modify queue properties
 - Journaling
 - Security
 - Inspect queues
 - Message count
 - Message body content
 - Purge messages
 - Delete queues

Using MSMQ with .NET

Client library built
into .NET Framework:
System.Messaging

MSMQ version 4.0
features available
from **.NET 3.0**

Simple to use:
MessageQueue and
Message classes

MessageQueue



Queue administration

Create and modify

Purge and delete

Messaging operations

Send and receive

Default behaviour

XML serialization

Non-recoverable messages

Message



Fine control

Serialization format

Body content

Customized behaviour

Recoverable

Acknowledgement required

Demo 2: Using MSMQ with .NET

Feature

Explore MSMQ
functionality from
.NET

Task

Create and
configure queues

Task

Send messages
using different
queue setups

Task

Clear down and
delete queues

Demo 2: Using MSMQ with .NET

Demo 2: Using MSMQ with .NET

- **MessageQueue class**

- Create transactional and non-transactional queues

```
MessageQueue.Create(@".\private$\default-queue");  
MessageQueue.Create(@".\private$\transactional-queue", true);
```

- Set queue properties

```
MessageQueue.Create(@".\private$\secure-queue");  
var secure = new MessageQueue(@".\private$\secure-queue");  
secure.EncryptionRequired = EncryptionRequired.Body;  
secure.Authenticate = true;
```

Demo 2: Using MSMQ with .NET

- **MessageQueue class**

- Sending messages, non-recoverable

```
queue = new MessageQueue(@".\private$\default-queue");  
for (int i=0; i<1000; i++) {  
    queue.Send("Message: " + i);  
}
```

- Default send properties, recoverable

```
queue = new MessageQueue(@".\private$\durable-queue");  
queue.DefaultPropertiesToSend.Recoverable = true;  
for (int i=0; i<1000; i++) {  
    queue.Send("Message: " + i);  
}
```

Demo 2: Using MSMQ with .NET

- **MessageQueue class**

- Transactional messaging
- One transaction per message

```
queue = new MessageQueue(@".\private$\transactional-queue");  
for (int i=0; i<1000; i++)  
{  
    tx = new MessageQueueTransaction();  
    tx.Begin();  
    queue.Send("Message: " + i, tx);  
    tx.Commit();  
}
```

- Batched – one transaction for all messages

Demo 2: Using MSMQ with .NET

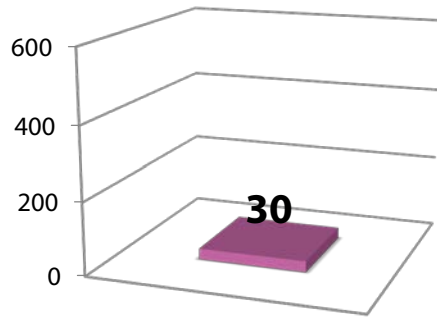
- **MessageQueue class**

- Secure messaging
- Requires Active Directory integration

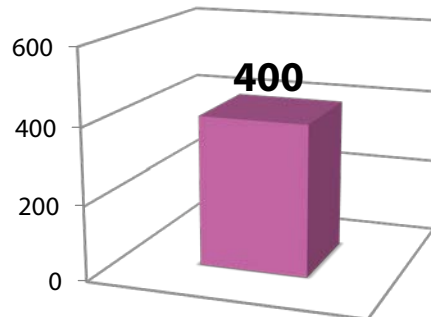
```
var queue = new MessageQueue(@".\private$\secure-queue");
queue.DefaultPropertiesToSend.UseAuthentication = true;
queue.DefaultPropertiesToSend.UseEncryption = true;
for (int i=0; i<1000; i++)
{
    queue.Send("Message: " + i);
}
```

- May fail silently

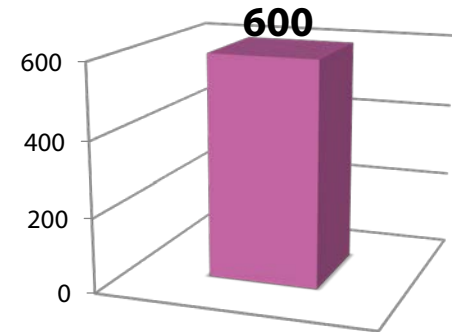
Demo 2: Using MSMQ with .NET




Default messaging
non-recoverable



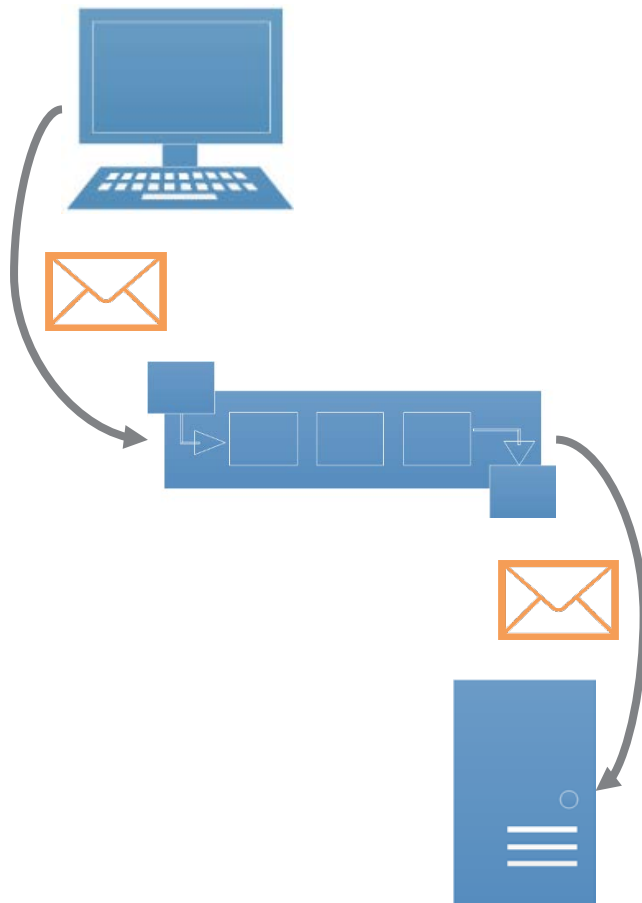
Recoverable
messages



Transactional
queues

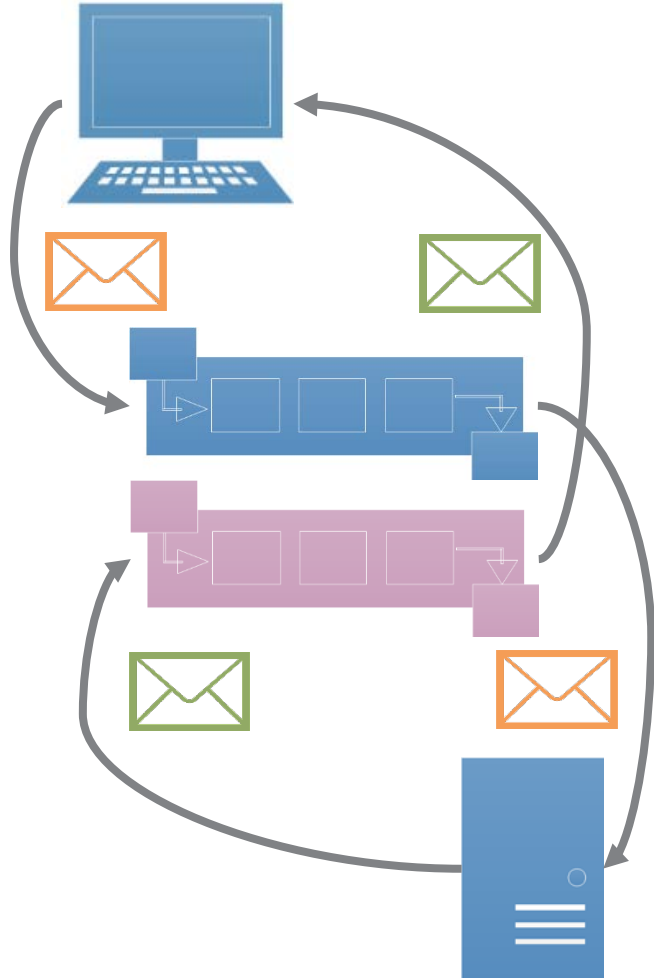
 - milliseconds to send 1,000 messages

Messaging Pattern Support



Fire-and-forget
Reliability options

Messaging Pattern Support



Fire-and-forget

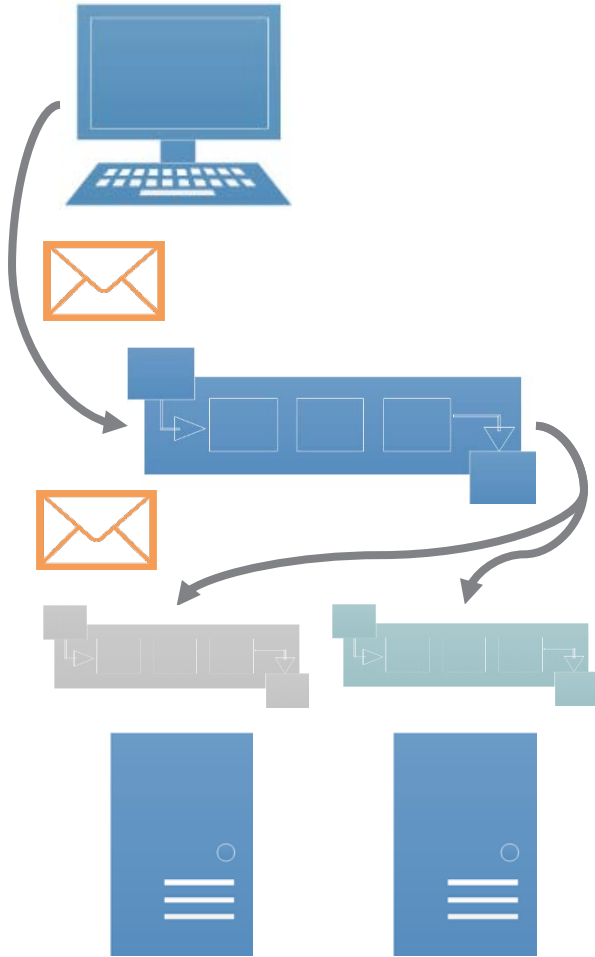
Reliability options

Request-response

Permanent reply queue

Temporary reply queue

Messaging Pattern Support



Fire-and-forget

Reliability options

Request-response

Permanent reply queue





Temporary reply queue

Publish-subscribe

Over PGM (Pragmatic Multicast)

Send to one queue, MSMQ relays

Summary

- **Introducing MSMQ** 
 - Architecture and background
- **Feature set** 
 - Active Directory integration
 - Public & private queues
 - Reliability levels
- **Deployment & administration** 
- **.NET client library** 
 - System.Messaging
 - Queue management & messaging



MSMQ