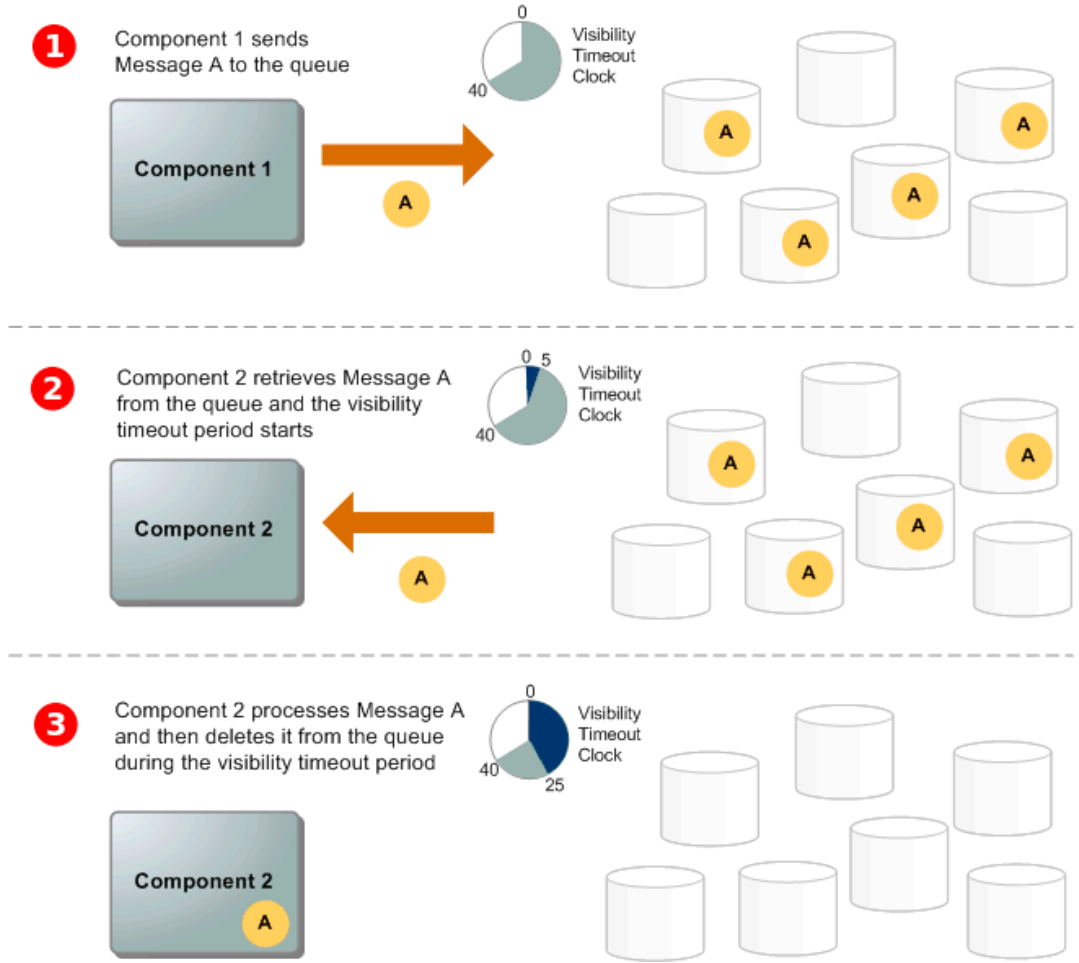
Amazon SQS

What is AWS SQS?

- Amazon Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications.
- SQS eliminates the complexity and overhead associated with managing and operating message oriented middleware, and empowers developers to focus on differentiating work.
- Using SQS, you can send, store, and receive messages between software components at any volume, without losing messages or requiring other services to be available.

Message Lifecycle

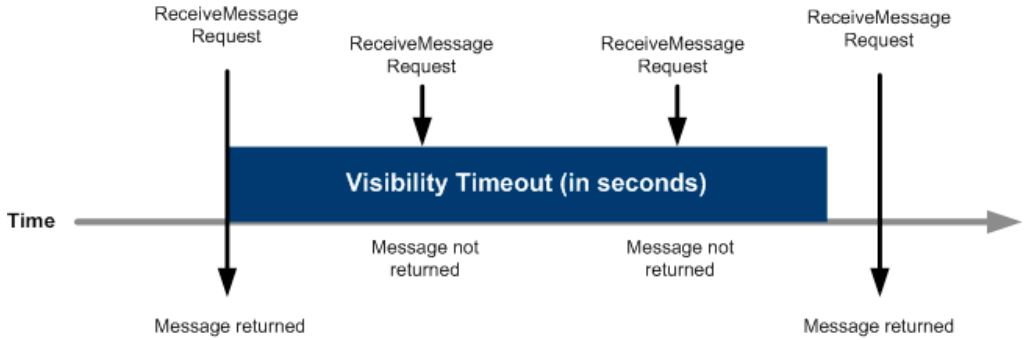


Amazon SQS automatically deletes messages that have been in a queue for more than maximum message retention period. The default message retention period is 4 days. However, you can set the message retention period to a value from 60 seconds to 1,209,600 seconds (14 days)

SQS Queue Types

- SQS offers two types of message queues. Standard queues offer maximum throughput, best-effort ordering, and at-least-once delivery.
- SQS FIFO queues are designed to guarantee that messages are processed exactly once, in the exact order that they are sent.

SQS Visibility Timeout

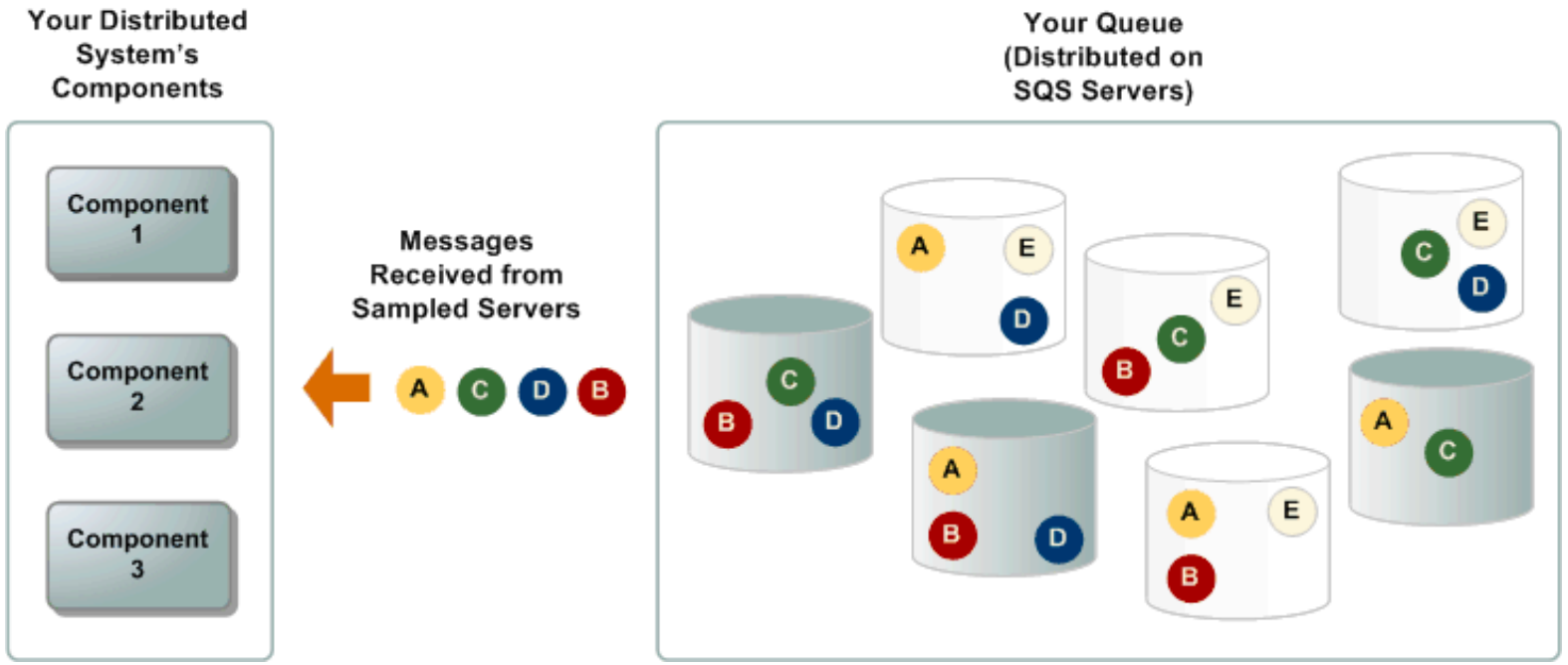


- When a consumer receives and processes a message from a queue, the message remains in the queue. Amazon SQS doesn't automatically delete the message. Because Amazon SQS is a distributed system, there's no guarantee that the consumer actually receives the message (for example, due to a connectivity issue, or due to an issue in the consumer application). Thus, the consumer must delete the message from the queue after receiving and processing it.

SQS Dead Letter Queues

- Amazon SQS supports *dead-letter queues*, which other queues (*source queues*) can target for messages that can't be processed (consumed) successfully. Dead-letter queues are useful for debugging your application or messaging system because they let you isolate problematic messages to determine why their processing doesn't succeed
- For example, if the source queue has a redrive policy with maxReceiveCount set to 5, and the consumer of the source queue receives a message 6 times without ever deleting it, Amazon SQS moves the message to the dead-letter queue.

SQS Long and Short Polling



When you consume messages from a queue using short polling, Amazon SQS samples a subset of its servers (based on a weighted random distribution) and returns messages from only those servers.

Short Polling is set by setting WaitTimeSeconds parameter to 0
or ReceiveMessageWaitTimeSeconds property of the queue to 0.

SQS Long Polling

- Eliminate empty responses by allowing Amazon SQS to wait until a message is available in a queue before sending a response. Unless the connection times out, the response to the ReceiveMessage request contains at least one of the available messages, up to the maximum number of messages specified in the ReceiveMessage action.
- Eliminate false empty responses by querying all—rather than a subset of—Amazon SQS servers. There in reducing cost.
- Return messages as soon as they become available.