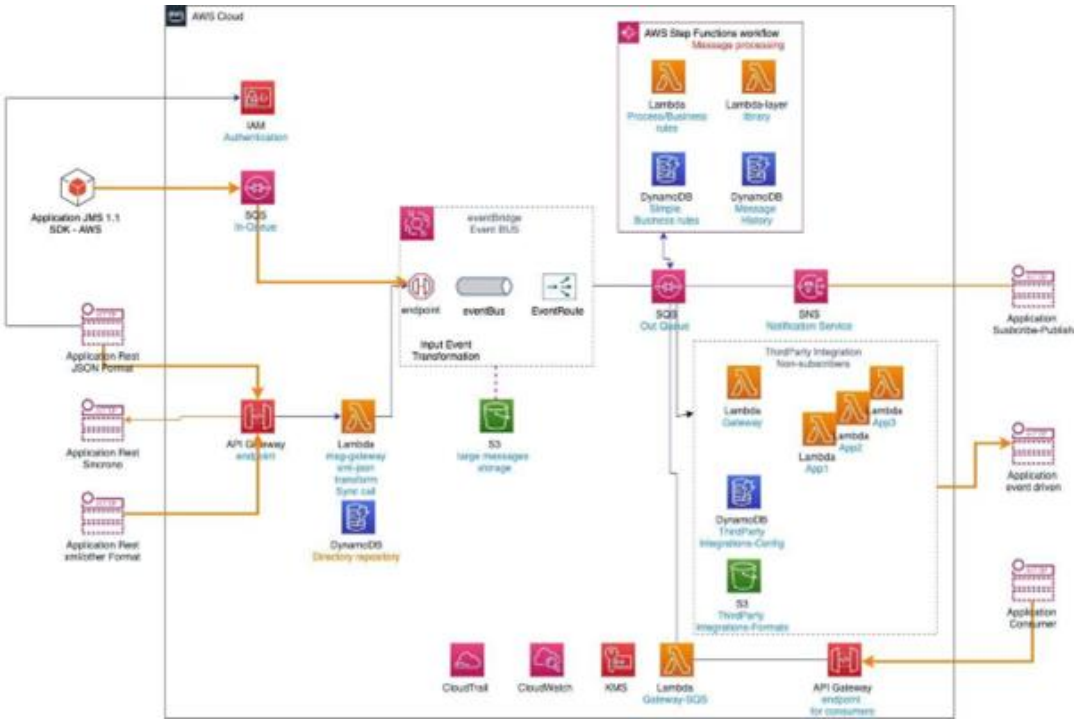
ANEXO 1 - Detalle Message Broker

El siguiente texto fue tomado de un blog post del Proveedor de Servicios de Nube AWS: https://aws.amazon.com/es/blogs/aws-spanish/integrando-aplicaciones-durante-la-modernizacion/ :

A continuación, se presenta un diagrama arquitectura como único punto de integración usando servicios de AWS, el cual se explicará con base en las necesidades de integración de microservicios y monolitos



Esta nueva arquitectura, a nivel de microservicios, permite:

- Cortos tiempo de entrega al mercado (TTM)
- Autonomía en el desarrollo de los microservicios
- Desacople (a nivel de fallas y tiempos de respuesta)
- Integraciones síncronas y asíncronas
- Seguridad de la información (ej. encriptación en tránsito y descanso)
- Alta flexibilidad
- Orquestación de mensajes.

A nivel de integración de aplicaciones empresariales (monolitos), la arquitectura debe cumplir con los diferentes patrones “patterns” que definen las funcionalidades necesarias para lograr ser el centro de flujo de información entre las aplicaciones. Tomando como referencia “Enterprise Integration Patterns (EIP)” https://www.enterpriseintegrationpatterns.com/patterns/messaging/ para el propósito de este blog, se compara las capacidades de la arquitectura contra los patrones de integración empresarial:

Categoría / Pattern category	Patrón / Patterns	Contempla en la arquitectura	Explicación corta
Messaging Endpoints	Messaging Endpoint	SI	En la arquitectura se plantean endpoint síncronos y asíncronos: 1) Amazon API Gateway (Expone Rest, https y websocket), 2) JMS con Amazon SQS (Simple Queue Service) y 3) Amazon SNS permite suscripción/publicación
	Messaging Gateway	SI	Todos los mensajes son enviados a Amazon EventBridge independiente del endPoint usado. Cuando el EndPoint usado es Amazon API Gateway, es posible configurar algunas acciones por defecto, como publicar la estructura del API.
	Messaging Mapper	SI	Amazon EventBridge puede seleccionar los mensajes que cumplan con ciertos criterios para luego iniciar un AWS step function y este puede mover datos entre dominios si el contenido del mensaje está en formato JSON, en caso de ser otro formato, AWS StepFunctions deberá orquestar un Lambda para que haga este movimiento
	Transactional Client	NO	
	Polling Consumer	SI	Cuando una aplicación requiere consumir los mensajes vía REST y HTTPS se debe conectar al API Gateway e indicar qué método necesita y recibirá el mensaje a demanda extraído de una de las colas de salida (Amazon SQS) por medio de una función de AWS Lambda.
	Event-driven Consumer	SI	Cuando la aplicación está lista para recibir los mensajes, tiene 4 formas de hacerlo, vía JMS conectado a la cola de salida Amazon SQS, vía SDK de la cola de salida de Amazon SQS, vía Lambda se consume un Web Service publicado por la aplicación o se suscribe a un tema en Amazon SNS para recibir el mensaje cada vez que es publicado
	Competing Consumers	SI	Cuando la aplicación destino no puede procesar los mensajes tan rápido como los genera el origen, es posible crear múltiples colas FIFO de salida (Amazon SQS) para que el destino pueda procesar los mensajes en paralelo. Si el destino no tiene la capacidad de procesar los mensajes en paralelo, es posible forzar la entrega del siguiente mensaje solo si el anterior fue completamente procesado usando AWS Stepfunctions
	Message Dispatcher	SI	Cuando el mensaje tiene que ser procesado por varios destinos, en cierto orden o coordinación, Amazon EventBridge es capaz de iniciar un AWS Step Function, y este se encargará de orquestar el orden y condiciones de ejecución de las entregas de los mensajes por medio de múltiples colas de salida (Amazon SQS)
	Selective Consumer	PARCIAL	La aplicación que consume, no puede seleccionarlos mensajes, pero si la cola de salida (Amazon SQS) de la cual obtiene los mensajes. A partir del manejo de múltiples colas de salida, es posible segmentar los mensajes que el consumidor recibe
	Durable Subscriber	SI	Cuando el suscriptor de mensajes no está escuchando, él podrá cuando retorna extraer el

			mensaje de la cola de salida (Amazon SQS) en cualquier momento (hasta 14 días). Tener en cuenta que existe una cola para mensajes no entregados “Amazon SQS dead letter queue» que permite reprocesar la información más adelante.
	Idempotent Receiver	SI	Si el mensaje se genera varias veces, pero se debe entregar solo una vez, es posible crear reglas de de duplicación de mensajes en AWS lambda, coordinadas por AWS Step Function y basadas en los históricos almacenados
	Service Activator	NO	
Message Construction	Message	SI	El canal lo componen dos servicios: Amazon EventBridge (event bus) y Amazon SQS (administración de colas). El mensaje es enviado por el generador a alguno de los endpoint de entrada, luego ingresa a Amazon Event Bridge que de acuerdo a las reglas de enrutamiento lo envía a una cola de salida (Amazon SQS) de donde es entregado al receptor vía alguno de los endpoint de la arquitectura o enviado (vía AWS Lambda) al destinatario
	Command Message	SI	En la arquitectura los mensajes de tipo comando se manejarán como un cualquier mensaje, dependerá del receptor interpretarlo como comando.
	Document Message	SI	Nativamente Amazon Event Bridge solo puede transportar mensajes hasta 256 KB incluido estructura, en el caso de mensajes de mayor tamaño, serán almacenados de manera temporal en Amazon S3, y se transportará vía Amazon EventBridge y Amazon SQS solamente los identificadores y el encabezado de los mensajes.
	Event Message	SI	Cuando el mensaje tiene que ser procesado por uno o varios destinos, Amazon EventBridge es capaz de iniciar un AWS Step Function, y este se encargará de entregar el mensaje a varias colas (Amazon SQS) las cuales pueden ser “standard» o «FIFO»
	Request-Reply	PARCIAL	Cuando la aplicación que genera el mensaje requiere una respuesta de la aplicación receptora (integración síncrona), vía HTTPS o REST, es posible que API Gateway entregue una respuesta genérica e indicando donde la aplicación generadora puede buscar la respuesta. También es posible que vía AWS Lambda se consuma un WebService y se envía la respuesta directamente
	Return Address	PARCIAL	Cuando hay varios generadores de mensajes con un mismo destino, el mensaje de respuesta se enrutará gracias al almacenamiento histórico en Amazon DynamoDB en donde se podrá identificar el origen del mensaje y enrutar la respuesta a la cola (Amazon SQS) correspondiente
	Correlation Identifier	SI	Para lograr la correlación entre mensajes y respuestas, se usarán dos identificadores, el primero es originID que debe ser parte del mensaje enviado por aplicación generador y el otro es “RequestID», es identificador único asignado al mensaje cuando este es recibido por

			el endPoint. Estos dos IDs acompañaran al mensaje mientras viajan en la arquitectura
	Message Sequence	PARCIAL	Cuando la aplicación receptora no es capaz de procesar mensajes de gran tamaño, Amazon EventBridge puede iniciar un AWS Step Function que coordine funciones AWS Lambdas para dividir el mensaje en unos más pequeños y entregarlos a la cola correspondiente (Amazon SQS). Estos mensajes deberán conservar el «RequestID» y el originID del mensaje original, igualmente se deberá indicar que el mensaje fue dividido y un identificador de la secuencia de la cual está compuesto.
	Message Expiration	PARCIAL	Aunque el generador del mensaje no puede establecer el vencimiento (expiration) de cada mensaje, es posible configurar el vencimiento en la configuración de la cola de salida (Amazon SQS) de tal manera que, si el mensaje no es tomado/eliminado de la cola, se mueve a la cola «dead-letter queue»
	Format Indicator	PARCIAL	Cuando el generador del mensaje requiere indicar cual es el formato con el cual debe ser tratado el mensaje. Amazon EventBridge recibe todos los mensajes, existen unas reglas que permiten filtrar e iniciar un Amazon Step Function, este podría usar diferentes formatos para procesar el mensaje de pendiendo de un valor proveniente en el mensaje original
Messaging Channels	Message Channel	SI	El canal lo componen dos servicios Amazon EventBridge (event bus) y Amazon SQS (cola de salida). El mensaje es enviado por el generador a alguno de los endpoint de entrada, luego ingresa a Amazon Event Bridge que de acuerdo a las reglas de enrutamiento lo envía a una cola de salida (Amazon SQS) de donde es entregado al receptor vía alguno de los endpoint de la arquitectura o enviado (vía AWS Lambda) al destinatario directamente
	Point to Point channel	SI	Es posible garantizar que solo un destinatario reciba el mensaje cuando Amazon EventBridge enruta el mensaje solo a una cola de tipo FIFO en Amazon SQS.
	Publish-Subscribe Channel	PARCIAL	Los receptores de los eventos pueden suscribirse a un tema «topic» en el servicio Amazon SNS de manera que reciban el mensaje cuando se realice una publicación. En la arquitectura es posible realizar esto cuando la cola de salida (Amazon SQS) publica el mensaje en el tema “topic» de Amazon SNS.
	Datatype Channel	SI	Cuando el receptor recibe múltiples mensajes del mismo generador (origen) pero los mensajes (tipo / estructura) son diferentes, puede diferenciarlos gracias a la cola de salida (Amazon SQS) en la cual Amazon EventBridge lo envió.
	Invalid messages channel	SI	Cuando AWS Lambda envía un mensaje a un receptor, este puede decidir si el mensaje es válido o no, en la función sería necesario configurar un procedimiento para mover el

			mensaje a una cola de mensajes inválidos (Amazon SQS)
	Dead letter channel	SI	Cuando un mensaje no puede ser enviado o no es reclamado por el receptor desde la Cola (AmazonSQS), es movido a una cola llamada Dead letter queue.
	Guaranteed Delivery	PARCIAL	En la arquitectura es posible persistir en la entrega del mensaje solo cuando el mensaje ya se encuentra en Step Function. La arquitectura basa la garantía de entrega en el compromiso de disponibilidad de SQS que es del 99,9%.
	Channel adapter	SI	En la arquitectura se plantean endpoint síncronos y asíncronos: 1) Amazon API Gateway (Expone Rest, https y websocket), 2) JMS con Amazon SQS (Simple Queue Service) y 3) Amazon SNS permite suscripción/publicación. Si el mensaje debe ser entregado vía un API o WebService es posible enviarlo usando una función «AWS Lambda»
	Messaging Bridge	NO	
	Message Bus	SI	En la arquitectura el servicio Amazon EventBridge se comporta como un bus, gracias a su componente "EventBus». Es importante aclarar que se hace una analogía entre eventos y mensajes.
Message Routing	Pipes and Filters	SI	Es posible ejecutar tareas complejas sobre el mensaje, cuando Amazon EventBridge inicia un AWS step function, es posible orquestar la ejecución de funciones AWS Lambda que realicen actividades complejas como desencripción, deduplicación, transformación, filtrados, etc.
	Message Router	SI	Usando las reglas de enrutamiento de Amazon Event Bridge es posible enviar mensajes a diferentes colas de salida o incluso a AWS StepFunction para su procesamiento.
	Content-Based Router	SI	Las reglas de enrutamiento de Amazon EventBridge puede filtrar los mensajes basados en la información de los mismos. Si la información que permite decidir el destino está en un formato diferente a JSON, será necesario realizar la conversión a JSON.
	Message Filter	SI	La arquitectura presenta dos niveles de filtrado para los mensajes entrantes: 1. Amazon API Gateway filtrará los mensajes malformados o con problemas de protocolo. 2. Amazon Event Bridge filtrará los mensajes que no cumplan con ninguna regla.
	Dynamic Router	NO	
	Recipient List	PARCIAL	Una vez entregado el mensaje por Amazon EventBridge a AWS Step Function, se construye una lista de receptores a quienes se les envía el mensaje. Si el número de destinatarios es menor a 5, es posible que Amazon EventBridge envíe directamente el mensaje usando 5 colas (Amazon SQS)
	Splitter	SI	Una vez Amazon EventBridge entrega el mensaje a AWS Step Function, se puede orquestar la división del mensaje original en unos mensajes que solo contengan un fragmento de la

			información original, para que finalmente cada uno sea a una cola diferente. (Amazon SQS)
	Aggregator	PARCIAL	Usando AWS Step Functions es posible orquestar funciones AWS Lambda para unir la información del mensaje con información proveniente de otra fuente. Cuando se desea unir la información de dos mensajes, se usará Amazon DynamoDB/ Amazon S3 para almacenar de manera temporal los eventos y así poder hacer operaciones entre varios mensajes
	Resequencer	NO	
	Composed Message Processor	SI	Basado en el patrón “splitter» (dividir y enrutar) y el patrón «Aggregator” (unir varios mensajes), es posible procesar varios mensajes de manera compleja y paralela usando AWS Lambdas orquestados por AWS Step Functions, y finalmente unirlos para entregar un solo mensaje al destino.
	Scatter-Gather	SI	Usando Amazon EventBridge, Amazon SQS y Amazon SNS es posible entregar un mensaje a múltiples destinos en paralelo, y en el caso de este patrón es opcional de cada destino dar respuesta. Usando AWS Step Functions es posible esperar la respuesta de los destinos y agruparla en un solo mensaje de respuesta para el generador inicial.
	Routing Slip	PARCIAL	Usando las reglas de Amazon Event Bridge es posible enviar el mensaje a diferentes AWS StepFunctions quienes orquestarán AWS Lambdas de manera que el procesamiento del mensaje se asimile a los pasos en un flujo de trabajo. El flujo se basa en condiciones de acuerdo a los datos del mensaje.
	Process Manager	SI	En la arquitectura quien ejecuta el patrón de «Process Manager» es la combinación entre las reglas de Amazon EventBridge y las máquinas de estado de AWS Step Functions.
	Message Broker	SI	Usando las reglas de Amazon Event Bridge y las colas de Amazon SQS es posible manejar, enrutar y desacoplar la comunicación entre origen y destino (mensajes)
System Management	Control Bus	SI	AWS tiene una consola de administración unificada donde es posible configurar todos los servicios presentados en la arquitectura. Para controlar y visualizar el estado de los mensajes, su procesamiento y las colas, es posible usar la consola de AWS Step Function y los servicios Amazon CloudWatch y AWS CloudTrail
	Detour	SI	Usando las reglas de Amazon EventBridge es posible desviar los mensajes a una ruta alterna donde se realiza el seguimiento o algún tipo de control, finalmente se puede enviar a la cola de salida o alguna cola de prueba (Amazon SQS)
	Wire Tap	SI	Para los mensajes que solo pueden tener un destino y se debe entregar una sola vez, se pueden usar Amazon Event Bridge y 2 colas FIFO (Amazon SQS) para garantizar una sola entrega al destino y al mismo tiempo atrapar una copia de los mensajes para tareas de depuración o análisis

	Message History	SI	Todos los mensajes deberán ser almacenados en Amazon DynamoDB de manera temporal, se habilitarán las funcionalidades de backup y se ejecutará limpieza periódica. En caso de mensajes de gran tamaño, el contenido será almacenado en Amazon S3 y solo los identificadores y el encabezado serán almacenados en Amazon DynamoDB
	Message Store	SI	Todos los mensajes deberán ser almacenados en Amazon DynamoDB de manera temporal, se habilitarán las funcionalidades de backup y se ejecutará limpieza periódica. En caso de mensajes de gran tamaño, el contenido será almacenado en Amazon S3 y solo los identificadores y el encabezado serán almacenados en Amazon DynamoDB
	Smart Proxy	PARCIAL	Una vez el mensaje llega a la arquitectura, se obtiene la información del generador del mensaje y luego usando AWS Step Función y AWS lambda se entrega la respuesta al mismo origen. De igual manera, es posible que el mensaje de respuesta contenga la dirección a quien se desea entregar el mensaje o cola de salida en la cual se debe dejar el mensaje.
	Test message	SI	Es posible inyectar mensajes de pruebas directamente en el endpoint de entrada (Amazon SQS o Amazon API Gateway), se debe definir una marca (campo) en los mensajes para que estos sean enviados a las colas de salida (después de procesamiento) especialmente destinadas a monitoreo en lugar de las colas de salida de producción (Amazon SQS)
	Channel Purger	PARCIAL	En la arquitectura se presenta un canal como la combinación entre Amazon Event Bridge (Event Bus) y Amazon SQS (Queue). Vía AWS console o AWS CLI es posible purgar las colas de Amazon SQS, en caso de ser necesario garantizar que las colas estén vacías.
Message Transformation	Message Translator	SI	Amazon EventBridge puede filtrar los mensajes que requieran algún tipo de transformación e iniciar un AWS StepFunction para que orqueste funciones AWS Lambda para transformar el mensaje en el formato de destino, luego es colocado en la cola de salida (Amazon SQS). Las funciones AWS Lambda pueden ser desarrolladas por defecto en 7 lenguajes de programación.
	Envelope Wrapper	SI	Idealmente todos los mensajes deben tener 2 secciones, un encabezado y el contenido. Siendo el encabezado toda la información necesaria para identificar y clasificar los mensajes y el contenido es la carga útil del mensaje. Usando AWS KMS, se administra las llaves con las cuales se puede encriptar/desencriptar el contenido del mensaje cuando sea necesario
	Content Enricher	SI	En las reglas de Amazon EventBridge se pueden filtrar los mensajes que requieran algún tipo de enriquecimiento, estos mensajes inician un AWS Step Function, el cual orquestará una

			función AWS Lambda para enriquecer los mensajes basados en cualquier base de datos, como por ejemplo Amazon DynamoDB,
	Content Filter	SI	Usando AWS Step Functions es posible filtrar/seleccionar parcialmente el contenido del mensaje cuando está en formato JSON, en caso que el contenido del mensaje tenga otro formato, AWS Step Functions orquestará funciones AWS Lambda para filtrar/seleccionar solamente la información del mensaje original que se requiere entregar a la cola de salida (Amazon SQS)
	Claim Check	PARCIAL	Nativamente Amazon Event Bridge solo puede transportar mensajes hasta 256 KB incluido estructura, en el caso de mensajes de mayor tamaño, serán almacenados de manera temporal en Amazon S3, y se transportará vía Amazon EventBridge y Amazon SQS solamente los identificadores y el encabezado de los mensajes.
	Normalizer	SI	Amazon EventBridge puede filtrar los mensajes que requieren algún tipo de normalización e inicia un AWS Step Function para que orqueste funciones AWS Lambda y finalmente el mensaje normalizado es colocado en la cola de salida. Las funciones AWS Lambda pueden ser desarrolladas por defecto en 7 lenguajes de programación.
	Canonical Data Model	SI	En la arquitectura se plantea que los mensajes sean convertidos a formato JSON, con dos secciones, encabezado y contenido. Este formato se usará como canónico cuando la información viaje entre los servicios de AWS.