

# Hands-on with Message Queues

**Workshop**

Build and Connect Microservices with AMQP



github.com/cloudamqp/lavinmq

Platform Solutions Resources Open Source Enterprise Pricing

Search Sign in Sign up

cloudamqp / lavinq Public

Notifications Fork 45 Star 782

Code Issues 69 Pull requests 32 Discussions Actions Security Insights

main Go to file Code

 **bengtmagnus** Sidebar cleanup and ic... 18931f9 · 10 hours ago 4,988 Commits

|            |                                             |              |
|------------|---------------------------------------------|--------------|
| .github    | Make code review prompt more con...         | 4 days ago   |
| extras     | Cleanup root dir (#1060)                    | 5 months ago |
| openapi    | Use Stoplight Elements to render Op...      | last year    |
| packaging  | RPM: enable CRB repo for el-9 build...      | 5 days ago   |
| spec       | Prevent delayed exchanges to bind t...      | 4 days ago   |
| src        | Prevent delayed exchanges to bind t...      | 4 days ago   |
| static     | Sidebar cleanup and icons added (#...       | 10 hours ago |
| views      | Sidebar cleanup and icons added (#...       | 10 hours ago |
| .ameba.yml | Add possibility to control time to fix t... | 3 months ago |
| .gitignore | UI updates (#821)                           | 9 months ago |

About

Ultra quick message queue and streaming server

[lavinmq.com](#)

mqtt streaming server amqp background-jobs message-queue job-queue message-broker persistent-logs

Readme Apache-2.0 license Contributing Security policy Activity Custom properties 782 stars 19 watching



### SHARED INSTANCES

For development or small hobby projects. Not recommended for production due to variable performance.



#### Elegant Ermine

- Max 2000 queues
- Max 200 000 queued messages

**20 M**

Max MSGs/month

**200**

Max connections

**\$ 19**

PER MONTH

[Get Now](#)



#### Loyal Lemming

- Max 200 queues
- Max 20 000 queued messages
- Max idle queue time 28 days

**2 M**

Max MSGs/month

**40**

Max connections

**FREE**

[Get Now](#)

Patrik Ragnarsson

Area Lead, 84codes

@dentarg



Anders Bälter

CTO, 84codes

@baelter

## Agenda

- Introduction
- Why Message Queues?
- AMQP Theory
- Workshop walk-through
- Let's code together!
- Round off with demo
- Finish

Menti



7515 0571

# Why Message Queues?

# Decoupling

# Scalability

# Asynchronous Processing

# Resilience

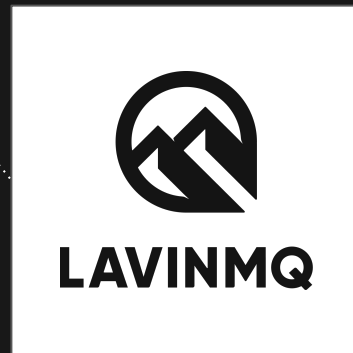
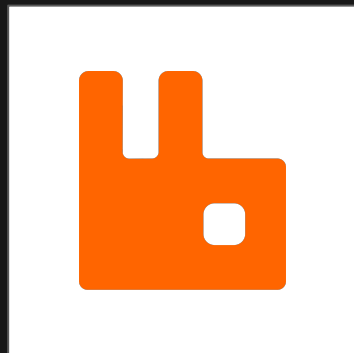




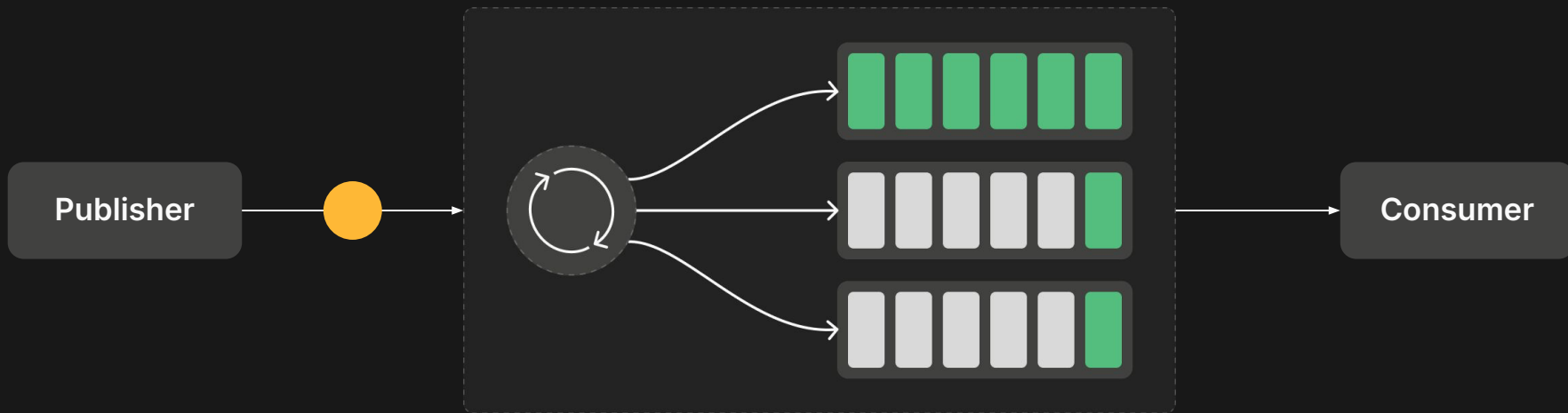
AMQP



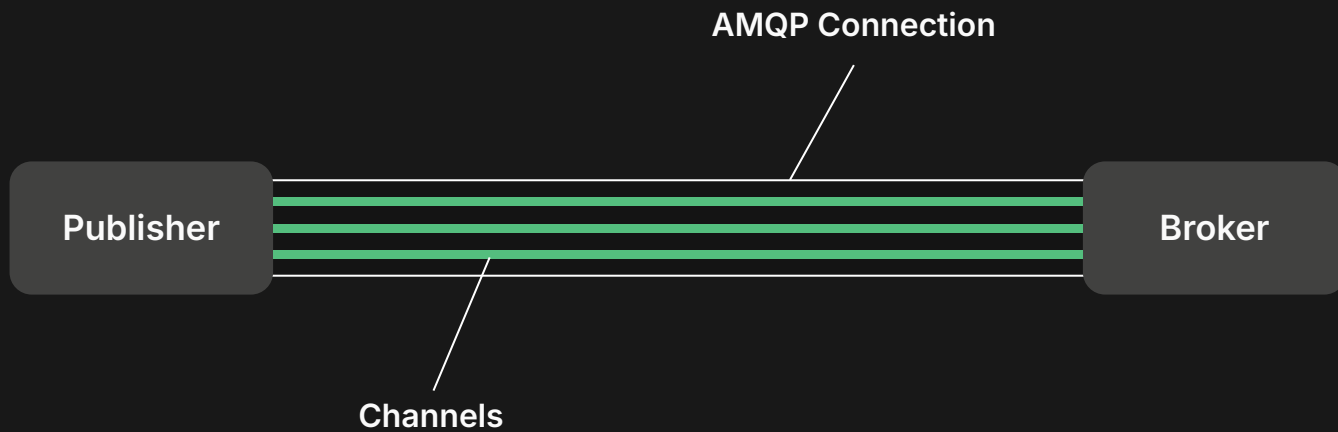
AMQP



## Concepts overview



## Connections & channels



## Messages

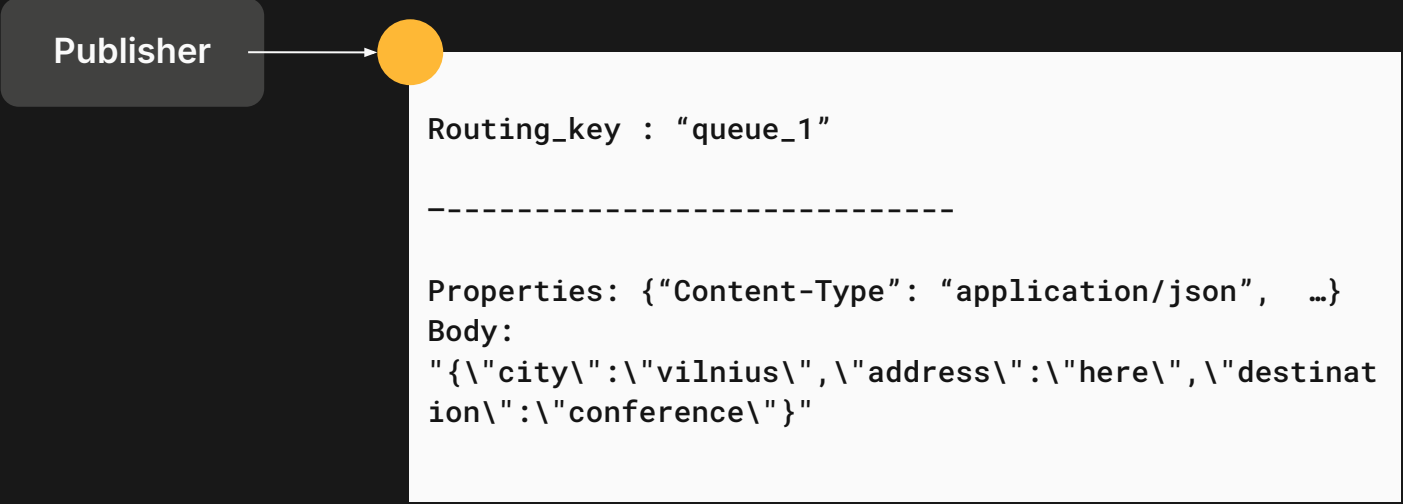


Properties: {...}

Body:

## Messages

Publisher



Routing\_key : "queue\_1"

-----

Properties: {"Content-Type": "application/json", ...}

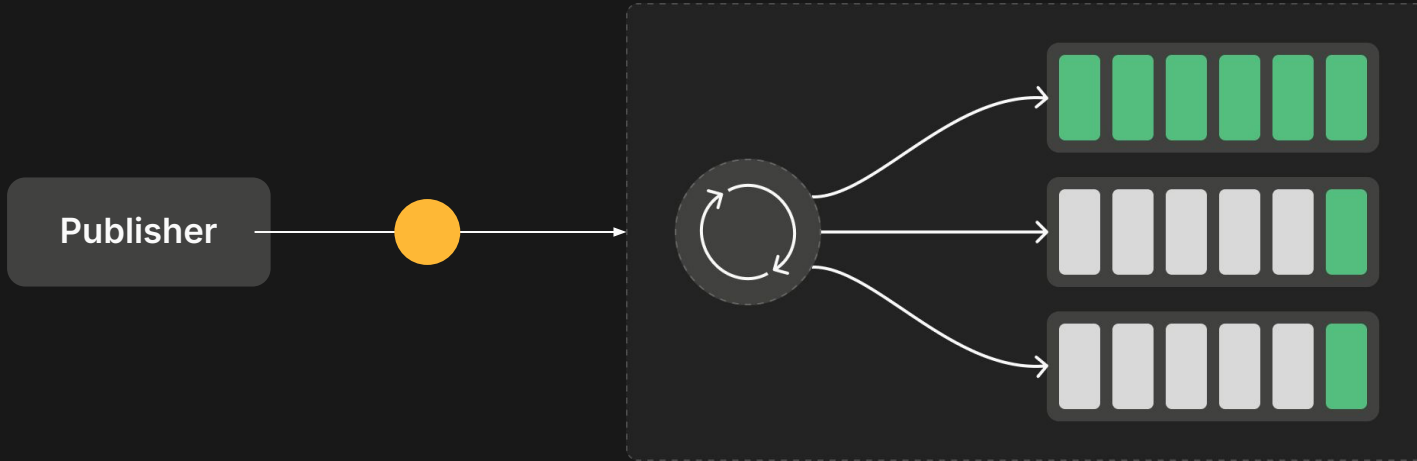
Body:

```
{"city": "vilnius", "address": "here", "destination": "conference"}
```

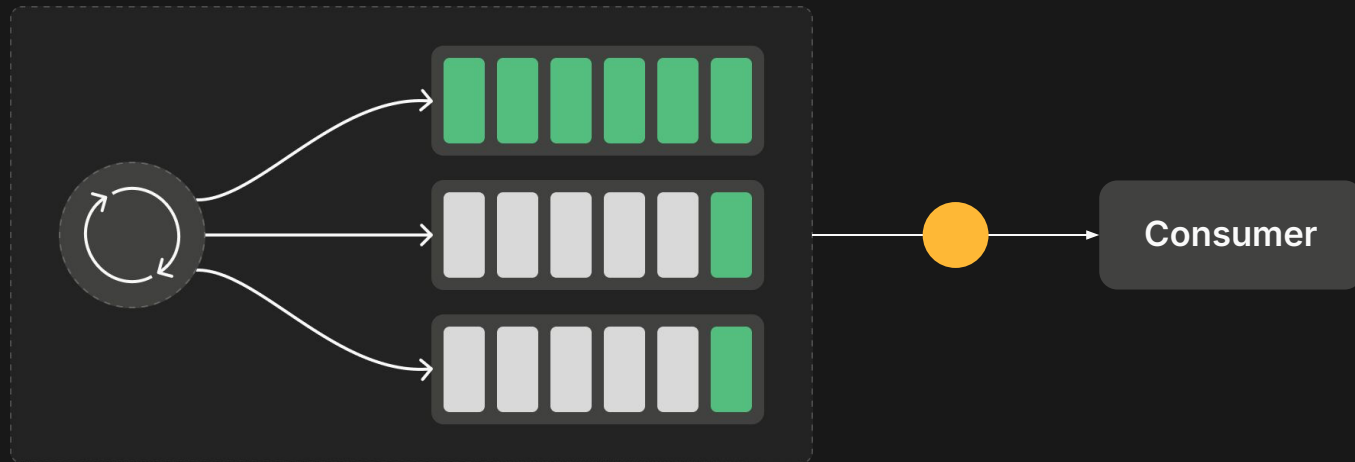
## Routing



## Routing



## Consumers



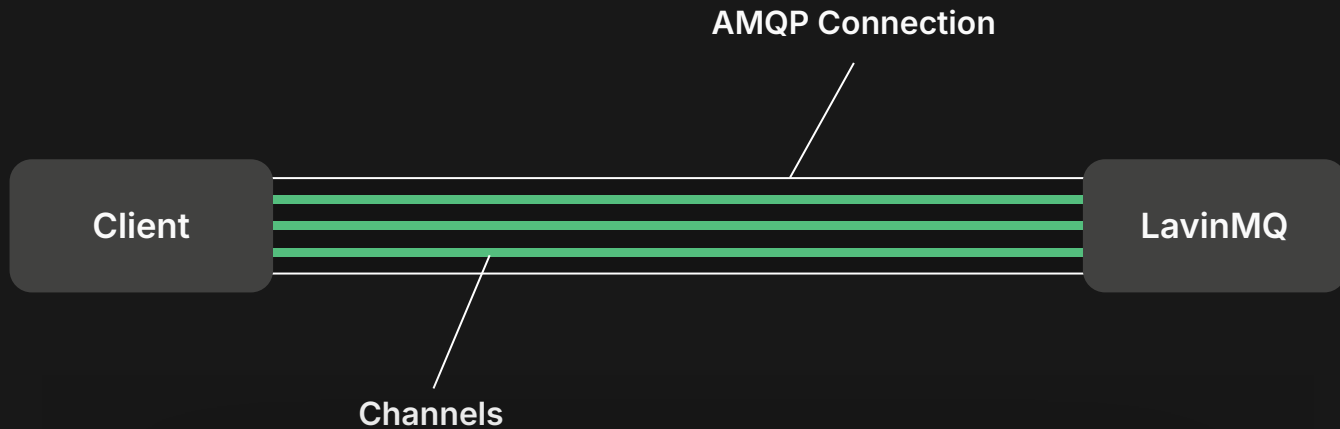
## Getting started with your client



euroku-workshop - consumer.rb

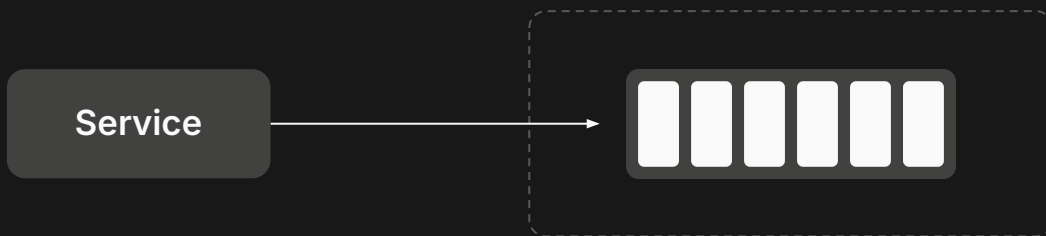
```
1 require "amqp-client"  
2  
3 amqp_url = "amqp://guest:guest@localhost:5672"
```

## Setting up connections and channels



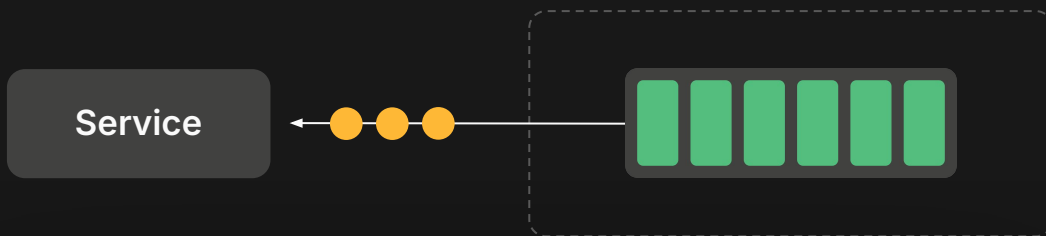
```
1 require "amqp-client"  
2  
3 amqp_url = "amqp://guest:guest@localhost:5672"  
4  
5 client = AMQP::Client.new(amqp_url).start
```

## Declaring a queue



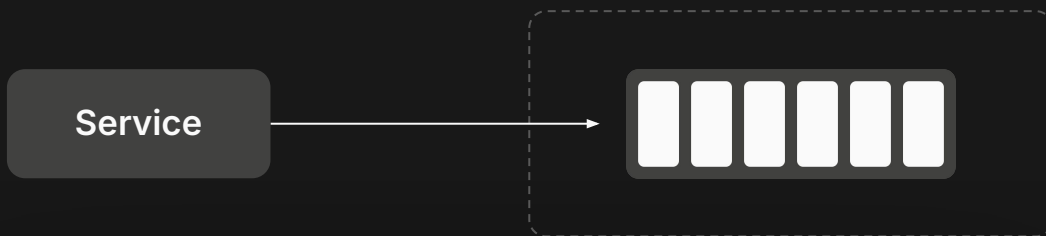
```
1 require "amqp-client"
2
3 amqp_url = "amqp://guest:guest@localhost:5672"
4
5 client = AMQP::Client.new(amqp_url).start
6 queue = client.queue("myQueue")
```

## Declaring a queue



```
1 require "amqp-client"
2
3 amqp_url = "amqp://guest:guest@localhost:5672"
4
5 client = AMQP::Client.new(amqp_url).start
6 queue = client.queue("myQueue")
7
8 queue.subscribe do |message|
9   puts "Received message: #{message.body}"
10 end
11
12 sleep
```

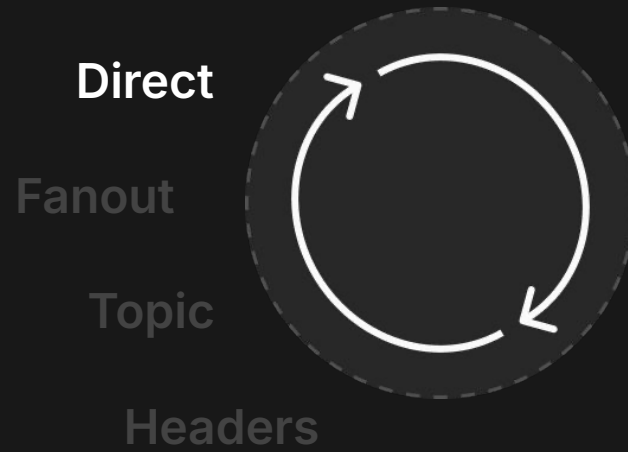
Create a message



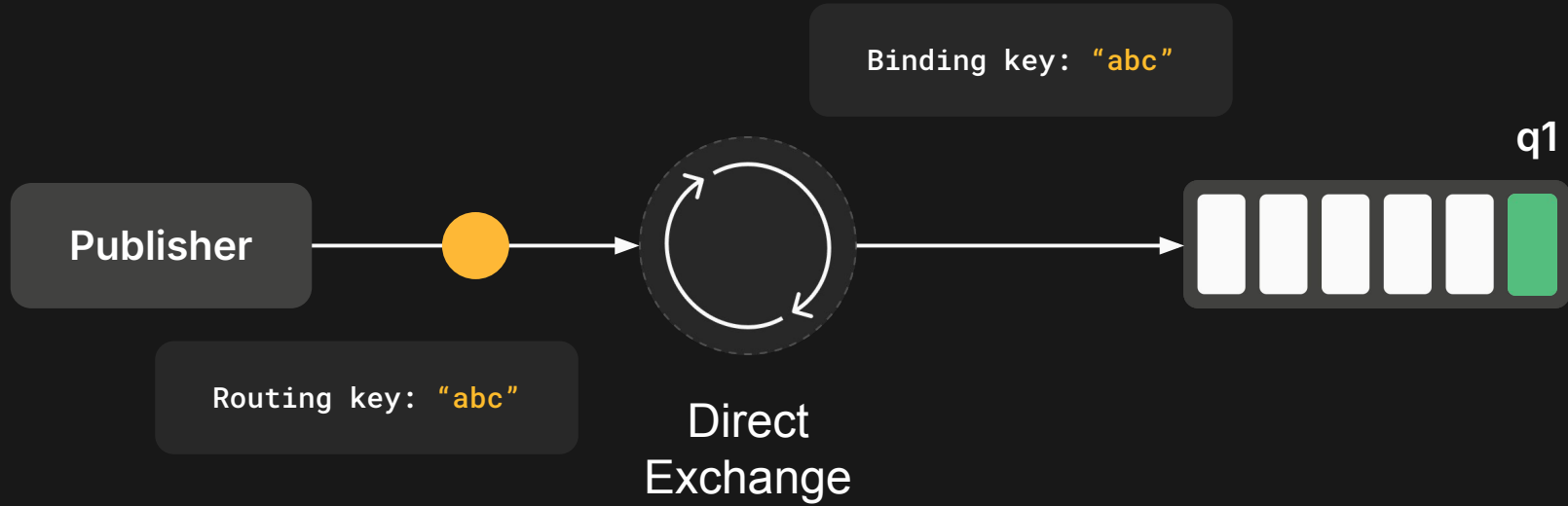
```
1 require "amqp-client"
2
3 amqp_url = "amqp://guest:guest@localhost:5672"
4
5 client = AMQP::Client.new(amqp_url).start
6 queue = client.queue("myQueue")
7
8 queue.subscribe do |message|
9   puts "Received message: #{message.body}"
10 end
11
12 sleep
```

# AMQP Exchanges

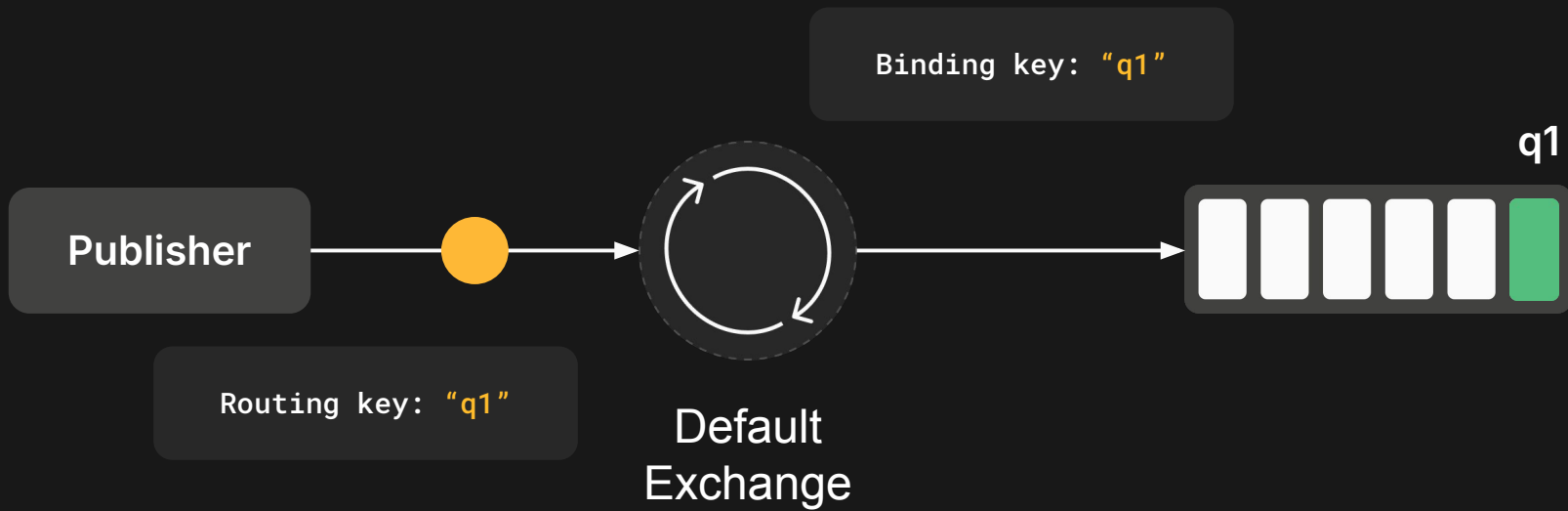
## Exchange types



## Direct Exchange



## Default Exchange



Establishing connection

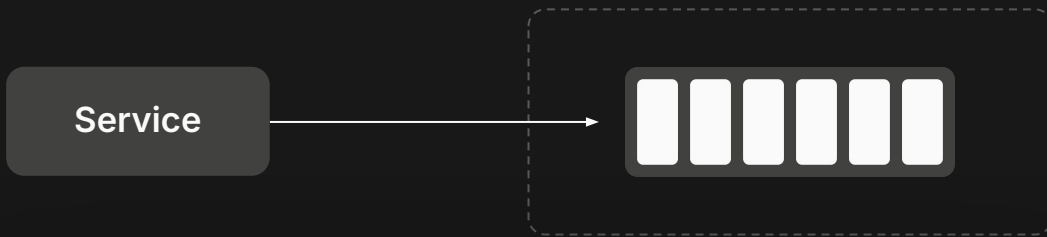
Service



```
1 require "amqp-client"  
2  
3 amqp_url = "amqp://guest:guest@localhost:5672"  
4  
5 client = AMQP::Client.new(amqp_url).start
```



## Declaring a queue



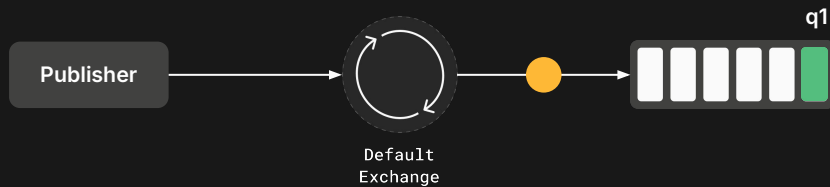
```
1  require "amqp-client"
2
3  amqp_url = "amqp://guest:guest@localhost:5672"
4
5  client = AMQP::Client.new(amqp_url).start
6
7  q1 = client.queue("q1")
```

## Default Exchange



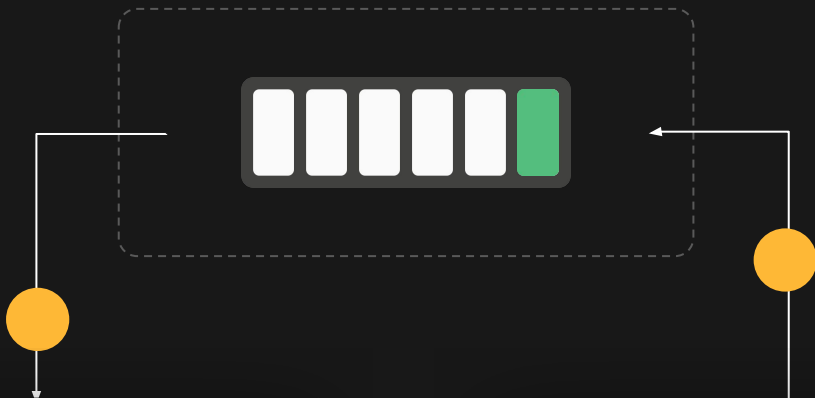
```
1  require "amqp-client"
2
3  amqp_url = "amqp://guest:guest@localhost:5672"
4
5  client = AMQP::Client.new(amqp_url).start
6
7  client.queue("q1")
8  ex = client.default
9  ex.publish("Hello Everyone!", "q1")
10
```

## Default Exchange



```
1 require "amqp-client"
2
3 amqp_url = "amqp://guest:guest@localhost:5672"
4
5 client = AMQP::Client.new(amqp_url).start
6
7 client.queue("q1")
8   .publish("Hello Everyone!")
9
```

## Publishing and consuming



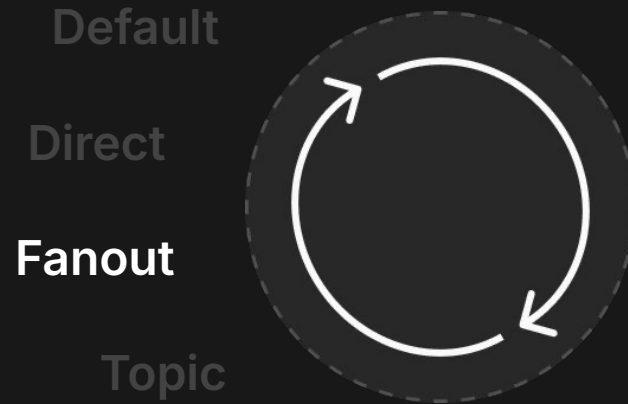
```
● → euroku-workshop ruby publisher.rb  
○ → euroku-workshop
```

```
○ → euroku-workshop ruby consumer.rb  
Received message: Hello World!
```

```
1 require "amqp-client"  
2  
3 amqp_url = "amqp://guest:guest@localhost:5672"  
4  
5 client = AMQP::Client.new(amqp_url).start  
6 queue = client.queue("myQueue")  
7  
8 queue.subscribe do |message|  
9   puts "Received message: #{message.body}"  
10 end  
11  
12 sleep
```

```
1 require "amqp-client"  
2  
3 amqp_url = "amqp://guest:guest@localhost:5672"  
4  
5 client = AMQP::Client.new(amqp_url).start  
6 queue = client.queue("myQueue")  
7  
8 queue.publish("Hello World!")  
9
```

## Exchange types

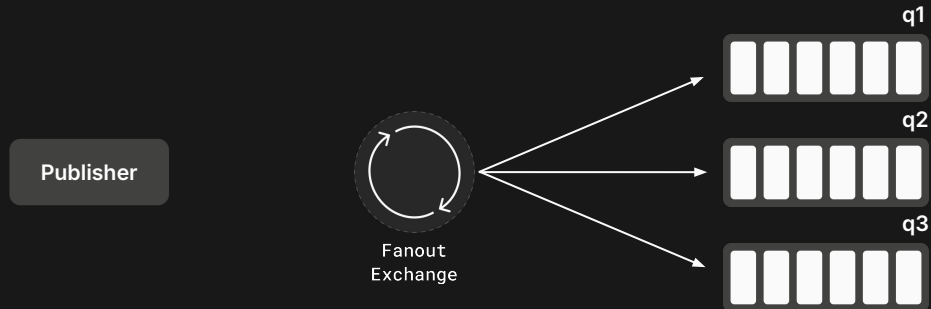


## Fanout Exchange



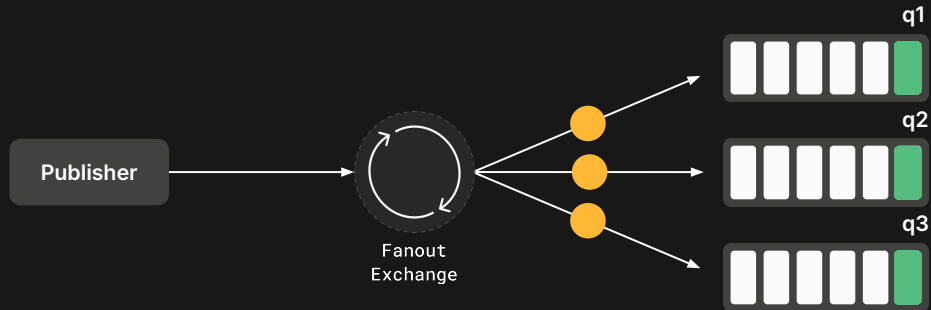
```
1  require "amqp-client"
2
3  amqp_url = "amqp://guest:guest@localhost:5672"
4
5  client = AMQP::Client.new(amqp_url).start
6
7  q1 = client.queue("q1")
8  q2 = client.queue("q2")
9  q3 = client.queue("q3")
10 ex = client.fanout
```

## Fanout Exchange



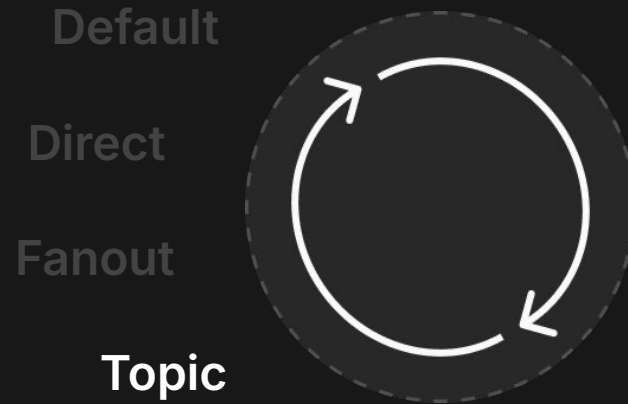
```
1  require "amqp-client"
2
3  amqp_url = "amqp://guest:guest@localhost:5672"
4
5  client = AMQP::Client.new(amqp_url).start
6
7  q1 = client.queue("q1")
8  q2 = client.queue("q2")
9  q3 = client.queue("q3")
10 ex = client.fanout
11 q1.bind(ex)
12 q2.bind(ex)
13 q3.bind(ex)
```

## Fanout Exchange



```
7  q1 = client.queue("q1")
8  q2 = client.queue("q2")
9  q3 = client.queue("q3")
10 ex = client.fanout
11 q1.bind(ex)
12 q2.bind(ex)
13 q3.bind(ex)
14 ex.publish("Hello Everyone!")
```

## Exchange types



## Topic Exchange



```
7  q1 = client.queue("q1")
8  q2 = client.queue("q2")
9  q3 = client.queue("q3")
10 ex = client.topic
```



```
binding_key = 'a.*.c'
```

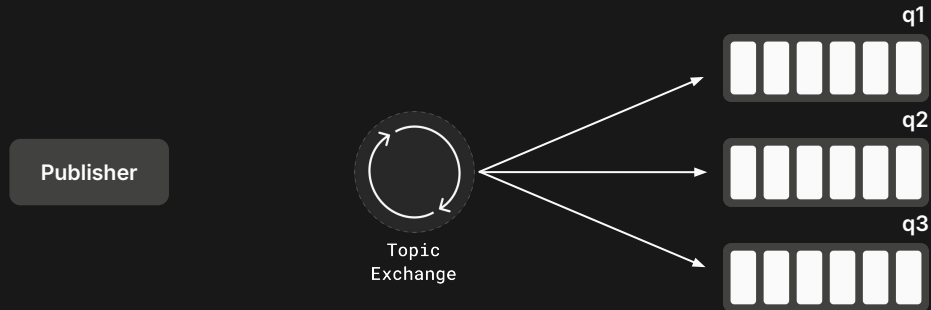
*\* (star) can substitute for exactly one word*



```
binding_key = 'a.##'
```

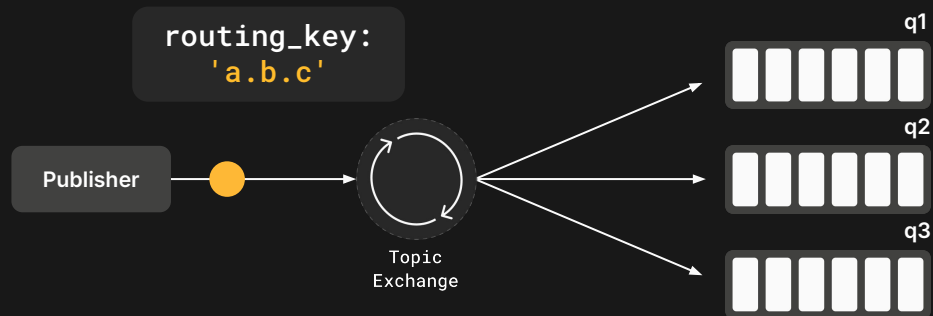
*# (hash) can substitute for zero or more words.*

## Topic Exchange



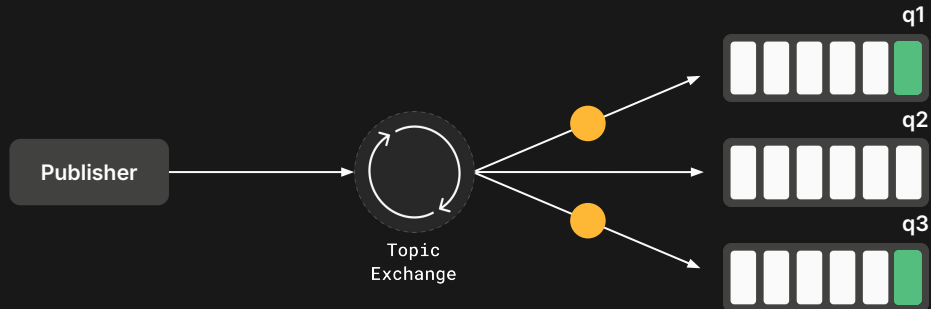
```
7 q1 = client.queue("q1")
8 q2 = client.queue("q2")
9 q3 = client.queue("q3")
10 ex = client.topic
11 q1.bind(ex, "a.b.c")
12 q2.bind(ex, "a.*")
13 q3.bind(ex, "a.#")
```

## Topic Exchange



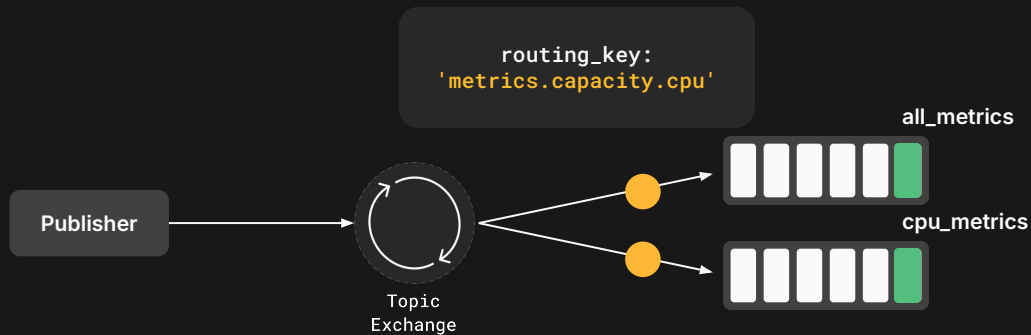
```
7  q1 = client.queue("q1")
8  q2 = client.queue("q2")
9  q3 = client.queue("q3")
10 ex = client.topic
11 q1.bind(ex, "a.b.c")
12 q2.bind(ex, "a.*")
13 q3.bind(ex, "a.#")
14 ex.publish("MINASWAN", routing_key: "a.b.c")
```

## Topic Exchange



```
7  q1 = client.queue("q1")
8  q2 = client.queue("q2")
9  q3 = client.queue("q3")
10 ex = client.topic
11 q1.bind(ex, "a.b.c")
12 q2.bind(ex, "a.*")
13 q3.bind(ex, "a.#")
14 ex.publish("MINASWAN", routing_key: "a.b.c")
```

## Topic Exchange



```
7 all_metrics = client.queue("all_metrics")
8 cpu_metrics = client.queue("cpu_metrics")
9 ex = client.topic
10 all_metrics.bind(ex, "metrics.#")
11 cpu_metrics.bind(ex, "metrics/*.cpu")
12 ex.publish("50%", routing_key: "metrics.capacity.cpu")
13 ex.publish("10%", routing_key: "metrics.capacity.ram")
```

# Workshop overview

## Task 1



[workshop.lavinmq.com](https://workshop.lavinmq.com)

# Workshop overview

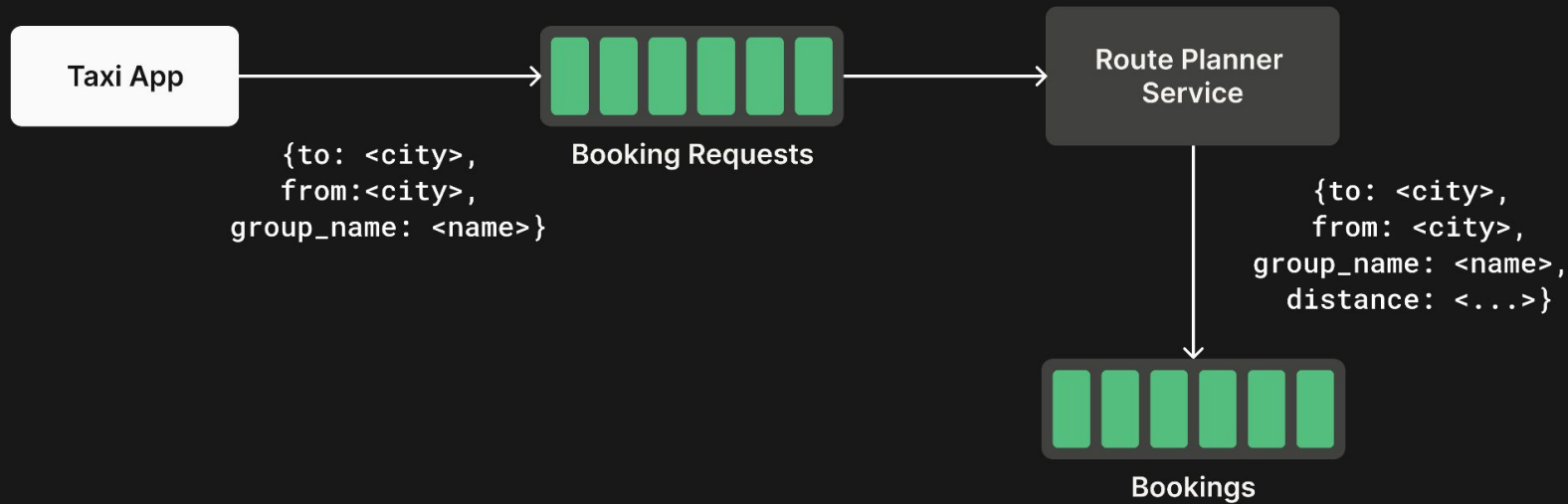
## Task 1



[workshop.lavinmq.com](https://workshop.lavinmq.com)

## Task 1

## Basic consume and publish



# Workshop overview

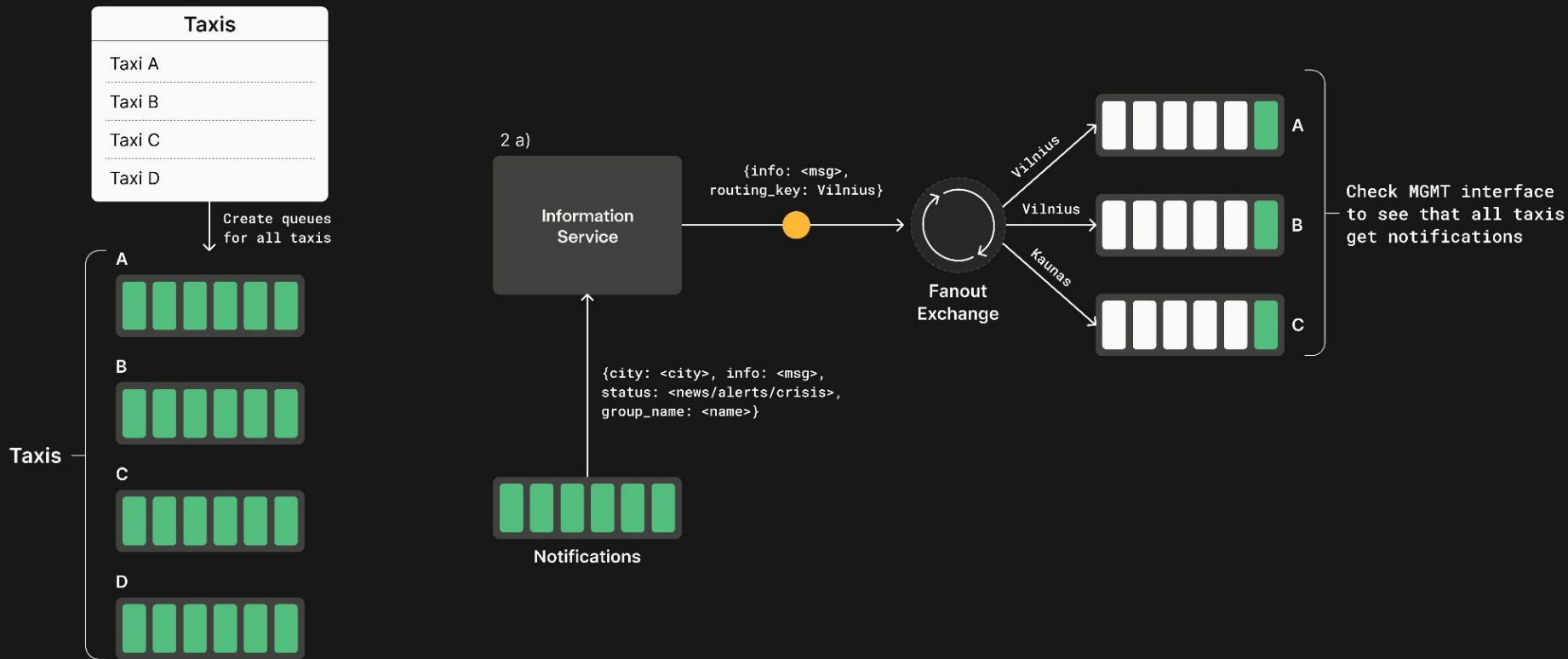
## Task 2



[workshop.lavinmq.com](https://workshop.lavinmq.com)

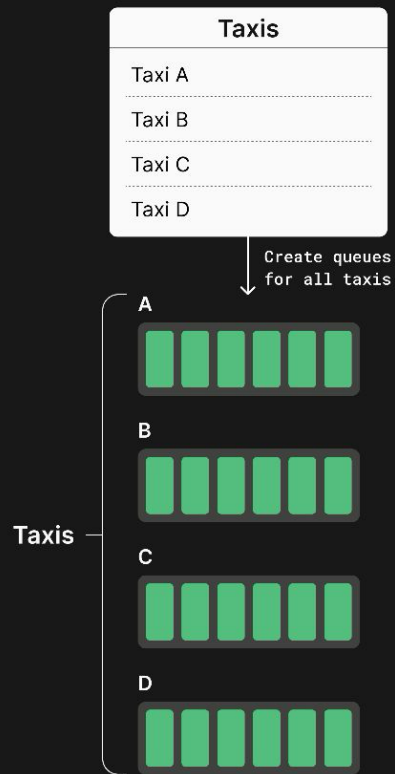
## Task 2A

## Route messages using Fanout Exchanges



## Task 2A

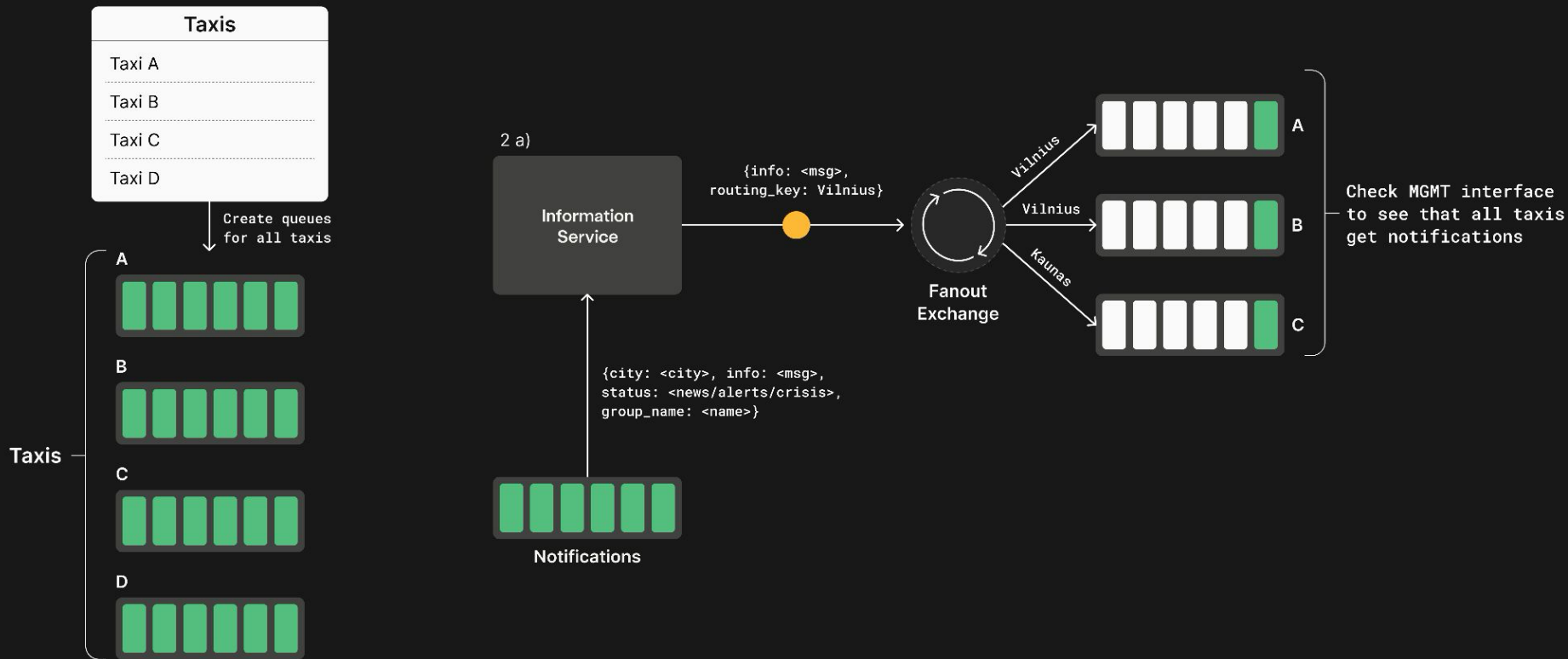
### Route messages using Fanout Exchanges



<https://workshop.lavinmq.com/taxis>

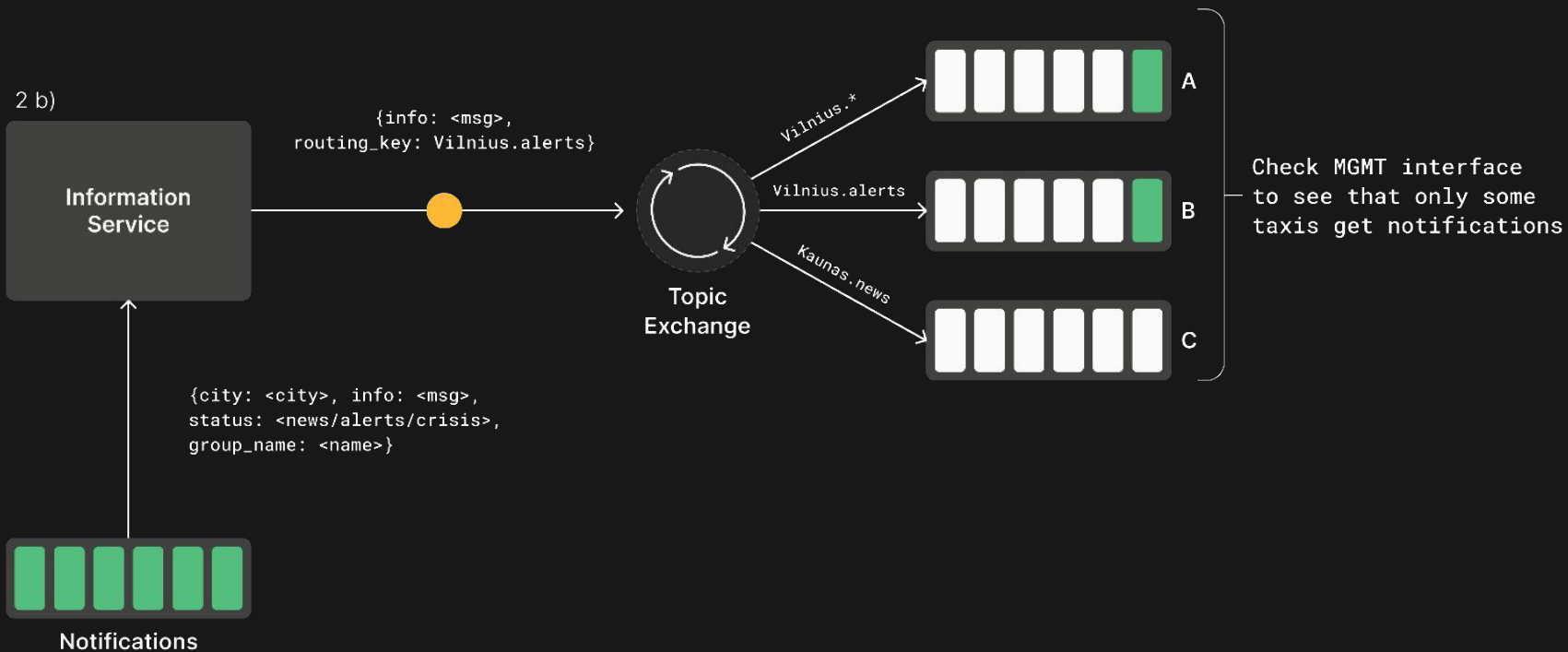
## Task 2A

## Route messages using Fanout Exchanges



## Task 2B

### Route messages using Topic exchanges



# Let's get started!



[workshop.lavinmq.com](https://workshop.lavinmq.com)

# Don't forget! cloudamqp.com for an easy way to try out LavinMQ

### SHARED INSTANCES

For development or small hobby projects. Not recommended for production due to variable performance.



#### Elegant Ermine

- Max 2000 queues
- Max 200 000 queued messages

20 M

Max MSGs/month

200

Max connections

\$ 19

PER MONTH

Get Now



#### Loyal Lemming

- Max 200 queues
- Max 20 000 queued messages
- Max idle queue time 28 days

2 M

Max MSGs/month

40

Max connections

FREE

Get Now

 LAVINMQ