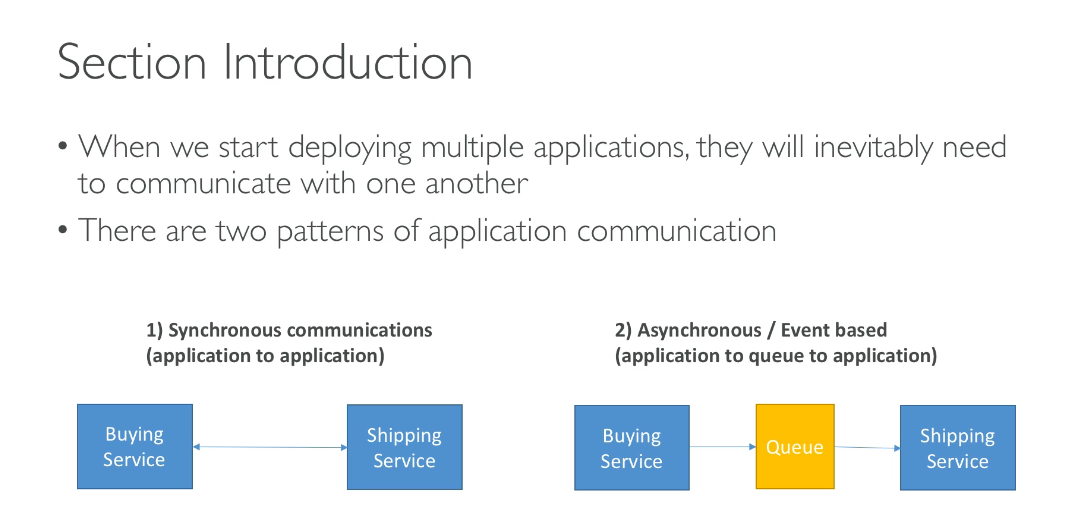
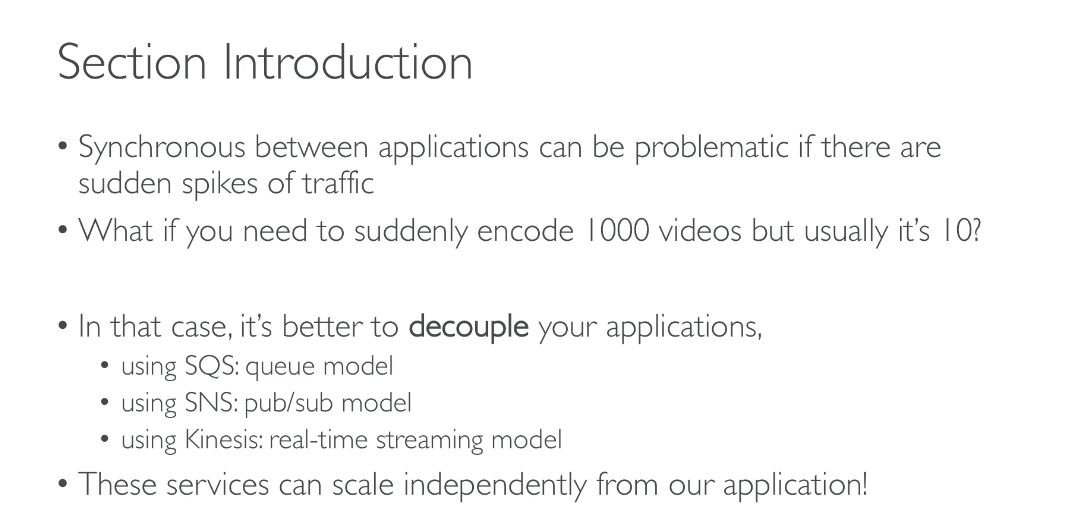
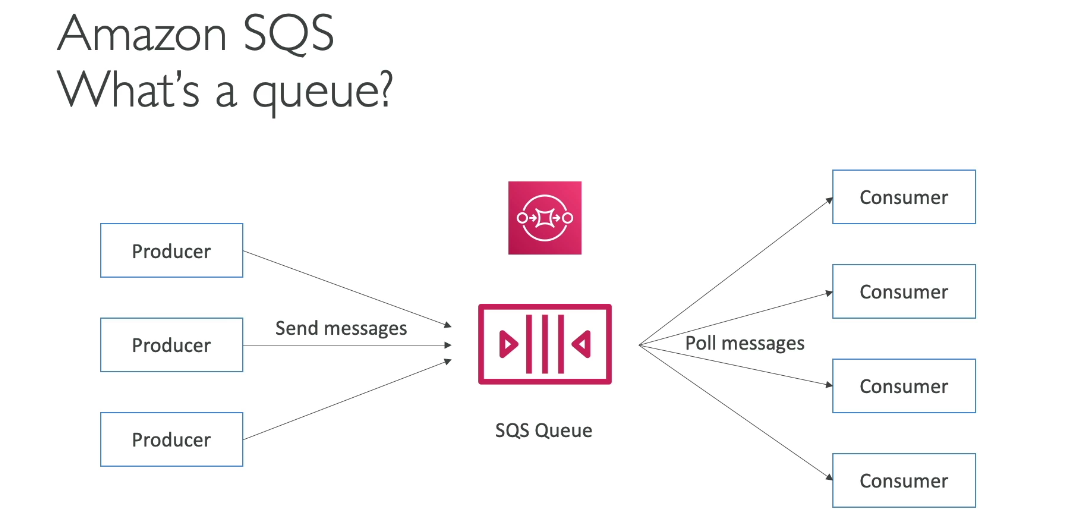
**SQS, SNS, KINESIS & ACTIVE MQ**

**Messaging System**





**SQS:**



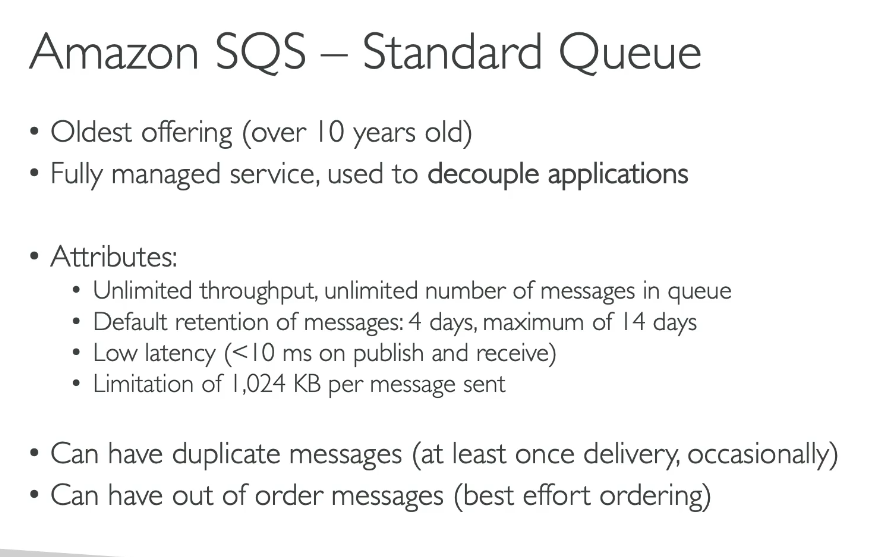
**Amazon SQS offers two types of queues:**

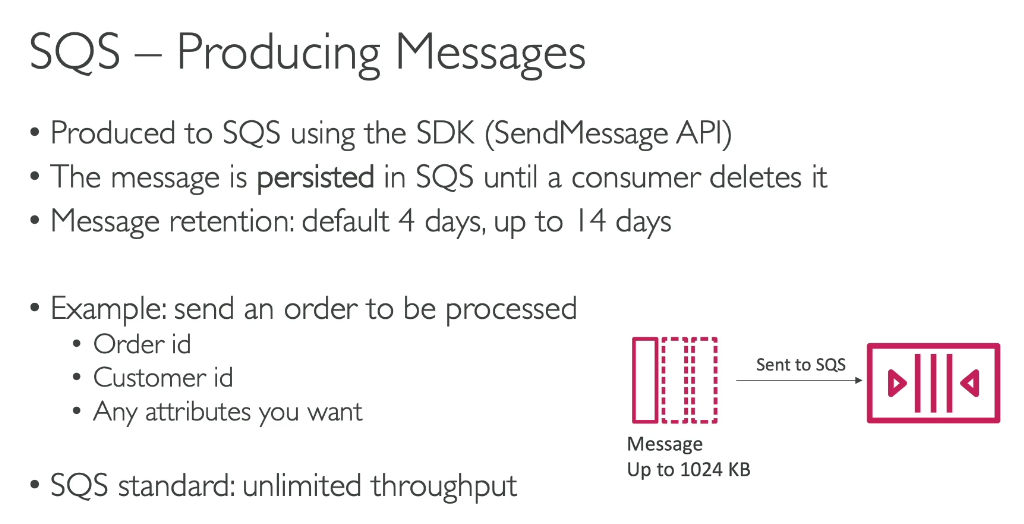
**Standard Queues:**

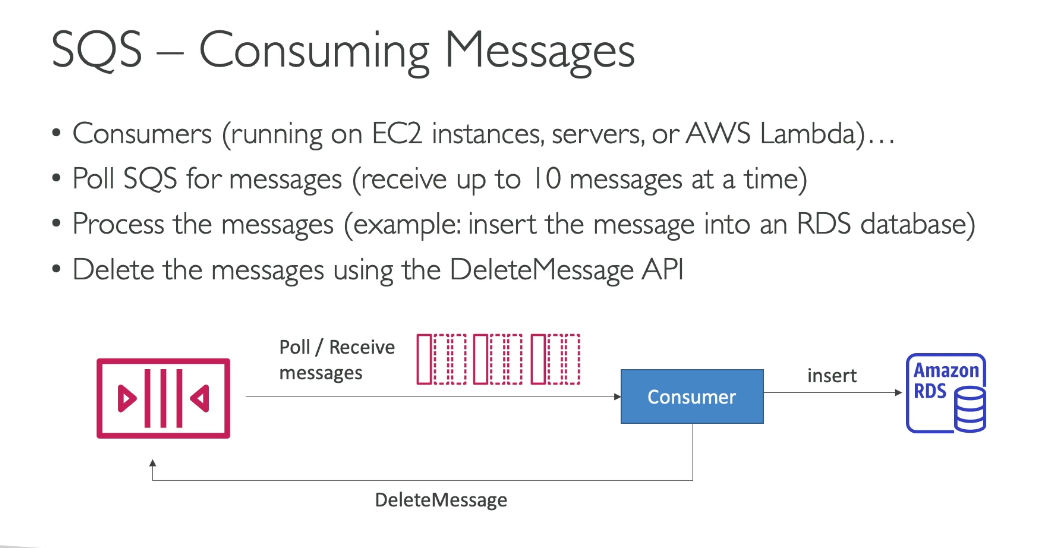
* Support nearly unlimited throughput with a very high number of API calls per second
* Provide at-least-once delivery, meaning a message is delivered at least once, but occasionally more than one copy may be delivered
* Offer best-effort ordering, where messages might occasionally be delivered in a different order than they were sent
* Ideal for scenarios requiring high throughput and where occasional duplicate messages or out-of-order delivery is acceptable

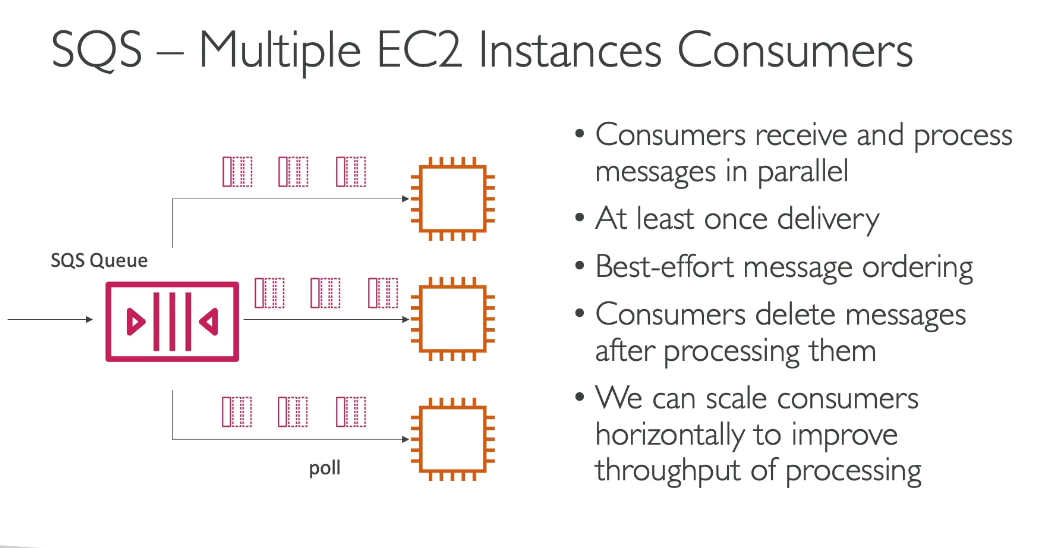
**FIFO Queues (First-In-First-Out):**

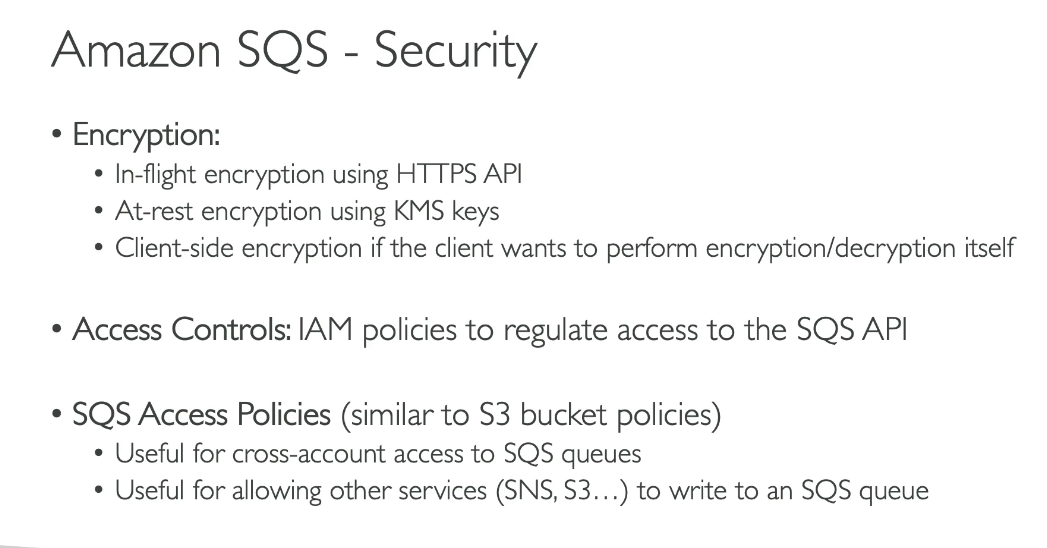
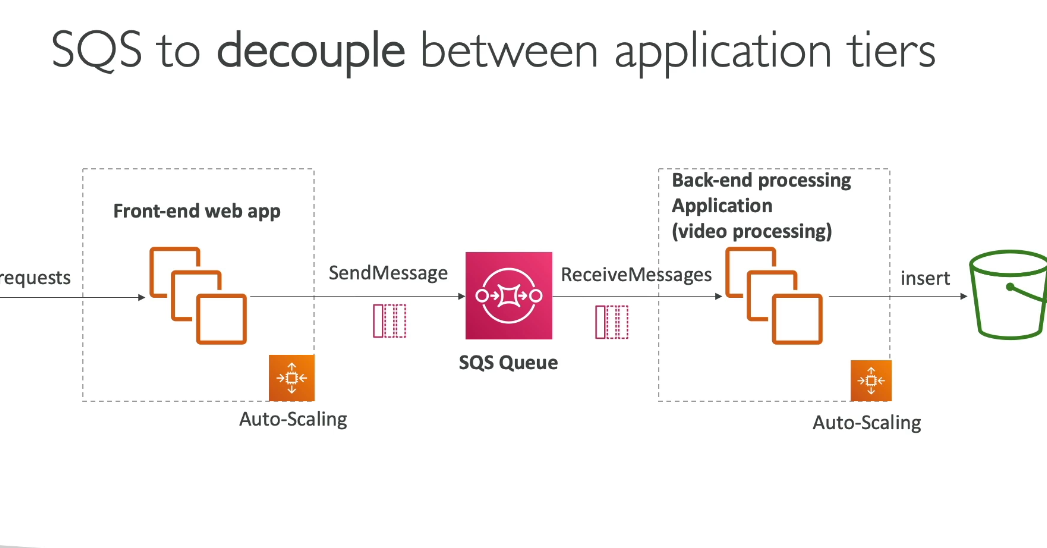
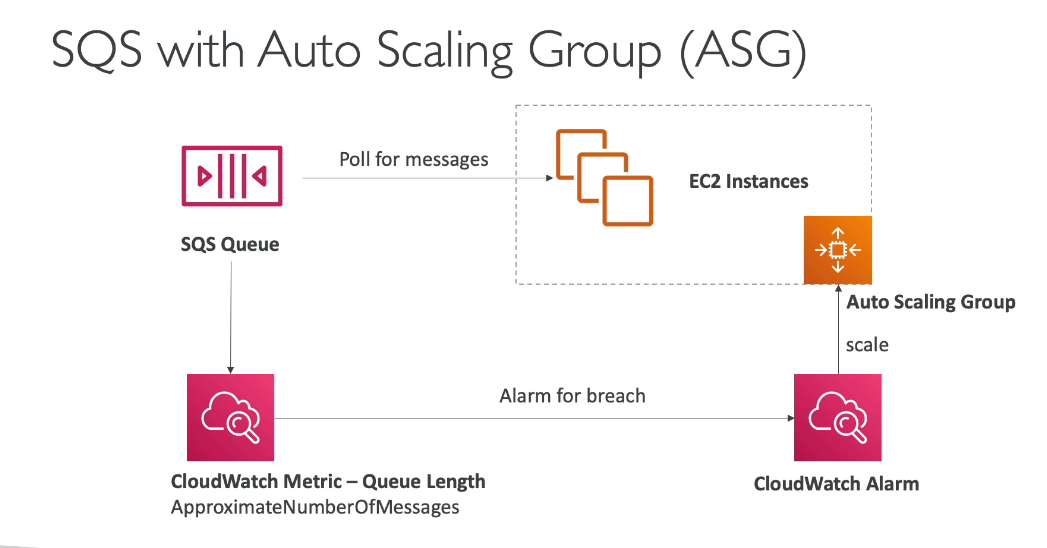
* Guarantee that messages are processed exactly once in the exact order they are received
* Support up to 300 transactions per second (TPS) per API action by default
* With batching, can handle up to 3,000 messages per second
* High throughput mode is available for increased capacity
* Queue names must end with the ".fifo" suffix
* Best suited for applications requiring strict message ordering and exactly-once processing

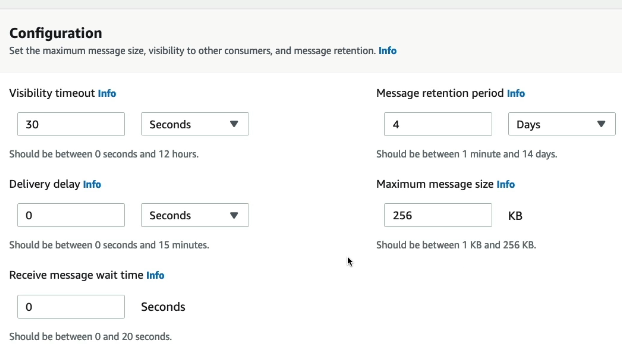




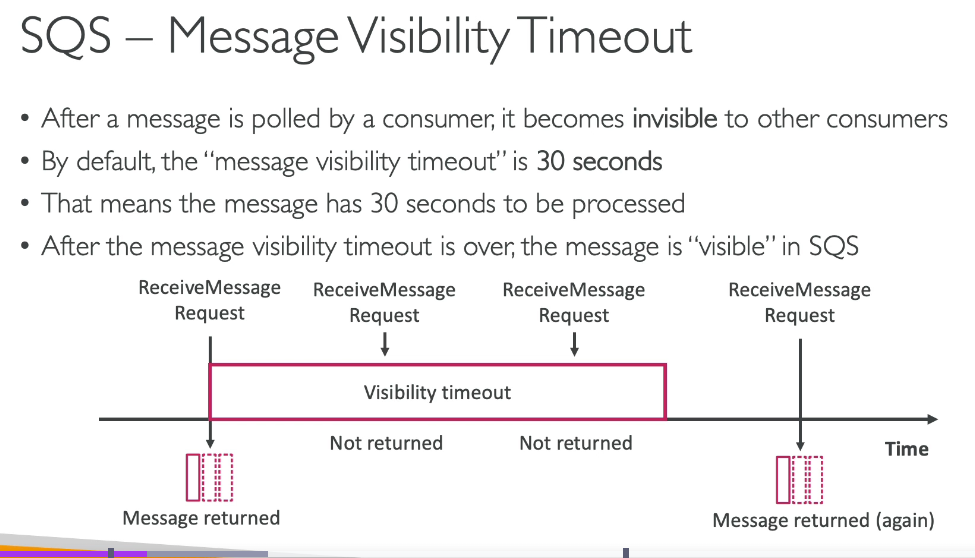


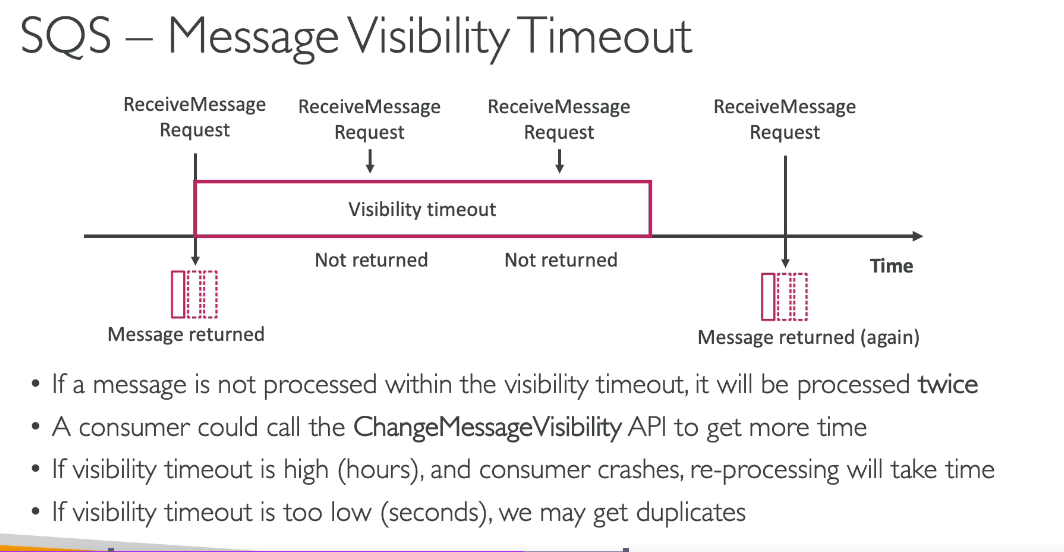


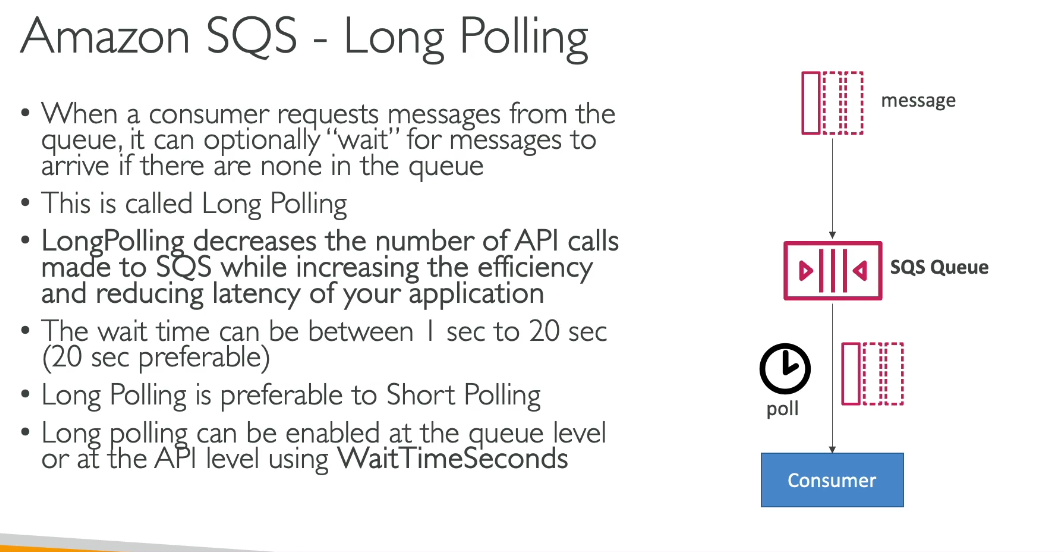


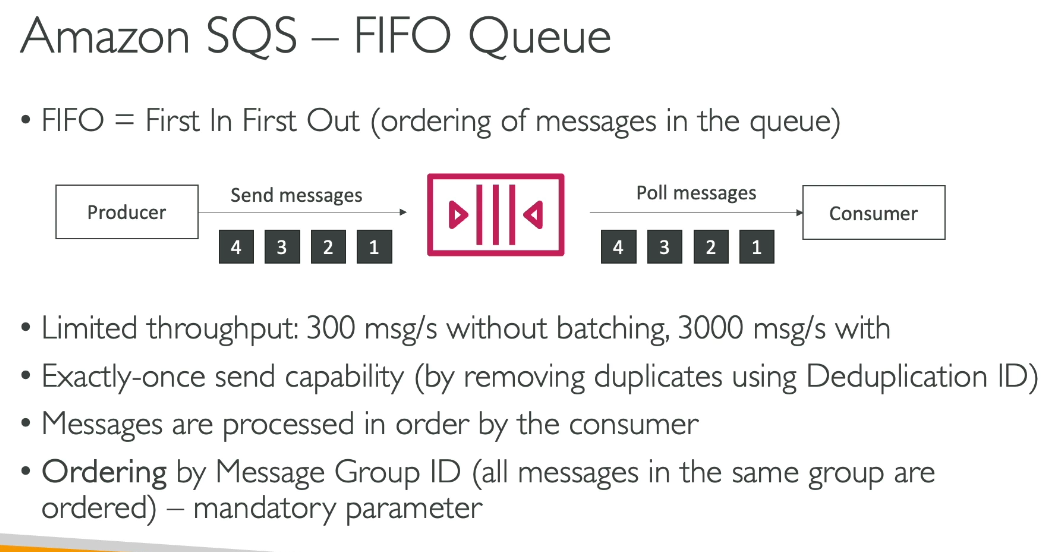


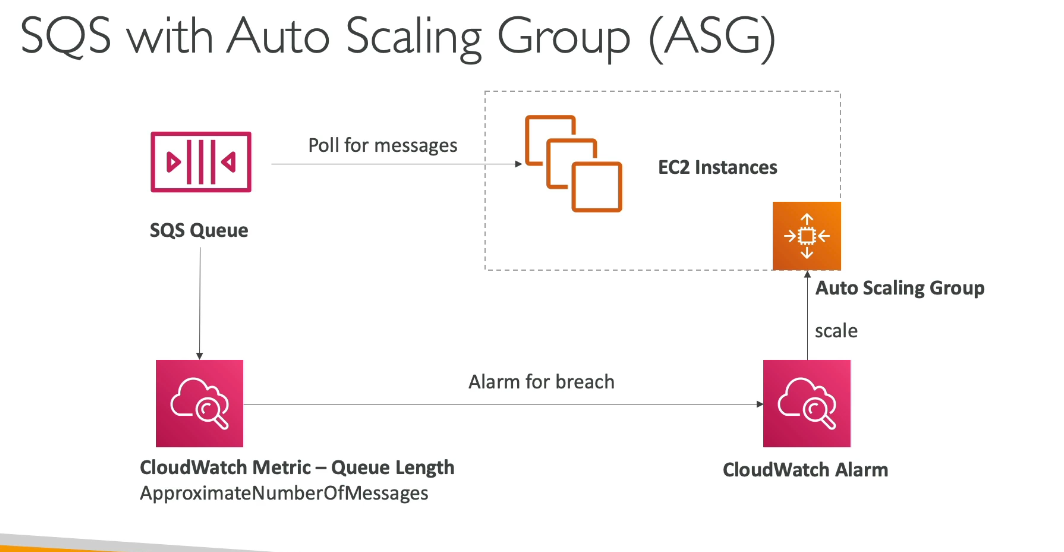
|  |  |
| --- | --- |
| **Setting** | **Purpose** |
| Visibility Timeout | Prevent parallel processing |
| Retention Period | How long messages live |
| Delivery Delay | Delay before first visibility |
| Max Message Size | Payload limit |
| Receive Wait Time | Long vs short polling |

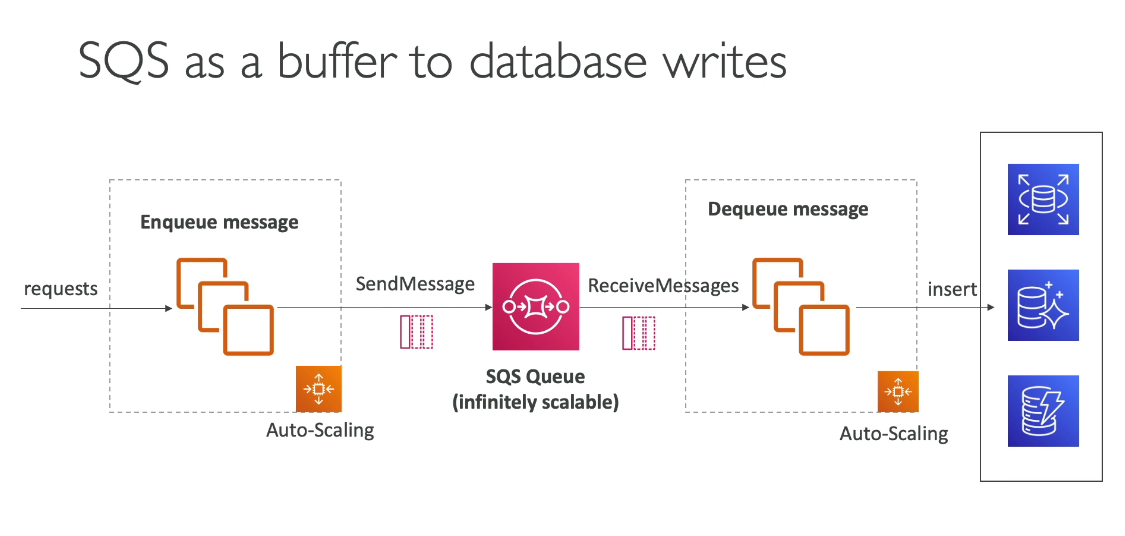


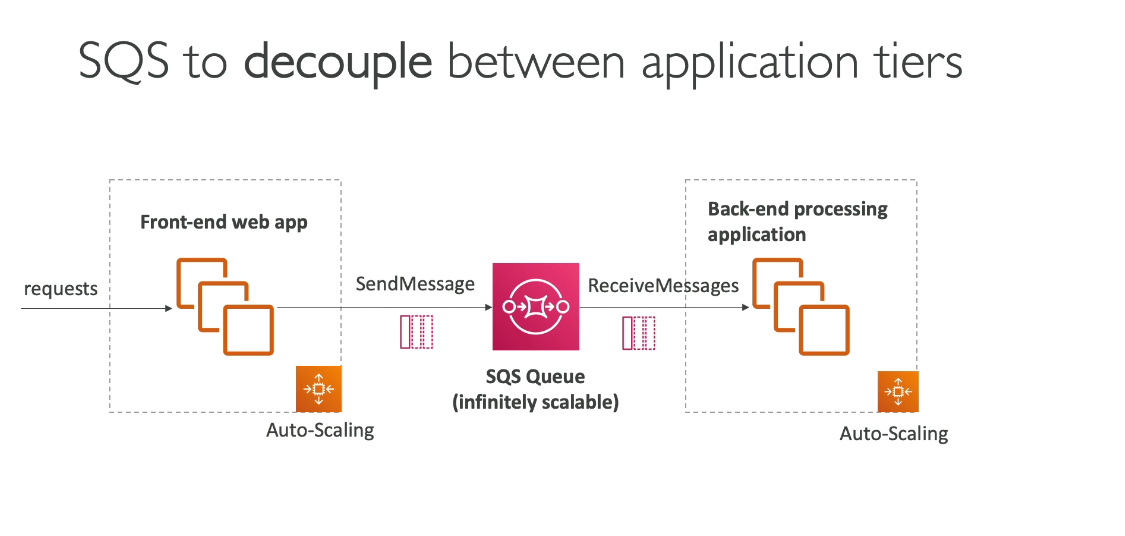




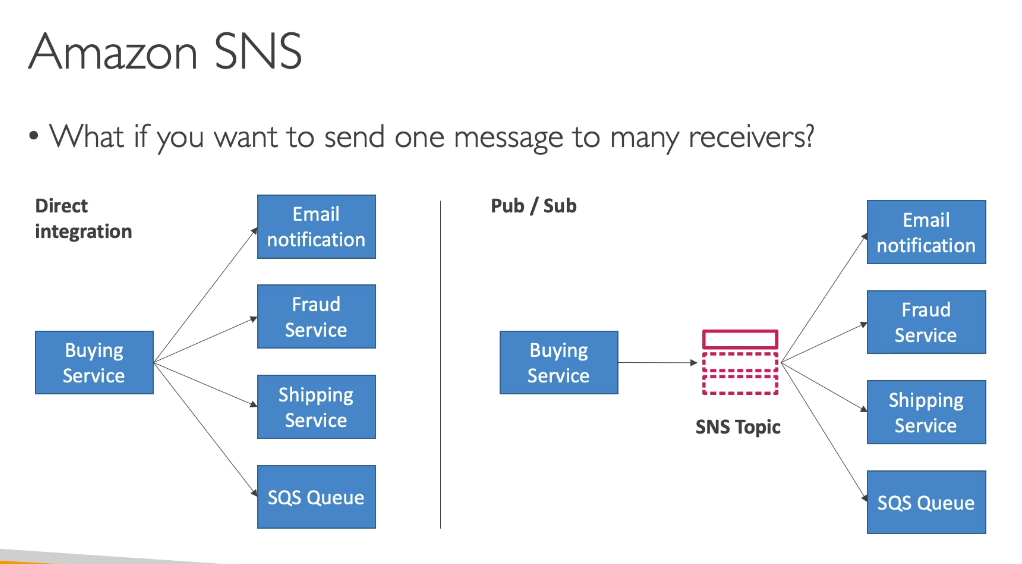


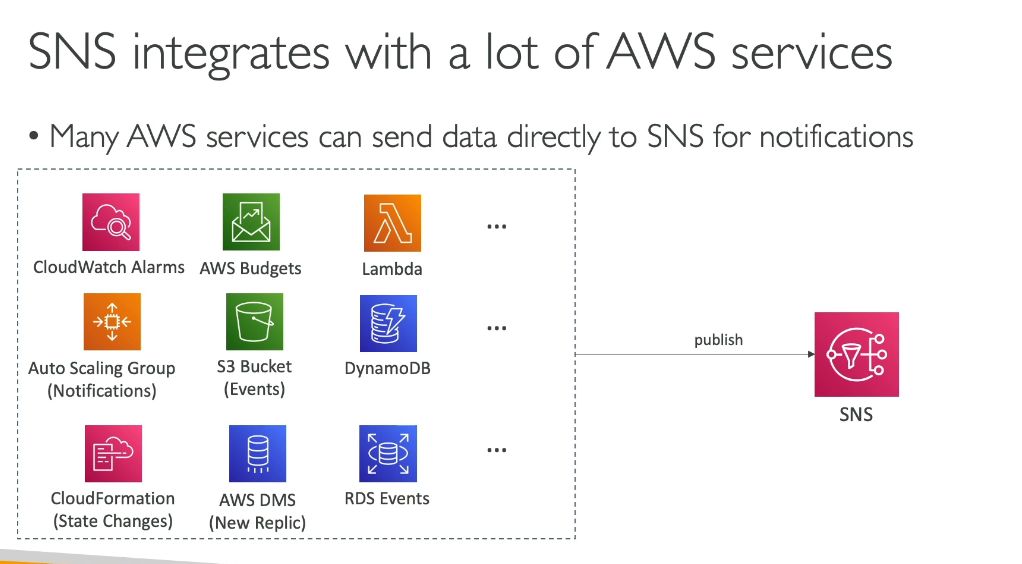


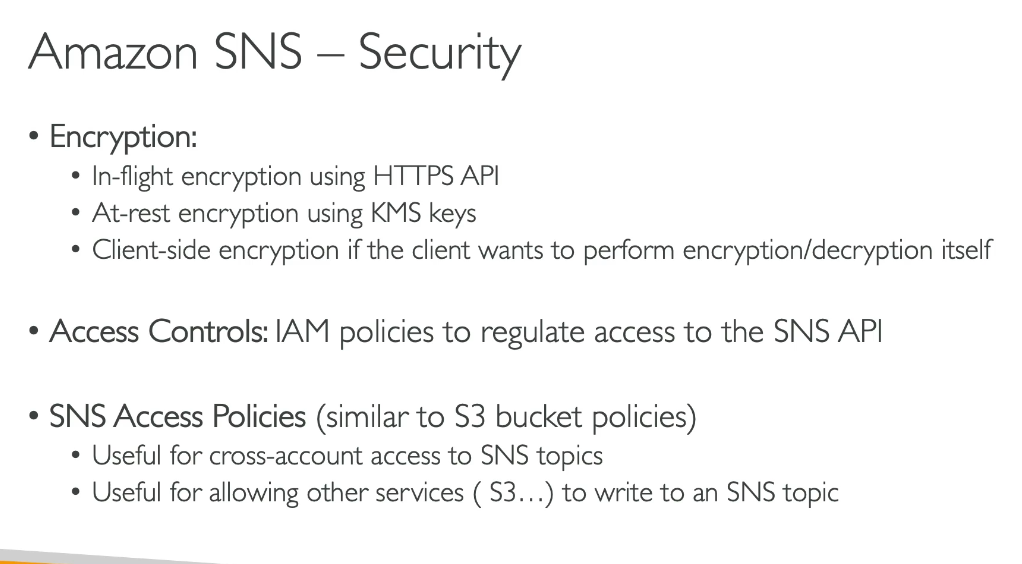


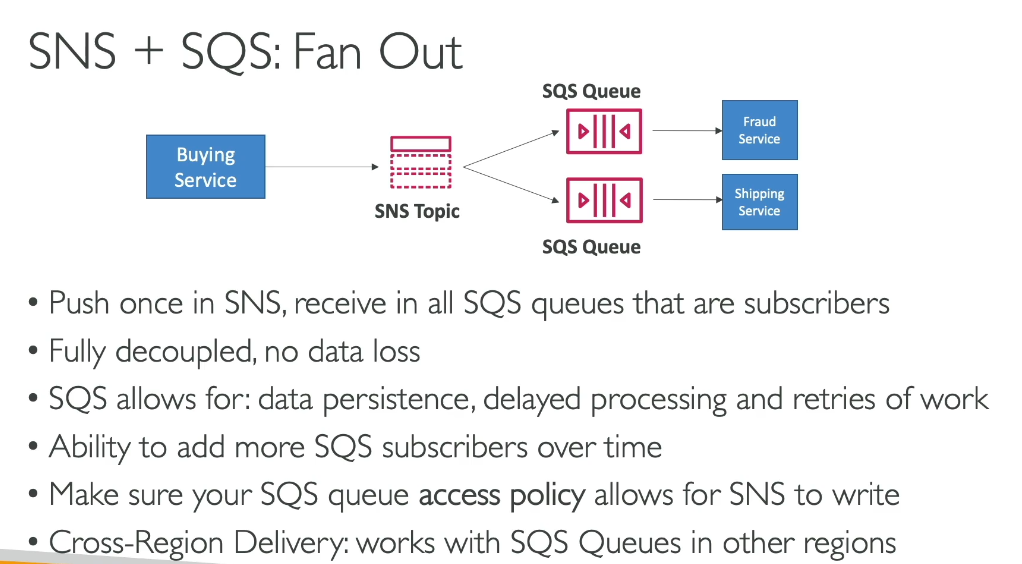


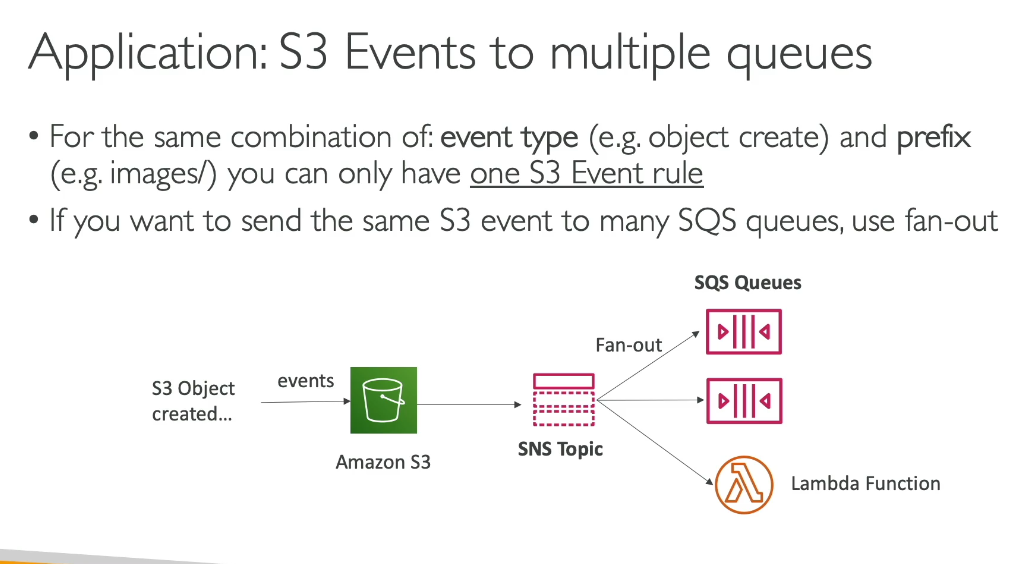
**SNS:**



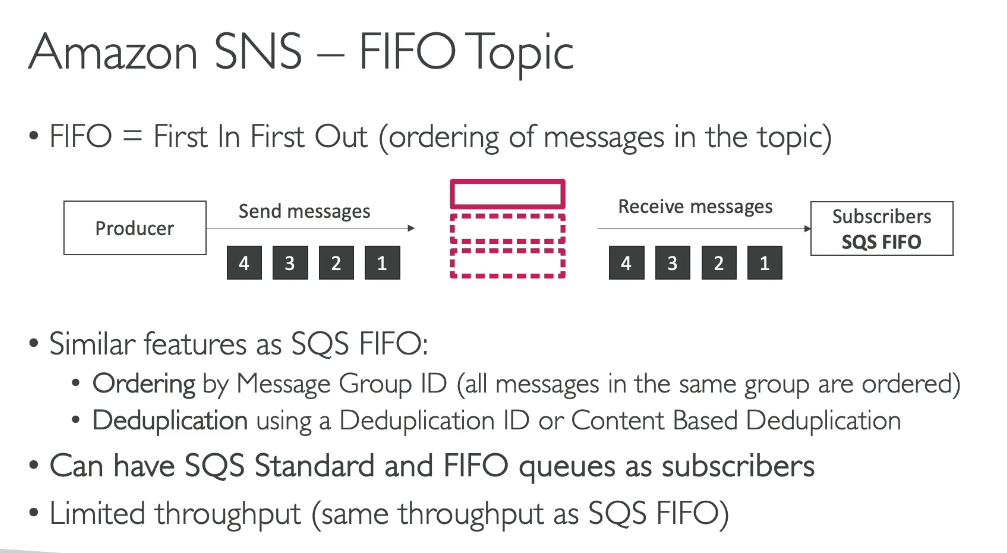


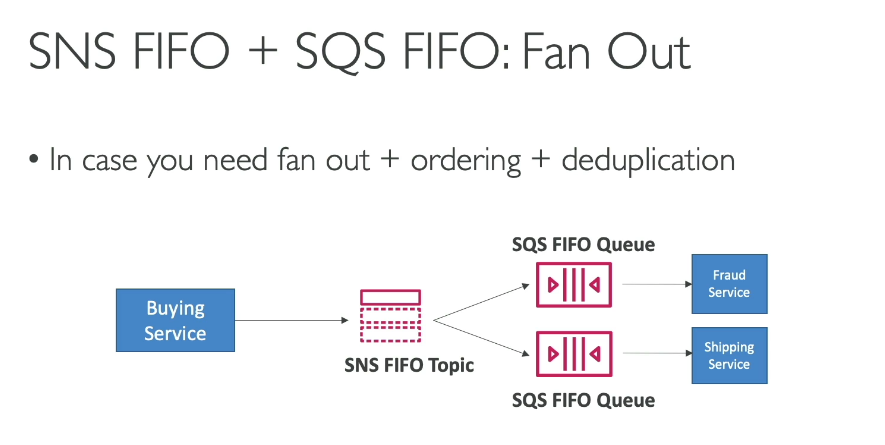


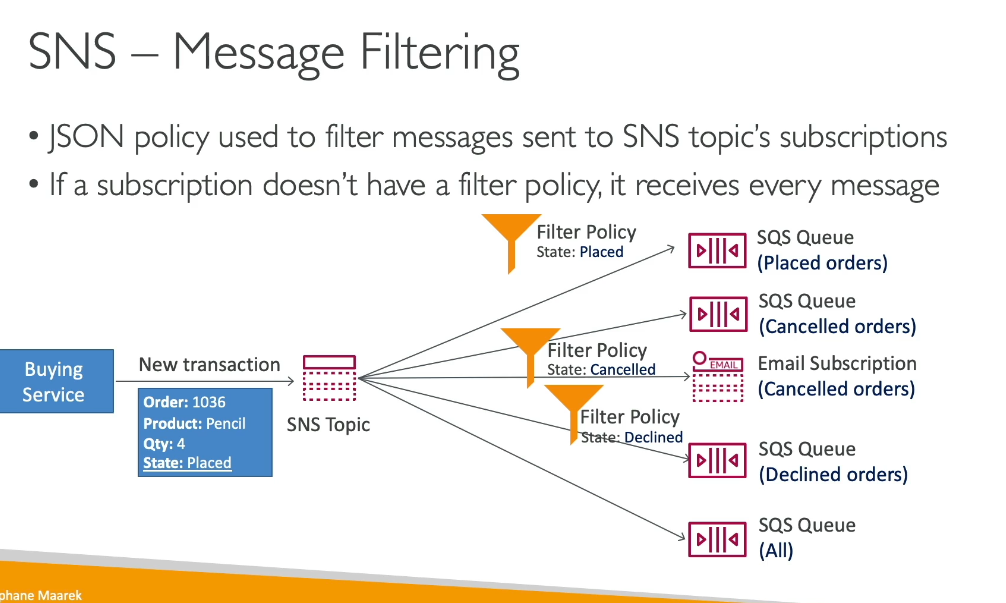




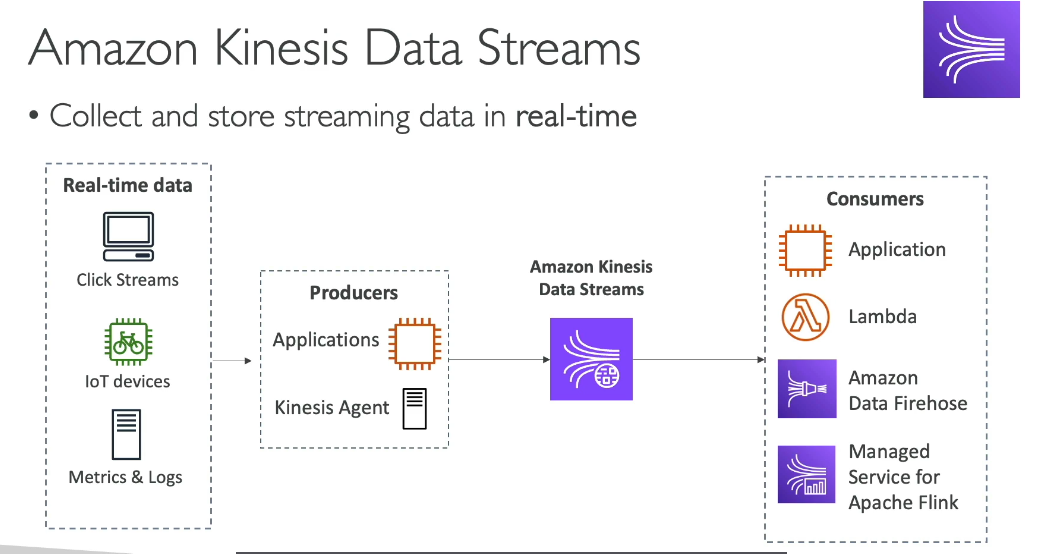


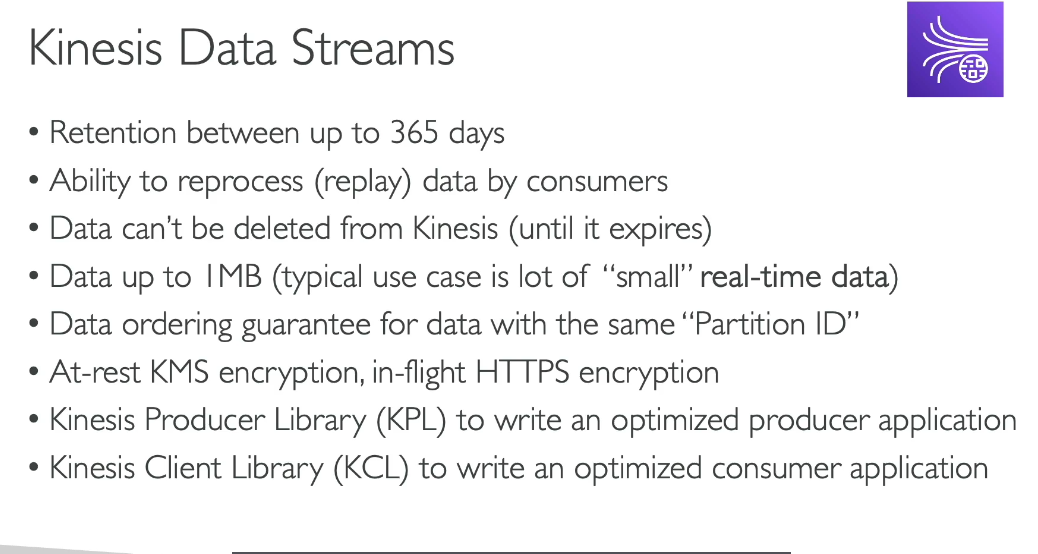


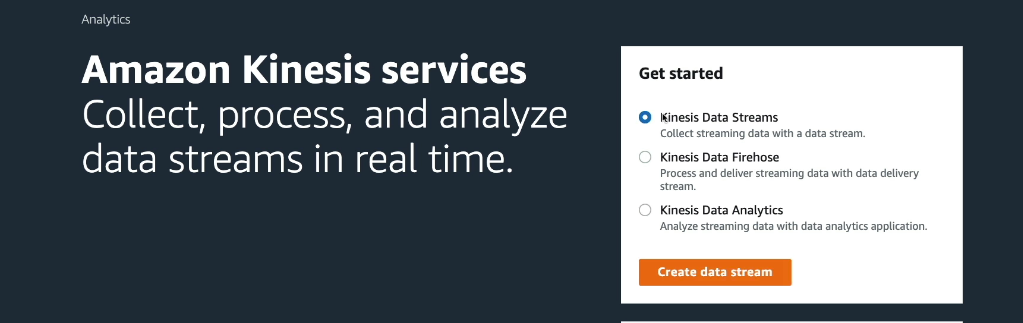


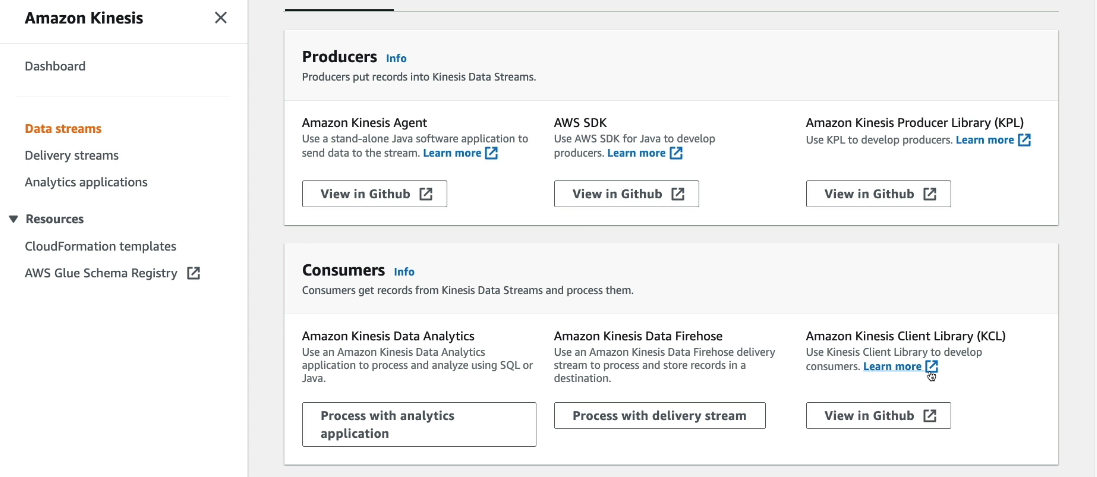


**KINESIS DATA STREAMS:**









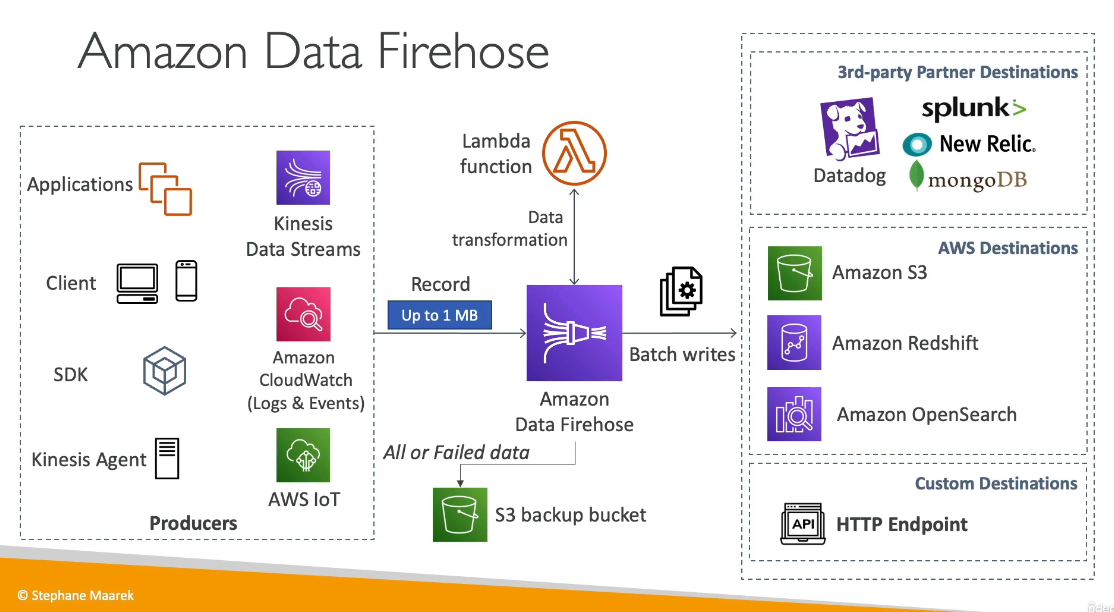
**Fan-out** is a messaging pattern where a single message published to an SNS topic is distributed to multiple SQS queues simultaneously, enabling parallel, asynchronous processing.

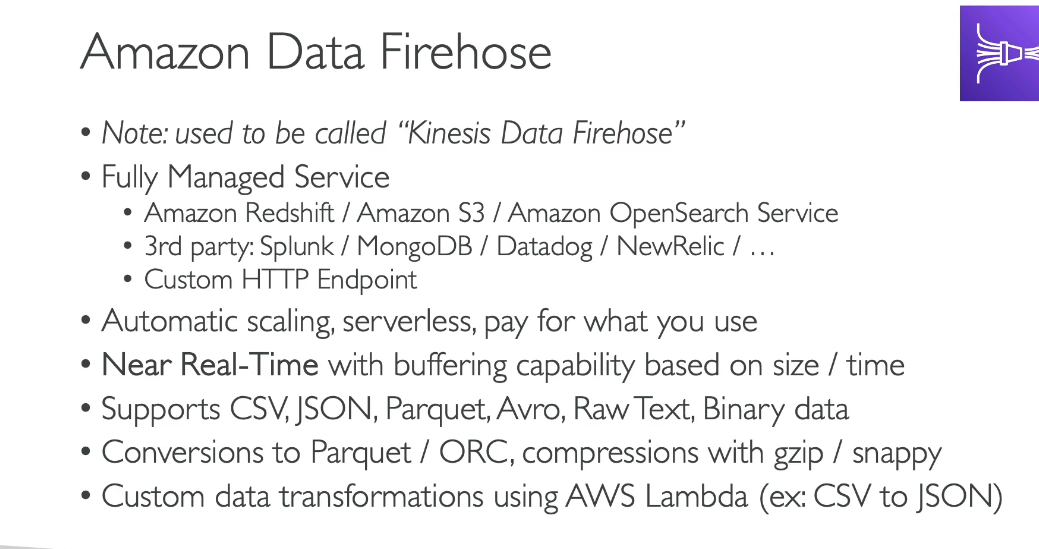
How it works:

1. A message is published to an SNS topic
2. Multiple SQS queues subscribe to that topic
3. Each subscribed queue receives an identical copy of the message
4. Independent processes read from each queue and process the message in parallel

Example - Image Processing: When a new image is uploaded:

* A message is published to an SNS topic
* Multiple SQS queues subscribed to the topic each receive the message
* Different processes work in parallel:
  + One queue handles thumbnail generation
  + Another performs image recognition
  + A third stores metadata





**Buffering** in Amazon Kinesis Data Firehose is the process of temporarily collecting incoming streaming data before delivering it to destinations. This helps manage unpredictable data flows and prevents overwhelming downstream systems.

Two buffer conditions:

* Buffer size: The amount of data (in MiBs) that Firehose collects before delivery
  + Range: 1-100 MiBs (can be set to 0 for zero buffering)
* Buffer interval: The time period (in seconds) that Firehose waits before delivery
  + Range: 60-900 seconds (can be set to 0 for zero buffering)

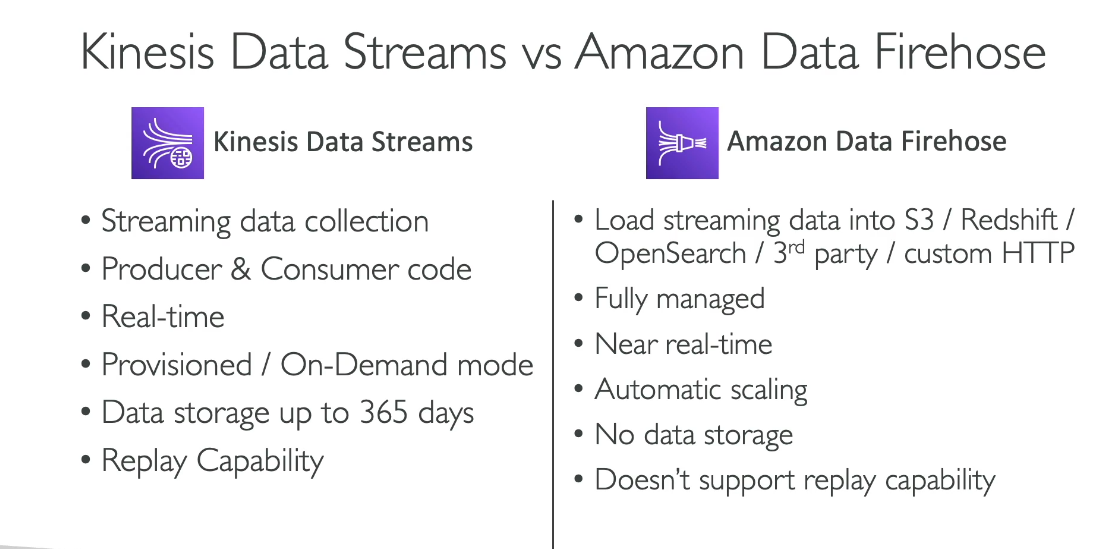
How it works:

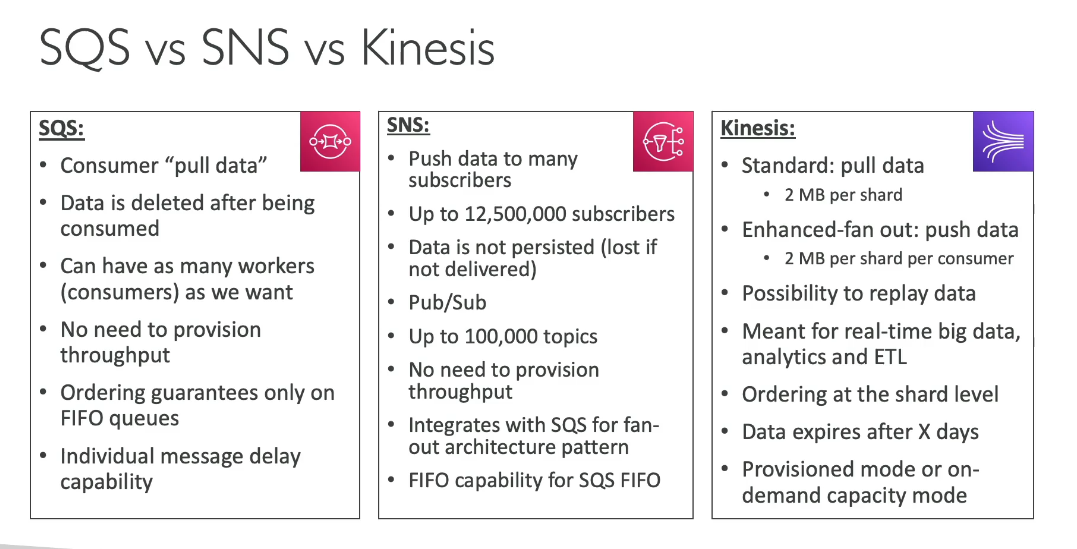
Firehose delivers data when either condition is met first. For example, if you set:

* Buffer size: 5 MB
* Buffer interval: 300 seconds Data will be delivered when either 5 MB accumulates OR 300 seconds pass, whichever happens first.

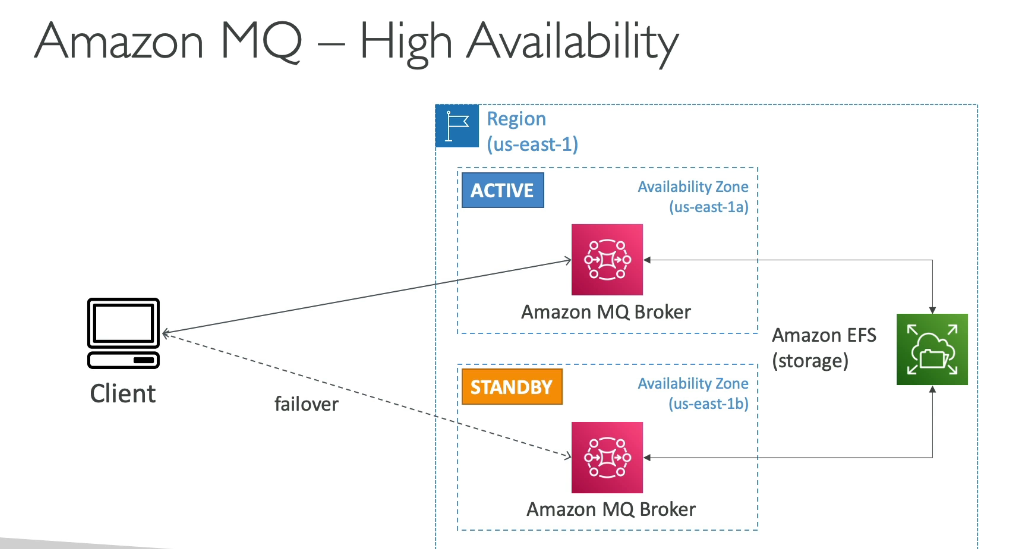
Zero buffering:

Kinesis Data Firehose now supports zero buffering, enabling data delivery within seconds (typically around 5 seconds) for real-time use cases. This is available for destinations like Amazon S3, Amazon OpenSearch Service, Amazon Redshift, and third-party HTTP endpoints.









**Quiz:**

