

# 메시지 큐를 제공하는 SQS

2023년 6월 4일 일요일 오후 4:17

SQS(Simple Queue Service)는 서버들끼리 사용할 수 있는 메시지 큐를 제공하는 서비스입니다.

서비스가 점점 커질수록 서버 한 대로는 처리가 힘들어집니다. 자연스럽게 각 기능들을 여러 서버에서 처리하게 됩니다. 하지만, 서버들끼리 주고 받는 메시지를 잃어버리지 않고 정확하게 처리하는 것은 매우 까다로운 기술입니다.

SQS는 서버들끼리 주고 받는 메시지를 정확하게 처리해줍니다. 이런 시스템을 개인이나 벤처기업 또는 스타트업과 같은 소규모 사업장에서 구현 및 구축하는 것은 상당한 노력과 시간, 비용이 소모됩니다.

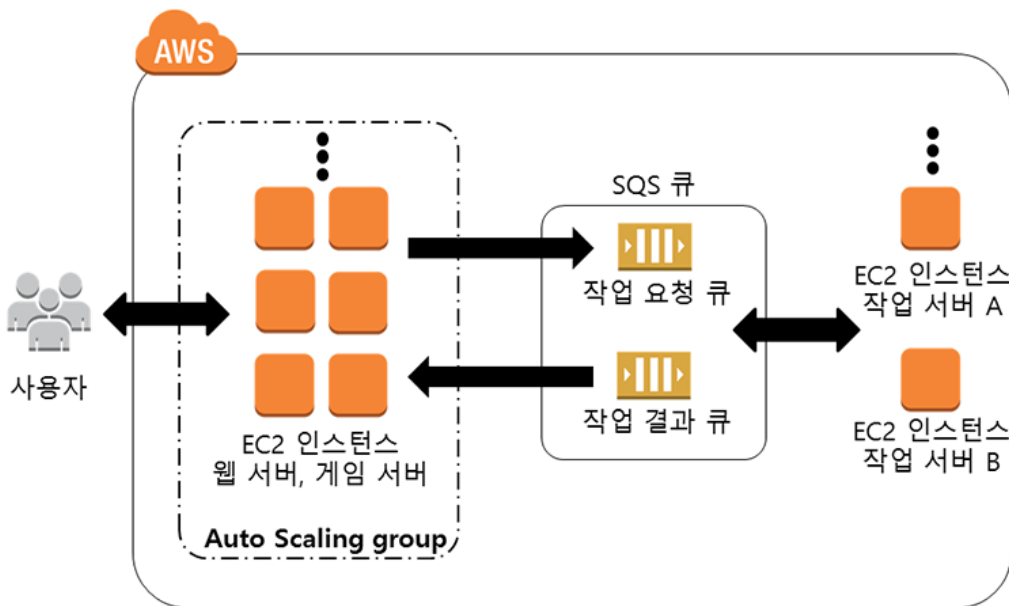
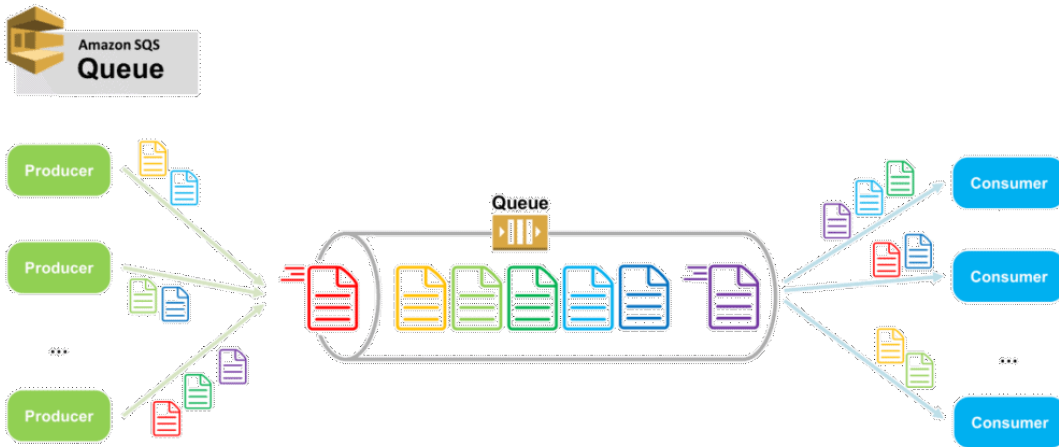
SQS를 사용하면 고가용성 메시지 큐 시스템 구축에 드는 비용과 고가용성과 신뢰성을 유지하기 위해 지속적으로 소모되는 노력과 비용도 절감할 수 있습니다. 특히 메시지 큐 시스템의 장애로 인해 발생하는 금전적인 손실도 막을 수 있습니다.

얼핏 보면 AWS에서 가장 먼저 생긴 리소스는 EC2일 것 같은데 SQS가 가장 먼저 생겼습니다. AWS는 태생적으로 모든 리소스가 아마존닷컴을 위해 개발되었는데 그중 SQS의 메시지 큐는 대형 인터넷 쇼핑몰에서 가장 중요한 기능이기 때문입니다.

예를 들면 아마존닷컴은 대규모 서비스이기 때문에 상품을 보여주는 웹서버, 상품 정보를 저장하는 데이터베이스, 결제를 처리하는 서버, 배송을 처리하는 서버 등 여러 부분으로 분리되어 있습니다. 사용자가 상품을 주문하고 결제한 뒤 웹 서버가 중단되더라도 결제 정보는 잃어버리지 않고 정확히 처리되어야 합니다. SQS는 메시지를 손실 없이 정확하게 처리하고, 고가용성을 제공하기 때문에 대규모 서비스를 구축할 때 필수적인 시스템입니다.

작동 방식은 아주 간단합니다. 한 쪽에서 SQS 큐에 메시지를 넣고 다른 한 쪽에서 메시지를 꺼내보면 됩니다.

## [ SQS 기본 아키텍처 ]



요약하자면 사용자에게 결과를 빨리 보여줘야 하는 작업과 시간이 오래 걸리는 작업을 분리할 때, 중요한 작업과 중요하지 않은 작업을 분리할 때 SQS 큐를 유용하게 사용할 수 있습니다.

다음은 SQS의 기본 개념입니다.

#### 메시지: SQS의 기본 데이터 단위입니다

- 메시지는 XML, JSON과 같은 텍스트 형태이며 최대 64KB까지 보낼 수 있습니다.
- 유니코드 문자를 사용할 수 있습니다.
- 보관 기간을 초 단위로 설정 할 수 있습니다. 기본 보관 기간은 345,600초(4일)이며 60초(1분)부터 1,209,600초(14일)까지 설정 할 수 있습니다. 그리고 메시지 보관 기간이 지나면 자동으로 삭제됩니다.
- 메시지마다 고유한 ID가 부여됩니다.
- 3~4KB짜리 메시지라도 64KB로 책정됩니다. 따라서 용량이 작은 메시지를 자주 처리하는 것보다 메시지를 모아서 배치 API로 처리하면 요금을 절약할 수 있습니다.

#### 큐(Queue): 메시지를 담을 공간입니다.

- 리전별로 생성해야 하며 HTTP 프로토콜을 이용하여 다른 리전끼리도 메시지를 주고 받을 수 있습니다. 그러므로 큐의 이름은 모든 리전에서 유일해야 합니다.
- 담을 수 있는 메시지의 개수는 무제한입니다.
- 연속 30일 동안 아무 요청(메시지 보내기, 받기 등)이 발생하지 않으면 AWS가 큐를 삭제할 수 있으므로 설계 단계에서 이 부분을 고려해야 합니다.
- 다양한 접근 권한과 정책을 설정할 수 있으며 AWS 계정 번호를 이용하여 다른 AWS 계정과도 큐를 공유할 수 있습니다.
- 같은 리전 안에서는 데이터 전송이 무료이며 다른 리전에 있는 큐나 EC2 인스턴스와 메시지를 주고 받으면 데이터 요금이 부과됩니다.
- 큐 생성 개수는 무제한 입니다.

#### 배치(Batch) API

- 한 번에 메시지를 최대 10개 혹은 최대 256KB까지 동시에 처리할 수 있습니다. 여러 메시지를 합쳐서 64KB 이하일 때 배치 API를 이용하면 요청 1개로 청구됩니다.

#### 표시 제한 시간(Visibility Timeout)

- 메시지를 받은 뒤 특정 시간 동안 다른 곳에서 동일한 메시지를 다시 꺼내볼 수 없게 하는 기능입니다.
- 0초부터 12시간까지 설정할 수 있습니다. 큐 하나에 여러 서버가 메시지를 받을 때 동일한 메시지를 동시에 처리하는 것을 방지합니다.
- Messages Available(Visible): 내용을 꺼내서 볼 수 있는 상태인 메시지 개수입니다.
- Messages in Flight(Not Visible): 다른 곳에서 메시지를 보고 있어서 현재는 내용을 볼 수 없는 상태인 메시지 개수입니다. 최대 120,000개까지이며 최대치를 넘어서면 에러(OverLimit)가 발생합니다.

#### 지연 전송(Delay Delivery)

- 특정 시간 동안 메시지를 받지 못하게 하는 기능입니다. 지연 되는 시간 동안에는 Messages in Flight에 포함됩니다.

#### 처리 실패 큐(Dead Letter Queue)

- 보통 메시지를 받고 작업이 처리되면 메시지를 삭제합니다. 하지만, 설정한 횟수를 초과하여 메시지를 받았는데 삭제되지 않고 남아 있다면 처리 실패 큐로 보냅니다.
- 예를 들어 사용자가 특정 제품 ID에 웹 주문을 하는데 해당 제품 ID가 삭제된 경우, 웹 스토어의 코드가 실패하고 오류가 표시되며 해당 주문 요청이 포함된 메시지가 배달 못한 편지 대기열로 전송됩니다.
- 메시지를 받는 횟수는 1번부터 1,000번까지 설정할 수 있습니다.
- 일반 큐 하나에 여러 개의 처리 실패 큐를 연결할 수 있습니다.
- 처리 실패 큐는 일반 큐와 같은 리전에 생성해야 합니다.

#### 짧은 폴링(Short Polling)

- 메시지 받기 요청을 하면 결과를 바로 받습니다. 메시지가 있으면 메시지를 가져오고 없으면 그냥 빠져나옵니다.
- ReceiveMessage 요청에서 WaitTimeSeconds를 0을 설정했을 때
- 큐 설정이 ReceiveMessageWaitTimeSeconds를 0으로 설정했을 때

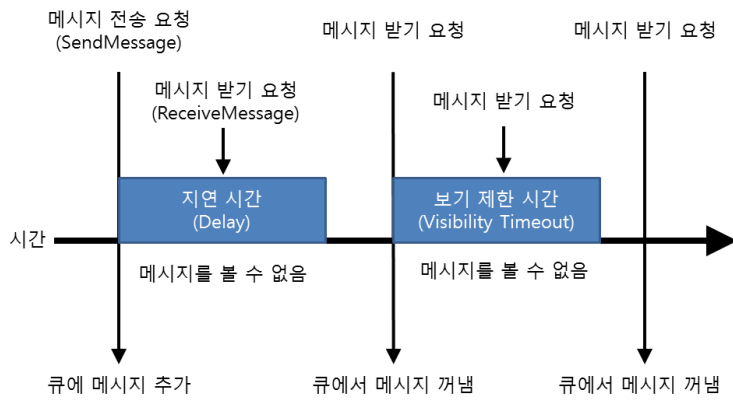
#### 긴 폴링(Long Polling)

- 메시지가 있으면 바로 가져오고, 메시지가 없으면 메시지가 올 때까지 기다립니다. 또는 메시지가 계속 오지 않으면 긴 폴링 제한 시간까지 기다립니다. 기본 제한 시간은 20초이며 1초부터 최대 20초까지 설정할 수 있습니다.
- ReceiveMessage 요청의 WaitTimeSeconds가 0보다 크면 큐 설정의 ReceiveMessageWaitTimeSeconds 값보다 높은 우선순위를 갖습니다.

#### 요금

- 64KB당 요청 1개로 청구됩니다. 256KB 짜리 배치 API를 한번 호출하면 요청 4개로 청구됩니다.

아래 그림은 지연 전송과 표시 제한 시간을 그래프로 나타낸 것입니다.



지연 전송을 하면 메시지를 보낸 뒤에 일정 시간 동안 메시지를 받을 수 없고, 보기 제한 시간(표시 제한 시간)을 설정하면 메시지를 받은 뒤 일정 시간 동안에는 다른 곳에서 메시지를 받을 수 없습니다.

SQS 함수의 종류는 다음과 같습니다.

- CreateQueue: 큐 생성
- ListQueues: 큐 목록 출력
- DeleteQueue: 큐 삭제
- SendMessage: 큐에 메시지 추가
- SendMessageBatch: 큐에 여러 메시지 추가
- ReceiveMessage: 큐에서 메시지를 꺼내서 보기
- ChangeMessageVisibility: 메시지의 보기 제한 시간 변경
- ChangeMessageVisibilityBatch: 여러 메시지의 보기 제한 시간 변경
- DeleteMessage: 큐에서 메시지 삭제
- DeleteMessageBatch: 큐에서 여러 메시지 삭제
- SetQueueAttributes: 큐 설정 변경(지연 전송 시간, 최대 메시지 크기, 메시지 보관 기간, 접근 정책, 짧은 폴링 또는 긴 폴링 시간, 처리 실패 큐)
- GetQueueAttributes: 큐 설정 확인(메시지 개수, 큐 생성 시간, 최종 큐 변경 시간, 큐 ARN, 지연 전송 시간, 최대 메시지 크기, 메시지 보관 기간, 접근 정책, 짧은 폴링 또는 긴 폴링 시간, 처리 실패 큐)
- GetQueueUrl: 큐 엔드포인트 URL 확인
- AddPermission: 다른 AWS 계정에 대한 접근 권한 설정
- RemovePermission: 다른 AWS 계정에 대한 접근 권한 삭제