

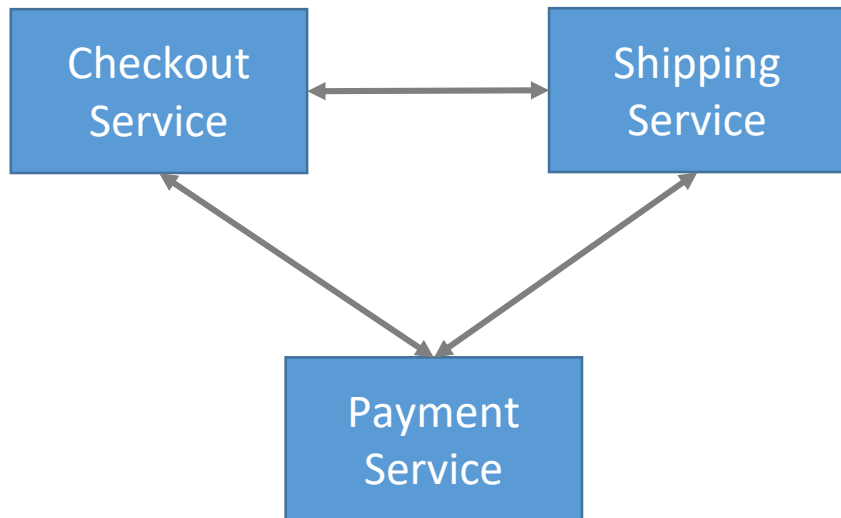
Decoupling Application

Contents

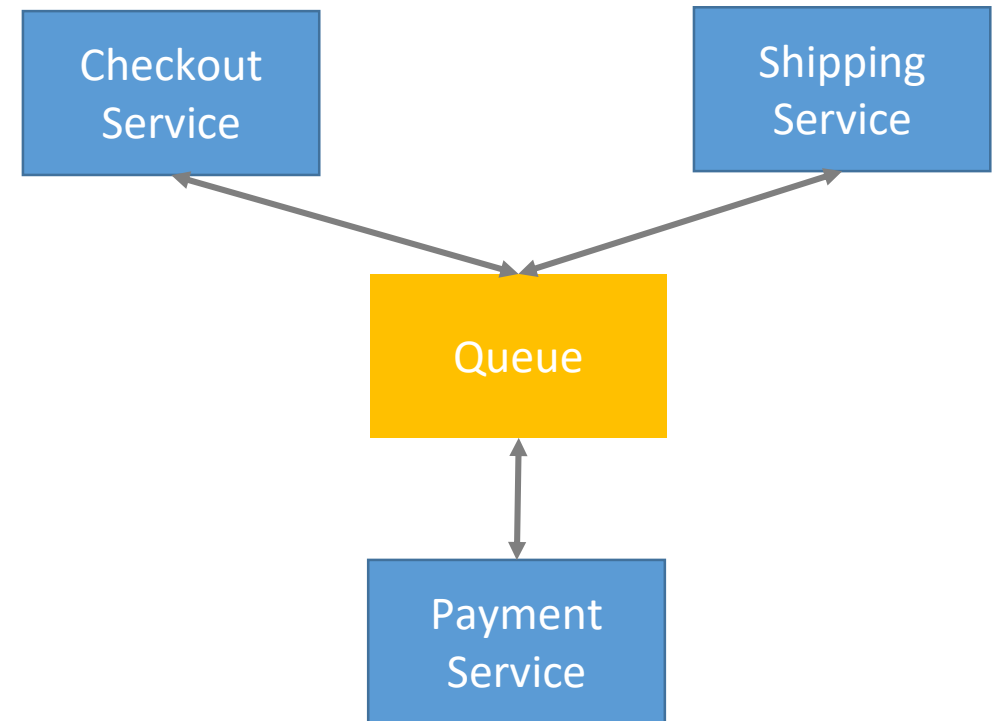
- Decoupling application
- SQS, SNS

Decoupling application

- Tight Coupling



- Loose Coupling



Decoupling application (cont.)

Items	Tight Coupling	Loose Coupling
Scalability	Hard to scale due to the application tight to each others	Application can be easy scale due to loose coupling among them
Stalibility	If the target application is interrupted, the message will be lost	The message is stored in queue
Speed	Direct connect between applications	Via a message queue

SQS

SQS



- SQS stand for Simple Queue Service
- Fully managed Queue service

SQS – Standard Queue



Attributes

- Unlimited throughput and unlimited number of messages in Queue
- Default retention period is up to 14 days (default 4 days)
- 256 KB size limitation for each messages
- Low latency (< 10ms for sending/receiving message)
- At least one delivery strategy
- Best Effort Ordering

SQS – FIFO Queue



Attributes

- Same features as **Standard Queue**
- **FIFO** (First In First Out) delivery
- Exactly-one processing
- Limited throughput 300 transaction per second (TPS)

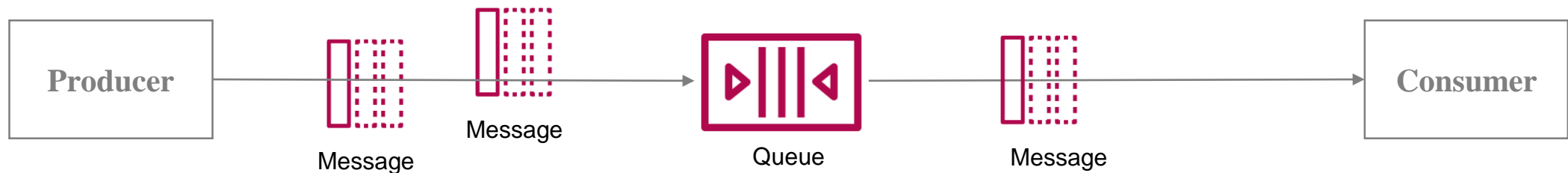
SQS – Producing/Consuming messages

Producing messages

- Produced messages to SQS using the SDK
- Message is persisted in SQS until Consumer deletes it
- Message retention period: 4 days (default) and up to 14 days

Consuming messages

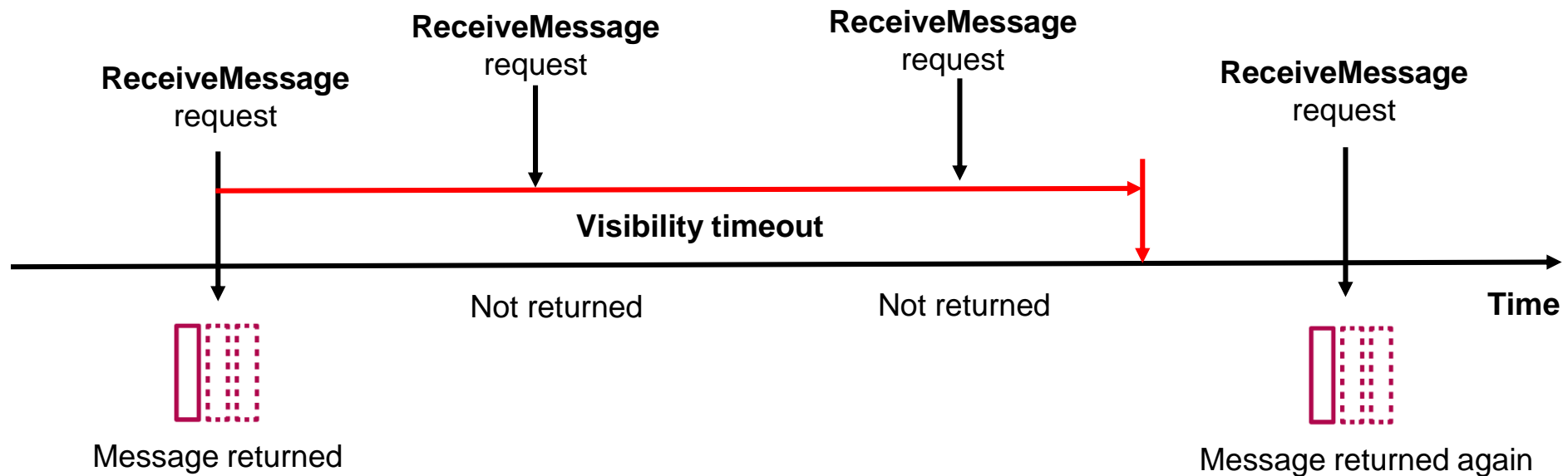
- Consumers run on EC2, Servers, Lambda
- Poll SQS for messages
- Process the messages
- Delete messages after processing completed



SQS - Message Visibility Timeout

- The message will be **invisible** to other consumers after one consumer pulled it out from **Queue**
- Visibility timeout is 30 seconds (by default)
- After the message visibility timeout is over, the message is visible to **other consumers**

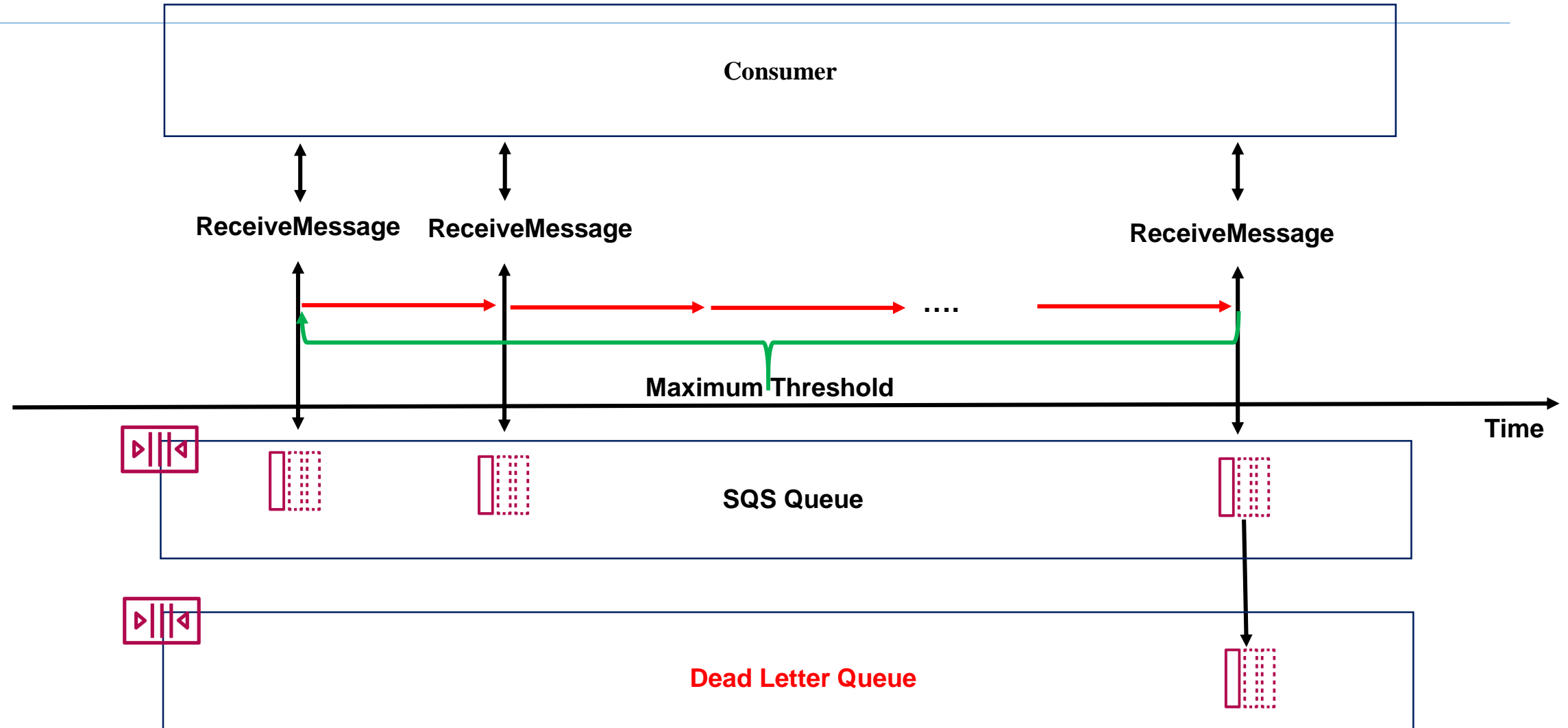
SQS - Message Visibility Timeout (cont.)



SQS - Message Visibility Timeout (cont.)

- If visibility timeout is short, the message may be processed twice
- If visibility timeout is high and the consumer is crashed, it takes time to re-process

SQS - Dead Letter Queue



Exam Tips

- Talking about Loosing Coupling, Decoupling => **SQS**
- SQS is pull-based, not push-based
- Messages are 256 KB in size limitation
- Messages are kept in Queue from 1 minute to 14 days. Default retention period is 4 days
- SQS guarantes that your messages will be deliveried at least once

Exam Tips

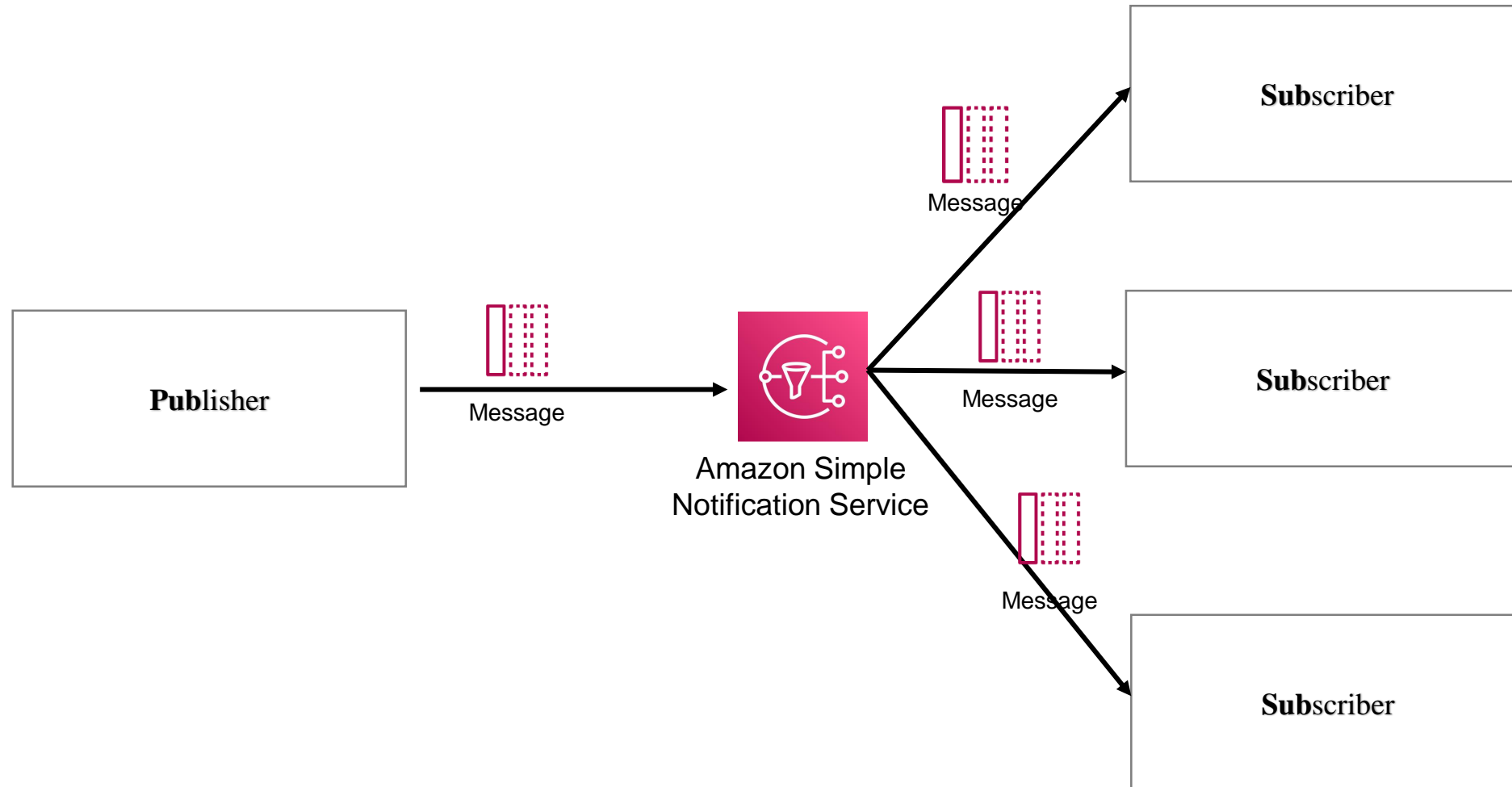
- Polling strategy
 - **Long-polling** will wait until there are message available in queue
 - **Short-polling** will return a reponse immediately even if there is no message in Queue
- Visibility timeout up to 12 hours
- If messages are processed twice by consumer => Increase Visibility timeout
- We can set a **threshold** of how many times a message can go back to the Queue

Exam Tips

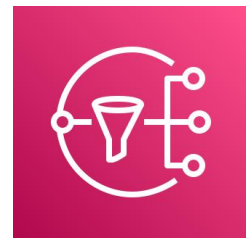
- DLQ is useful for Debugging
- Set retention period as long for your debugging time (~ 14 days)

Simple Notification Service

Pub/Sub Model



SNS



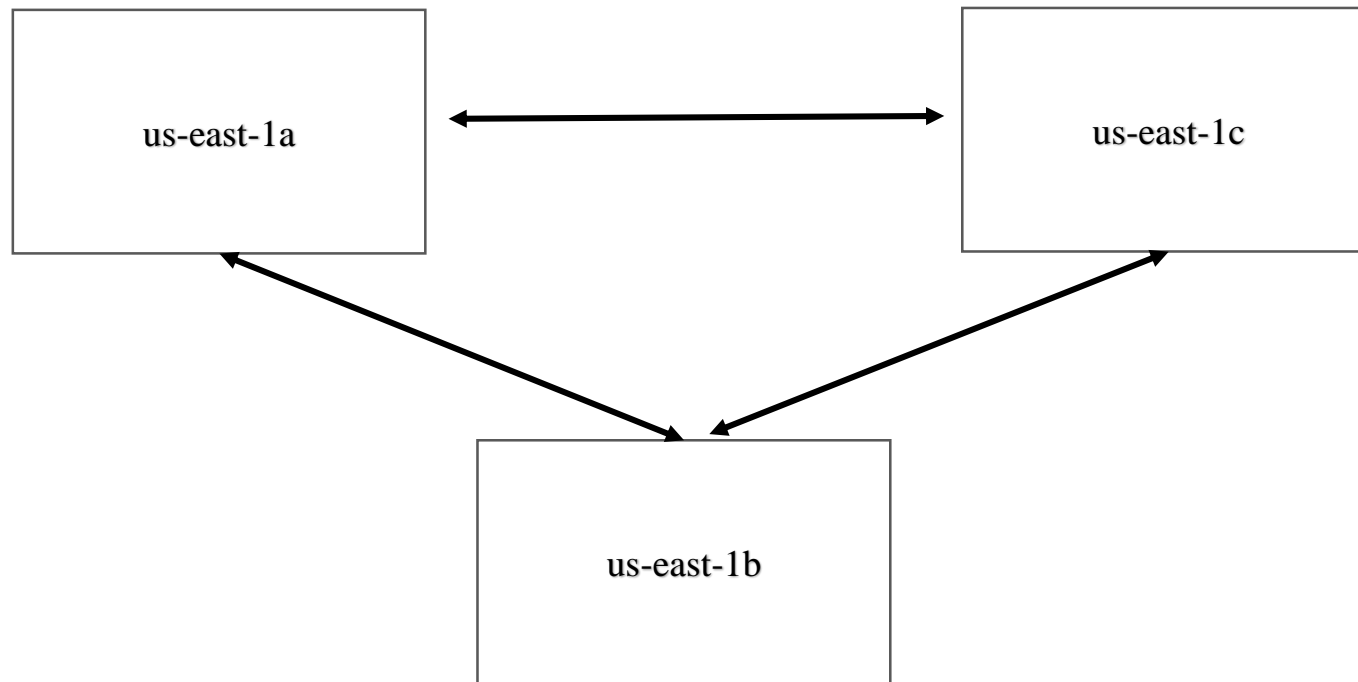
- Provides message delivery from publishers to subscribers
- Subscribers subscribe into the topic and will get all the messages from this topic
- Up to 10.000.000 subscriptions per topic
- Up to 100.000 topic limit

SNS Topic

- Use Topic to group recipients (subscribers)
- All recipients (subscribers) in the same topic will receive the same copy of message notification
- Support endpoint type:
 - AWS services (SQS, Kinesis Firehose, Lambda, Custom HTTP endpoint)
 - Email, SMS, Mobile Push

SNS availability

- Message which sent to SNS are stored redundant across multiple AZs



SNS security

- **Encryption**

- In transit using SSL
- Encryption at rest using KMS or your own managed key

- **Access Control**

- Using IAM identity policy to grant permission to access to SNS

- **Access Policies**

- Using Resources-based policy to grant permission

Exam Tips

- SQS and SNS both are managed messaging service
- SQS is **poll-based** and SNS is **push-based**