

RabbitMQ

CODE DI MESSAGGI

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Panoramica

AMQP

- Protocollo (strato applicativo) per messaggistica asincrona
- Non è un API (cf. JMS)
- Indipendente da linguaggi/implementazioni/produttori
- Standard enterprise (business automation)
- Estensibile (routing/entità definite dalle applicazioni)

RabbitMQ

- Server che implementa AMQP
- Gestisce le code di messaggi (broker)
- (scritto in Erlang, ma non avremo a che farci per ora)
- Svartati client (ogni client AMQP) in molti linguaggi

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

AMQP

Primitive:

- Messaggi
- Code

Ruoli in AMQP:

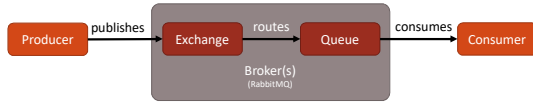
- Producers
- Consumers
- Brokers
 - ricevono, mantengono e rilasciano messaggi

Architetture di comunicazione:

- Queue
- Topic (pub/sub)
- Etc...

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Comunicazione in AMQP



I messaggi sono consegnati ai punti di scambio
Distribuiti alle code secondo le regole di routing (binding e metadati)
Recuperati dai consumatori sono sospesi dalla coda
Definitivamente rimossi solo alla ricezione di un ack

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Exchanges e routing

Gli exchanges sono caratterizzati da:

- Tipo (diretto, topic, headers...)
- Nome
- Durata
- (Argomenti specifici del broker)

Possono sopravvivere al broker

Possono essere rilasciati automaticamente (se nessuna coda li usa)

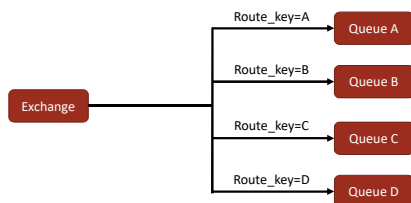
Esiste un exchange speciale (senza nome)

- È l'exchange di default
- Distribuisce i messaggi in base al nome della coda di destinazione
- Garantisce la semantica «semplice» delle code (e.g. JMS)

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Direct exchange

Instrada un messaggio a una coda in base alla sua chiave

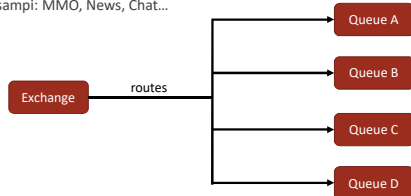


SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Fanout exchange

Instrada un messaggio a ogni coda cui è legato (ignora la chiave)

Esempi: MMO, News, Chat...

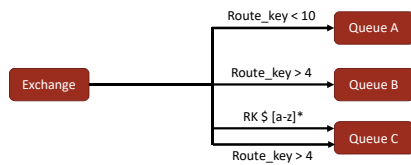


SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Topic exchange

L'inoltro è pattern-based

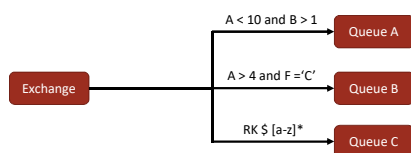
Le code sono legate all'exchange specificando un **pattern**



SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Header exchange

Simile al topic exchange, ma consente di specificare più attributi



SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Queue e consumer

Similmente agli exchange le code sono caratterizzate da:

- Nome
- Durata
- Rimozione automatica
- (Argomenti specifici dell'implementazione)

Le code durabili devono essere persistenti (e.g. su disco) e sopravvivere al broker

I messaggi possono essere:

- Recuperati attivamente dal consumatore (pull)
- Recapitati dalla coda (push)
- Nel caso push, i consumatori devono registrarsi/rimuoversi

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Ack/Nack/Rej

I messaggi recapitati sospesi dalla coda

La rimozione effettiva avviene alla ricezione dell'ACK da parte del client

- Automatico: presume ACK
- Ricezione: ACK alla ricezione del messaggio
- Check-point: ACK al salvataggio su storage persistente
- Completamento: ACK al completamento dell'elaborazione

In caso di timeout/fallimento il messaggio sospeso è riaccodato

Rifiuto uno (REJ) o più (NACK) messaggi

- E.g. errori interni al client/messaggio non atteso
- Richiede al broker di riaccodare o scartare

Supporto al prefetch di gruppi di messaggi

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Esempio



Setup della coda:

```
ConnectionFactory factory = new ConnectionFactory();
factory.setHost(hostName);
Connection connection = factory.newConnection();
Channel channel = connection.createChannel();
channel.queueDeclare(queueName,
    false, false, false, null);
```

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Esempio

Produttore:

```
String message = "Hello World!";  
channel.basicPublish("", QUEUE_NAME,  
    null, message.getBytes());
```

Consumatore:

```
QueueingConsumer consumer =  
    new QueueingConsumer(channel);  
channel.basicConsume(QUEUE_NAME, true, consumer);  
while (true) {  
    QueueingConsumer.Delivery delivery =  
        consumer.nextDelivery();  
    String message = new String(delivery.getBody());  
    ...  
}
```

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI

Esempi (allegati)

1. Pool di consumer
2. Servizio di log
 1. Fanout
 - Broadcast a tutti i sottoscrittori
 - Sottoscrizioni persistenti
 2. Direct
 - Log instradati per gravità (info, warning, error)
 3. Topic
 - Log filtrati in base a gravità ed evento (pattern based)
3. RPC
 - Code temporanee per recuperare il risultati
 - Sync/Async

SISTEMI DISTRIBUITI - M. MICULAN E M. PERESSOTTI
