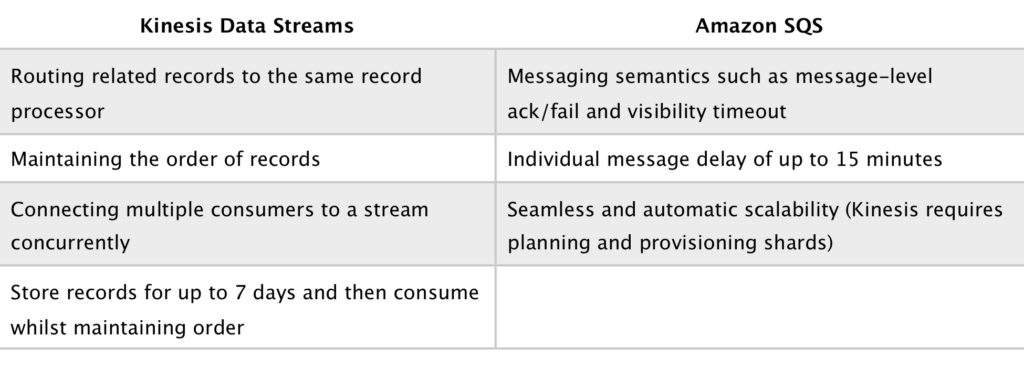
**SQS**

* SQS is a web service that gives you access to message queues that store messages waiting to be processed
* SQS offers a reliable, highly-scalable, hosted queue for storing messages in transit between computers
* SQS is used for distributed/decoupled applications
* SQS can be used with RedShift, DynamoDB, EC2, ECS, RDS, S3 and Lambda
* SQS uses a message-oriented API
* SQS uses pull based(polling) not push based
* Messages are 256KB in size
* Messages can be kept in the queue from 1 minute to 14 days (default is 4 days)
* The visibility timeout is the amount of time a message is invisible in the queue after a reader picks up the message
* If a job is processed within the visibiliyt timeout the message will be deleted
* If a job is not processed within the visibility timeout the message will become visible again(could be delivered twice)
* The maximum visibility timeout for an SQS message is 12 hours
* SQS message can contain up to 10 metadata attributes
* 
* Polling
  + SQS uses short polling and long polling
  + Short polling:
    - Does not wait for messages to appear in the queue
    - It queries only a subset of the available servers for messages(based on weighted random execution)
    - Short polling is the default
    - ReceiveMessageWaitTime is set to 0
    - More requests are used, which implies higher cost
  + Long polling:
    - Uses fewer requests and reduces cost
    - Eliminates false empty responses by querying all servers
    - SQS waits until a message is available in the queue before sending the response
    - Rrequests contain at least one of the available messages up to the maximum number of messages specified in the ReceiveMessage action
    - Shouldn’t be used if your application expects an immediate response to receive message calls
    - ReceiveMessageWaitTime is set to a non-zero value(up to 20 secodnds)
    - Same charge per million requests as short polling
* Queues
  + Queue names must be unique within a region
  + Queues can be either standard or FIFO
  + Standard queues provide a loose-FIFO capability that attempts to preserve the order of messages
  + Because standard queues are designed to be massively scalable and using a highly distributed architecture, receiving messages in the exact order they are sent is not guarnateed
  + Standard queues provide at-least-once delivery, which means that each message is delivered at least once
  + FIFO queues preserve the exact order in which messages are sent and received
  + FIFO queues are available in limited regions currently
  + If you use a FIFO queue you don’t have to place sequencing information in your message
  + FIFO queues provide exactly-once processing, which means that each message is delivered once and remains available until a consumer processes it and deletes it
  + Graphical user interface, text, application

    Description automatically generated
* **Limits**
  + In-flight messages are messages that have been picked up by a consumer but not yet deleted from the queue
  + Standard queues have a limit of 120,000 in-flight messages per queue
  + FIFO queues have a limit of 20,000 in-flight messages per queue
  + Queue names can be up to 80 cahracters
  + Messages are retained for 4 days by default and up to 14 days
  + FIFO queues support up to 3000 messages per second when batching or 300 per second otherwise
  + The maximum message size is 256KB
* **Scalability and Durability**
  + You can have multiple queues with different priorities
  + Scaling is performed by creating more queues
  + SQS stores all message queues and messages within a single, highly-available AWS region with multiple redundant Azs
* **Security**
  + You can use IAM policies to control who can read/write messages
  + Authentication can be used to secure messages within queues(who can send and receive)
  + SQS supports HTTPS and supports TLS versions 1.0, 1.1, 1.2
  + SQS and PCI DSS level 1 compliant and HIPAA eligibible
  + Server-side(SSE) Encryption lets you transimt sensitive data in encrypted queues)AWS KMS):
    - SSE encrypts messages as soon as SQS receives them
    - The messages are stored in encrypted form and SQS decrypts messages only when they are sent to an authorized consumer
    - Uses AES 256 bit encryption
    - Not available in all regions
    - Standard and FIFO queues
    - Body of the message is encrypted
    - Following is not encrypted:
      * Queue metadata
      * Message metadata
      * Per-queue metrics
* **Monitoring**
  + CW is integrated with SQS and you can view and monitor queue metrics
  + **CW metrics are automatically collected every 5 minutes**
  + CW considers a queue to be active for up to 6 hours if it contains any messages or if any API action accesses it
  + No charge for CW(no detailed monitoring)
  + CT captures API calls from SQS and logs to a specified S3 bucket
  + 