

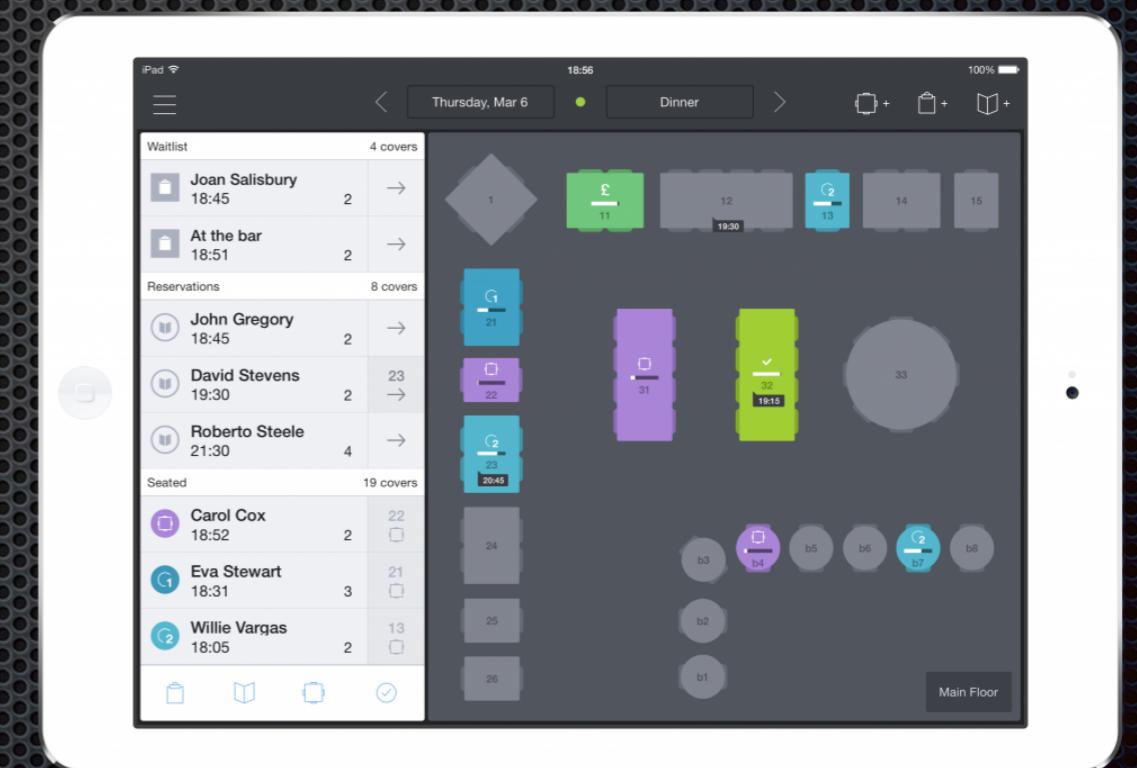
RabbitMQ for modern microservices

Łukasz Łysik, OpenTable
TWF Conference, 23. Sep 2016

OpenTable

- Real-time restaurant reservations.

The screenshot shows the OpenTable website interface. At the top, there's a header with the OpenTable logo, location "San Francisco Bay Area", user "Hi, Matteo", and a "Mobile" dropdown. Below the header is a large photo of a group of people at a restaurant. The main content area has a heading "SAN FRANCISCO RESTAURANTS Make a free reservation." with dropdowns for "2 people", "May 20, 2015", "7:00 PM", and a search bar "Location, Restaurant, or Cuisine". A red "Find a Table" button is prominent. Below this, a section titled "Treat your taste buds: Explore San Francisco restaurants." lists "MOTHER'S DAY", "BEST RESTAURANTS", "NEW RESTAURANTS", and "1,000-POINT TABLES". It features sections for "San Francisco Mother's Day Brunch, Lunch & Dinner Specials" and two restaurant reviews: "NEPTUNE's Waterfront Grill & Bar" and "Castagna - San Francisco".



@lukasz_lysic / llysic@gmail.com

Transaction Messaging System

- Sending reservation-related messages
 - Confirmation, reminder, etc.
 - Email, SMS, etc.
- General purpose messaging system.

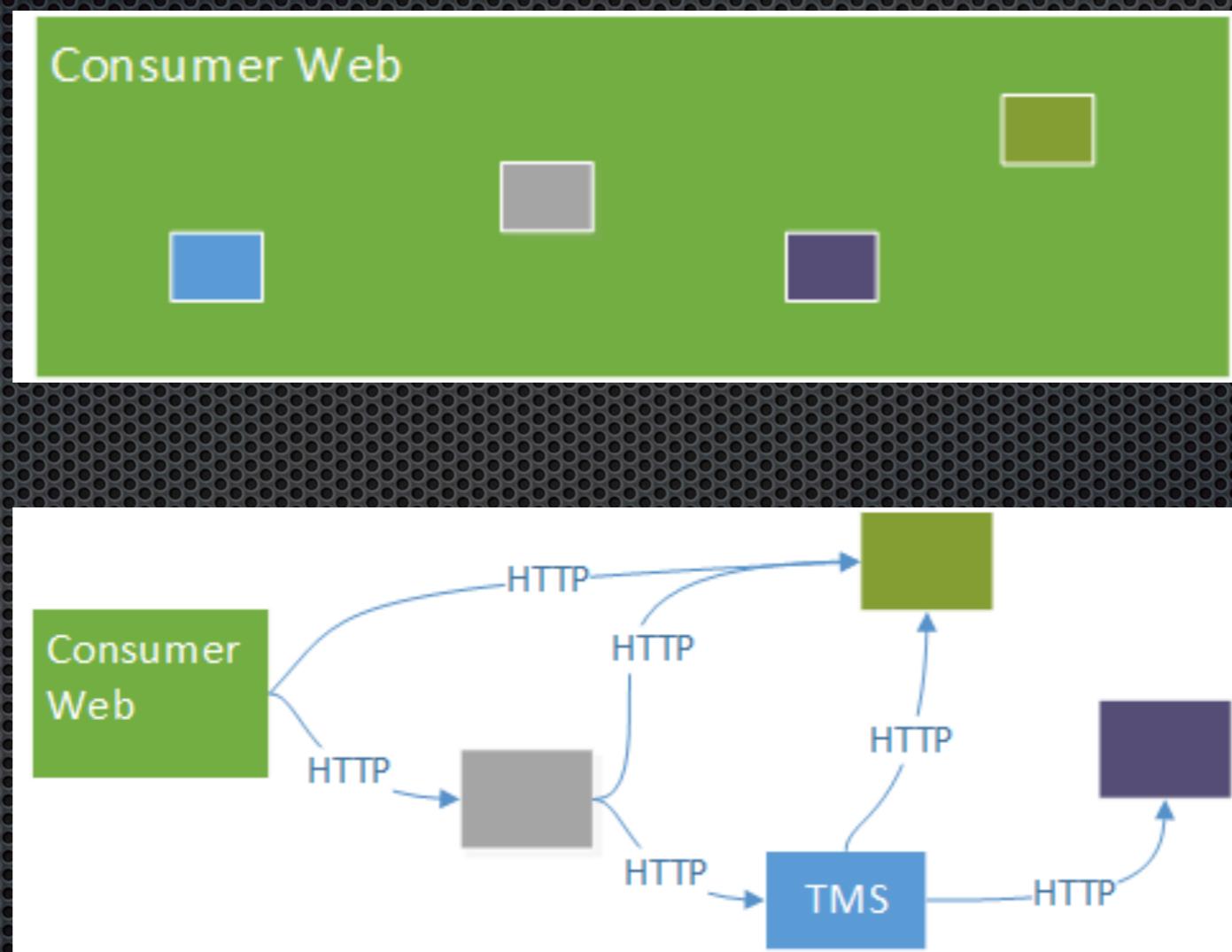
RabbitMQ



- Message Broker
- Open Source
- Implements AMQP standard (Advanced Messaging Queueing Protocol)
- Written in Erlang

Erlang

Transaction Messaging System



@lukasz_lysic / llysic@gmail.com

Transaction Messaging System

- Three steps:
 - Request validation (acceptance)
 - Artefacts (email, SMS, etc.) composition
 - Artefacts sending

Transaction Messaging System



Transaction Messaging System



1. Good API design.
2. Publishing message
3. Consuming message
4. Proper decoupling.
5. Dealing with failures.

Problem 1 - Good API design



Problem 1 - bad API design

Request

```
POST /v1/reservation-confirmation
{
  dinerEmail: "test@opentable.com",
  restaurantId: 1234,
  confirmationNumber: 5678
}
```

Response

```
StatusCode: 200
{
  status: "Email sent"
}
```

```
StatusCode: 500
{
  status: "Error",
  details: "Can't ..."
}
```



@lukasz_lysic / llysic@gmail.com

Problem 1 - good API design

Request

```
POST /v1/reservation-confirmation
{
  dinerEmail: "test@opentable.com",
  restaurantId: 1234,
  confirmationNumber: 5678
}
```



Response

```
StatusCode: 201
{
  href: "http://tms/v1/tickets/123-456-789",
  status: "Accepted"
}
```



@lukasz_lysic / llysic@gmail.com

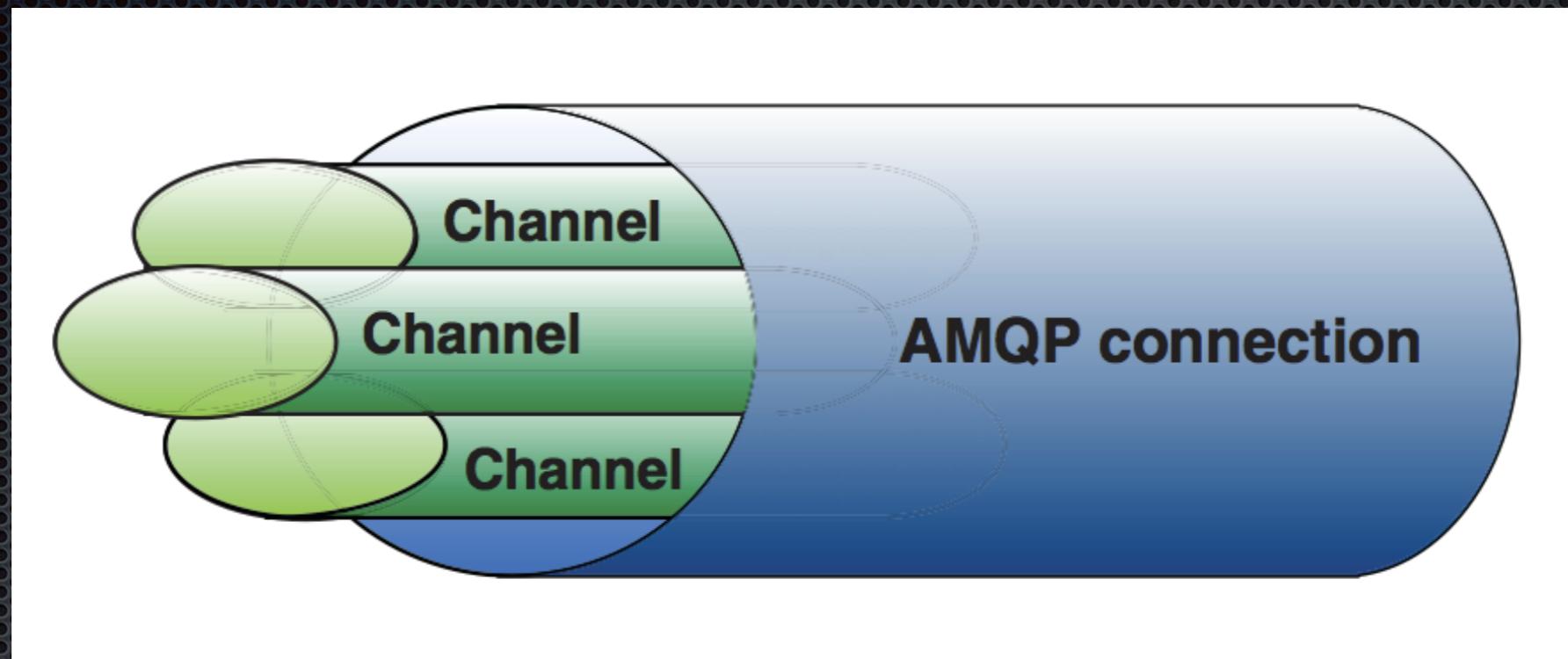
Problem 2 - Publishing message



Problem 2 - publishing message

- Message format.
- Message size.
- Connection management.
- Client libraries.

Problem 2 - connection



Source: "RabbitMQ in Action", Jason J. W. Williams, Alvaro Videla

```
var factory = new ConnectionFactory { HostName = "localhost" };
var connection = factory.CreateConnection(); // Thread safe
var channel = connection.CreateModel(); // Thread unsafe
```



@lukasz_lysic / llysic@gmail.com

Problem 2 - sending message

```
var message = "Hello World!";
var body = Encoding.UTF8.GetBytes(message);
```

```
var properties = channel.CreateBasicProperties();
properties.Persistent(true);
properties.ContentType = "application/vnd.ticket_changed.v1+json";
properties.Headers = new Dictionary<string, object>();
properties.Headers.Add("anything", "here");
```

```
channel.BasicPublish(
    exchange: "hello",
    routingKey: "",
    mandatory: false,
    basicProperties: properties,
    body: body);
```



@lukasz_lysic / llysic@gmail.com

Problem 3 - Receiving message



Problem 3 - receiving message

- Competing consumers.
- Processing errors.

Problem 3 - receiving message

```
var consumer = new QueueingBasicConsumer(channel);
channel.BasicConsume(queue: "hello", noAck: true, consumer: consumer);
```

```
while (true)
{
    var eventArgs = consumer.Queue.Dequeue();
    var message = Encoding.UTF8.GetString(eventArgs.Body);
    Console.WriteLine("Received: {0}", message);
}
```



@lukasz_lysic / llisyk@gmail.com

Problem 3 - competing consumers

```
C:\Code\RabbitMqBrownBag\Bin\HelloWorld.Producer.exe
Press ENTER to publish...
Published: 'Hello World' 1
Published: 'Hello World' 2
Published: 'Hello World' 3
Published: 'Hello World' 4
Published: 'Hello World' 5
Published: 'Hello World' 6
Published: 'Hello World' 7
Published: 'Hello World' 8
Published: 'Hello World' 9
```

```
C:\Code\RabbitMqBrownBag\Bin\HelloWorld.Consumer.exe
Received: Hello World! 1
Received: Hello World! 3
Received: Hello World! 5
Received: Hello World! 7
Received: Hello World! 9
```

```
C:\Code\RabbitMqBrownBag\Bin\HelloWorld.Consumer.exe
Received: Hello World! 2
Received: Hello World! 4
Received: Hello World! 6
Received: Hello World! 8
```



@lukasz_lysic / llysic@gmail.com

Problem 3 - manual ack

```
// noAck - "no manual acks"  
channel.BasicConsume(queue: "hello", noAck: false, consumer: consumer);
```

```
// Process message  
channel.BasicAck(eventArgs.DeliveryTag, multiple: false);
```

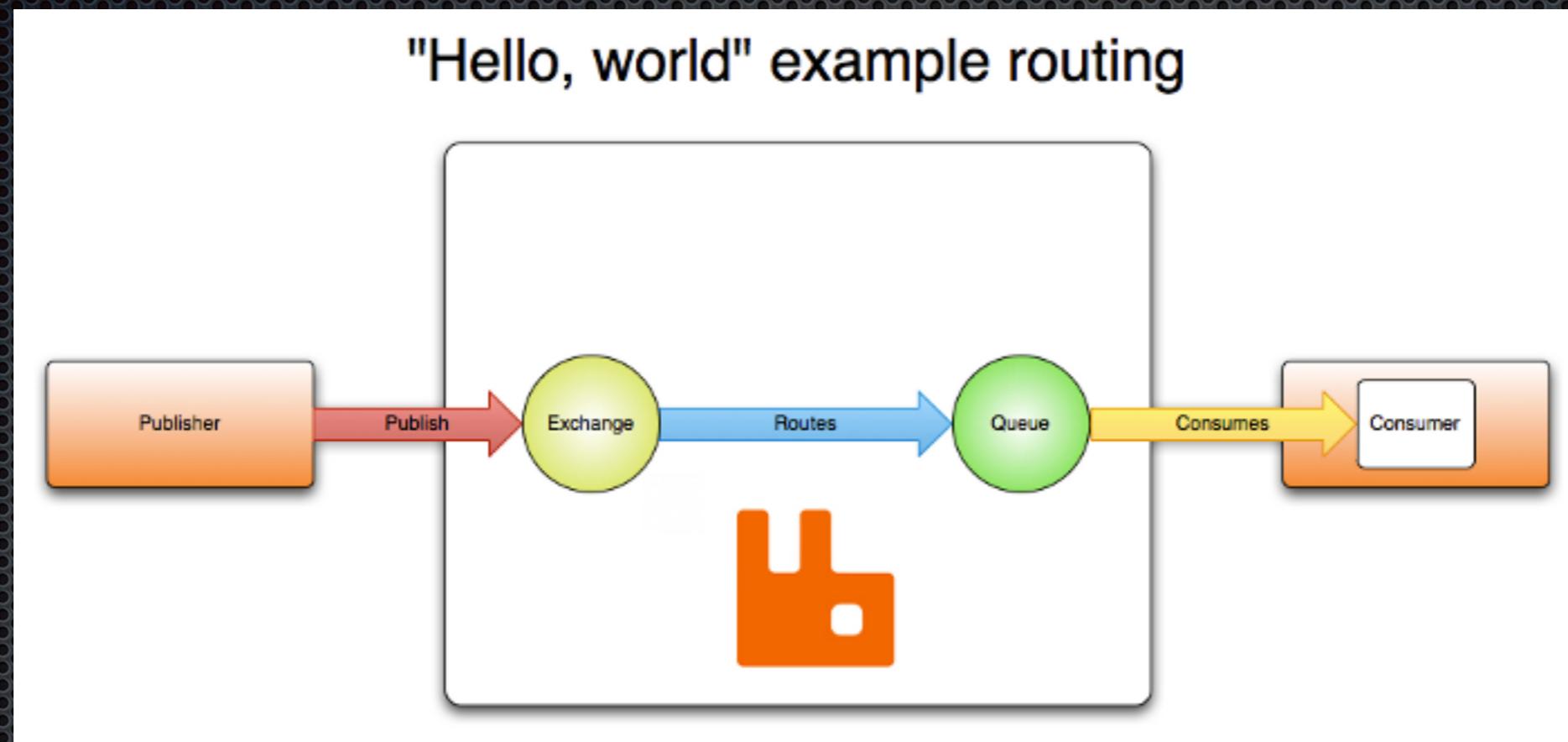


@lukasz_lysic / llisyk@gmail.com

Problem 4 - Proper decoupling



Problem 4 - routing



Source: <https://www.rabbitmq.com/tutorials/amqp-concepts.html>



@lukasz_lysic / llysic@gmail.com

Problem 4 - topic routing

USA.Sport.Video

UK.Sport.Audio

CA.Politics.Video

Ex1

USA.*.*

Queue-USA

.Sport.

Queue-Sport

..Audio

Queue-Audio

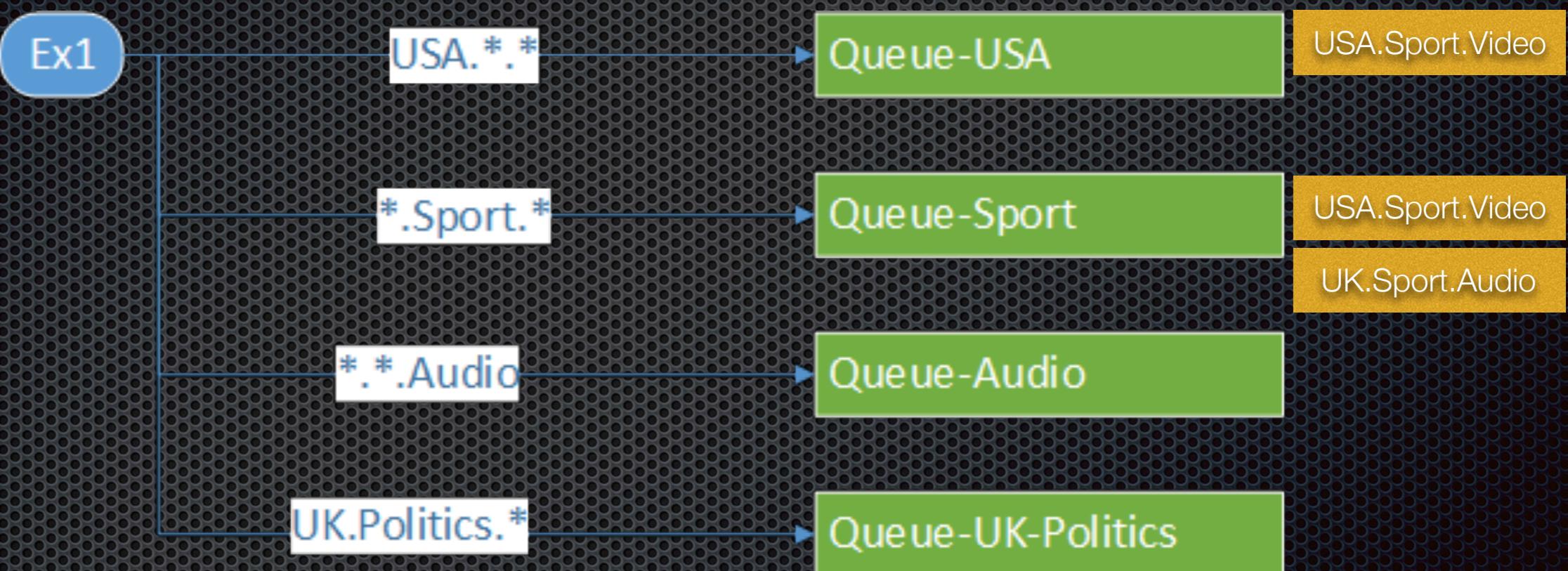
UK.Politics.*

Queue-UK-Politics



@lukasz_lysic / llnsic@gmail.com

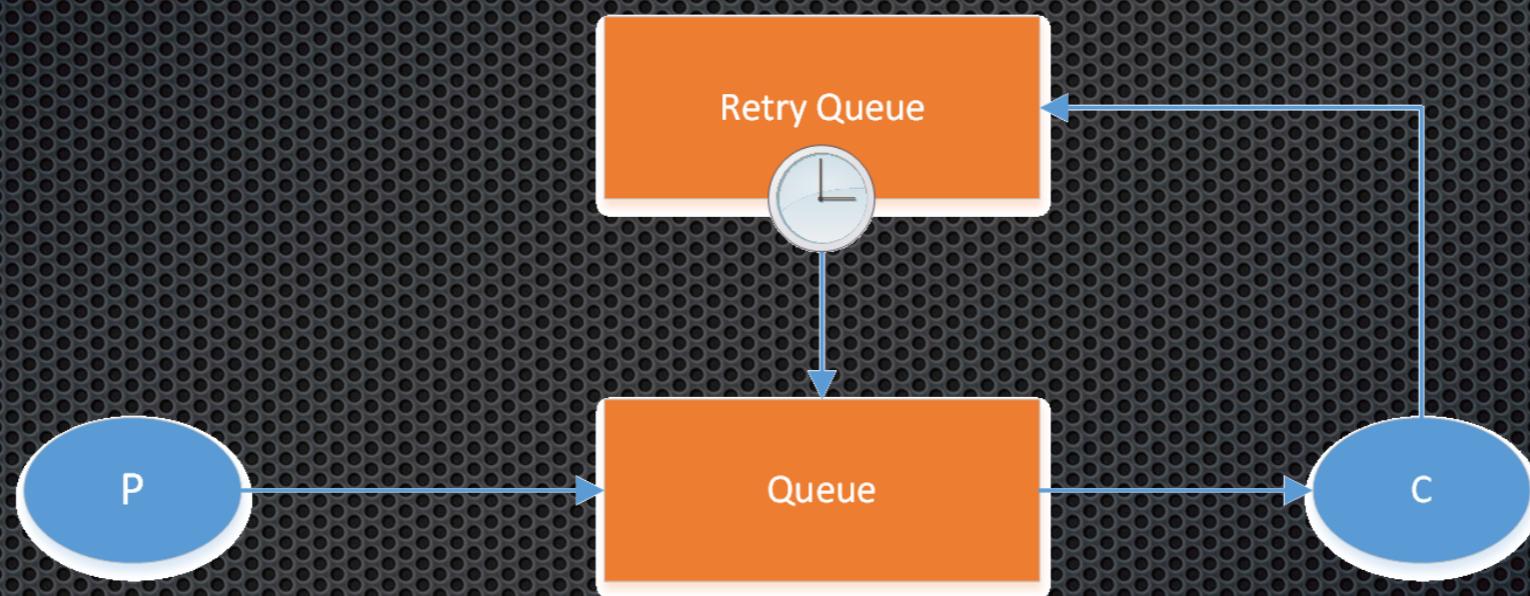
Problem 4 - topic routing



Problem 5- Dealing with failures

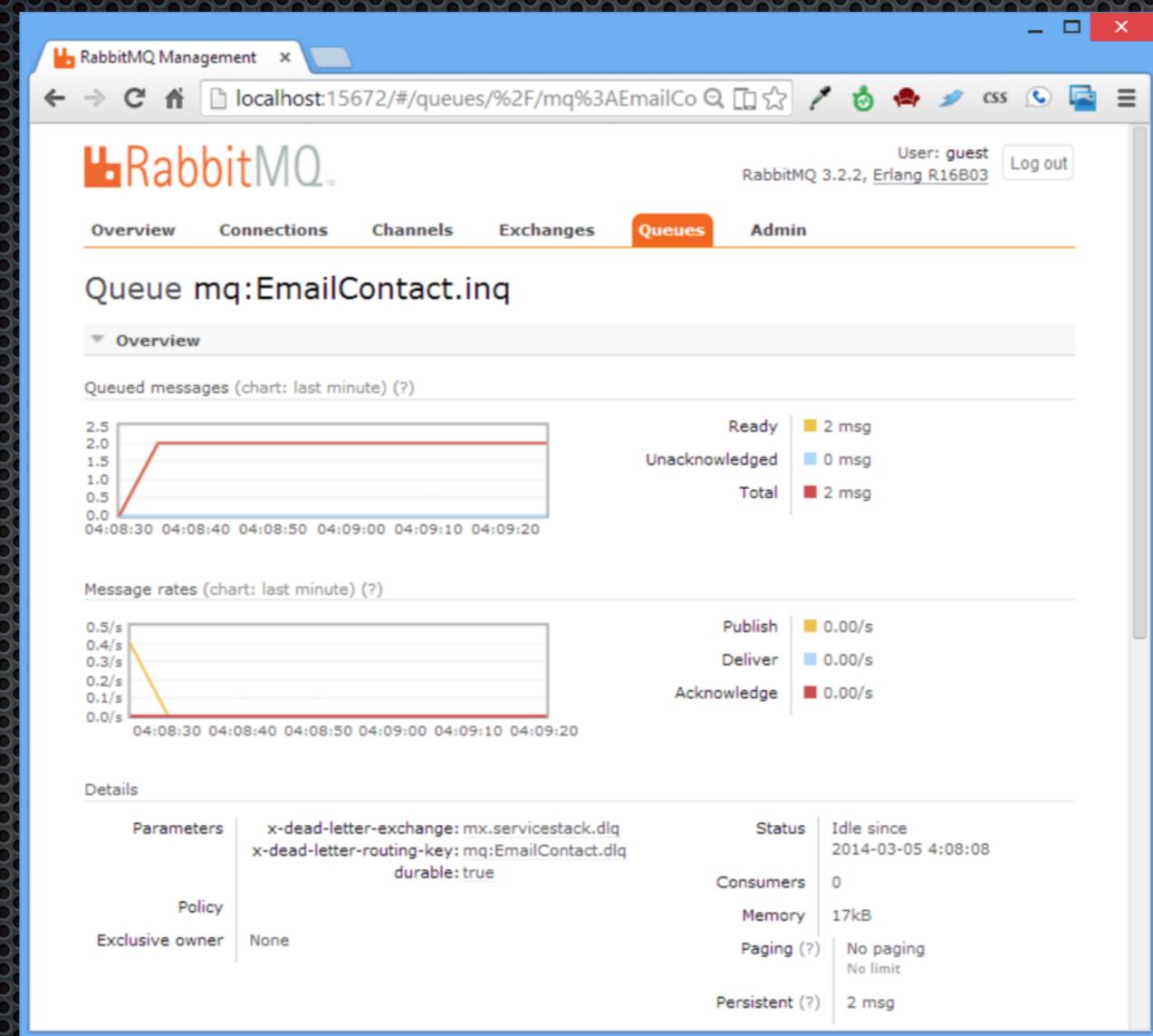


Problem 5 - Time-To-Live extensions and dead-letter exchange



Other useful features

Management plugin



@lukasz_lysic / llysic@gmail.com

HTTP API

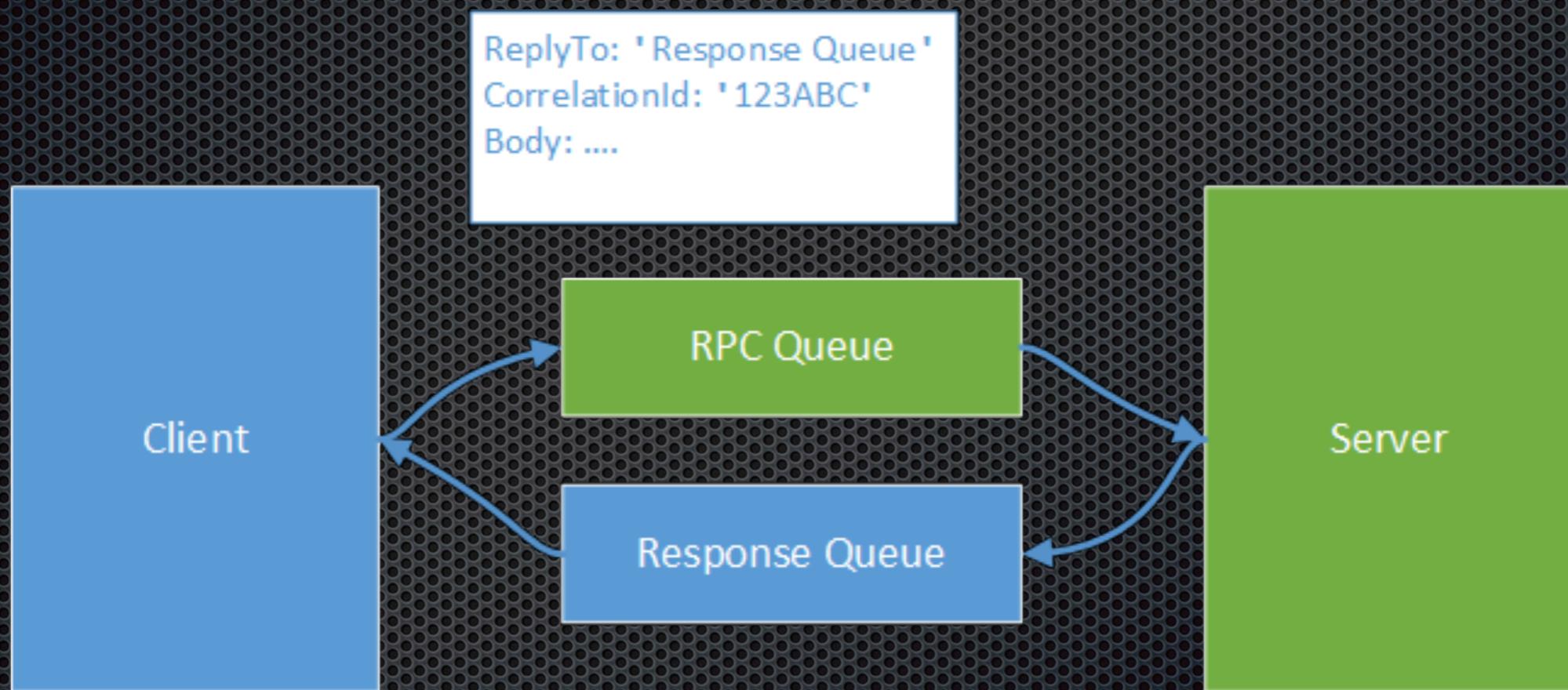
Reference

GET	PUT	DELETE	POST	Path
X				<u>/api/overview</u>
X				<u>/api/connections</u>
X		X		<u>/api/connections/name</u>
X				<u>/api/channels</u>
X				<u>/api/channels/channel</u>
X				<u>/api/exchanges</u>
X				<u>/api/exchanges/vhost</u>
X	X	X		<u>/api/exchanges/vhost/name</u>
X				<u>/api/exchanges/vhost/name/bindings</u>
X				<u>/api/queues</u>
X				<u>/api/queues/vhost</u>
X	X	X		<u>/api/queues/vhost/name</u>



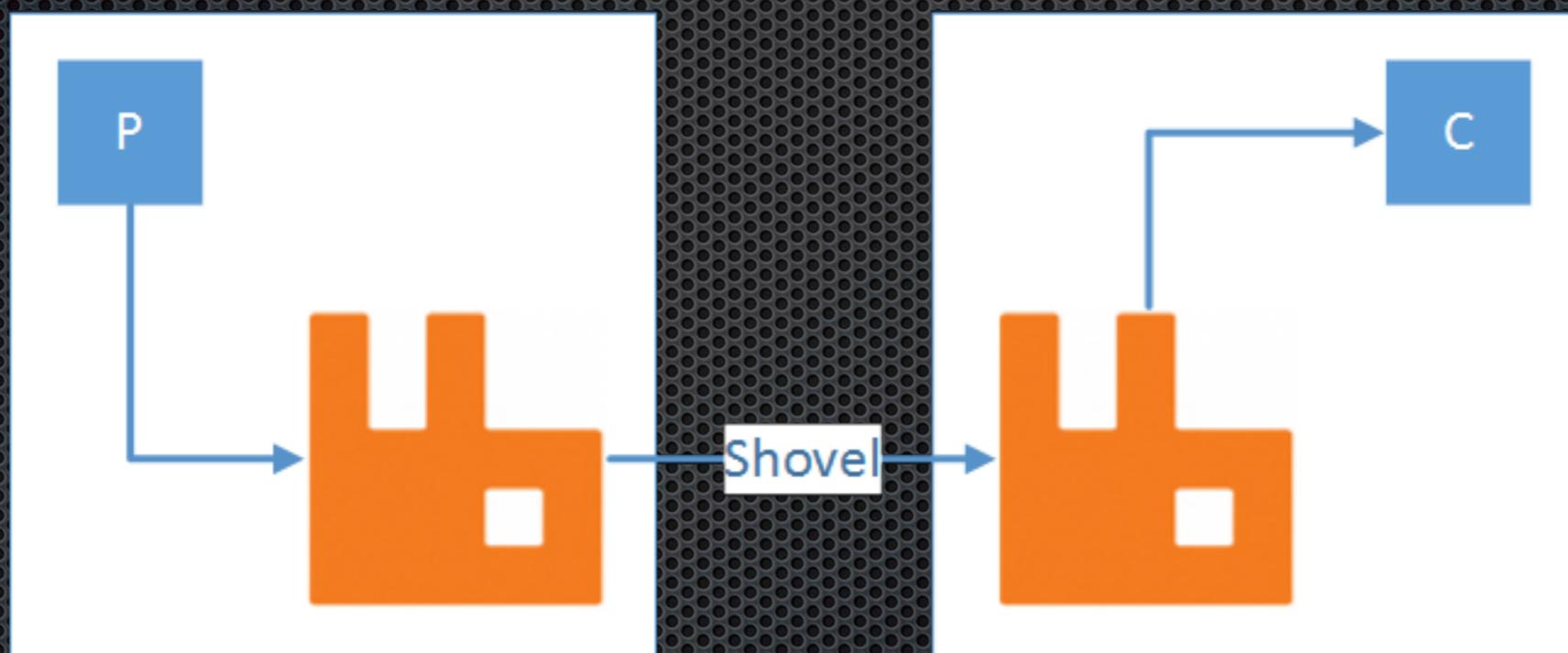
@lukasz_lysic / llysic@gmail.com

Remote Procedures Call (RPC)



@lukasz_lysic / llysic@gmail.com

Shovel plugin and federation



@lukasz_lysic / llysic@gmail.com

Questions?



@lukasz_lysic / llnsic@gmail.com