

Tecnicatura Superior en Telecomunicaciones. Año 2025

Profesor Ing. Jorge Morales.

Grupo 2 - Alumna: Emma Gutiérrez

Eje 3- Actividad Nro. 5

Protocolos Industriales

PROTOCOLO AMQP



Arquitectura

La versión 1.0 de AMQP proporciona las piezas necesarias para permitir que los clientes y servidores de mensajería interoperen sin problemas, independientemente de su tecnología.

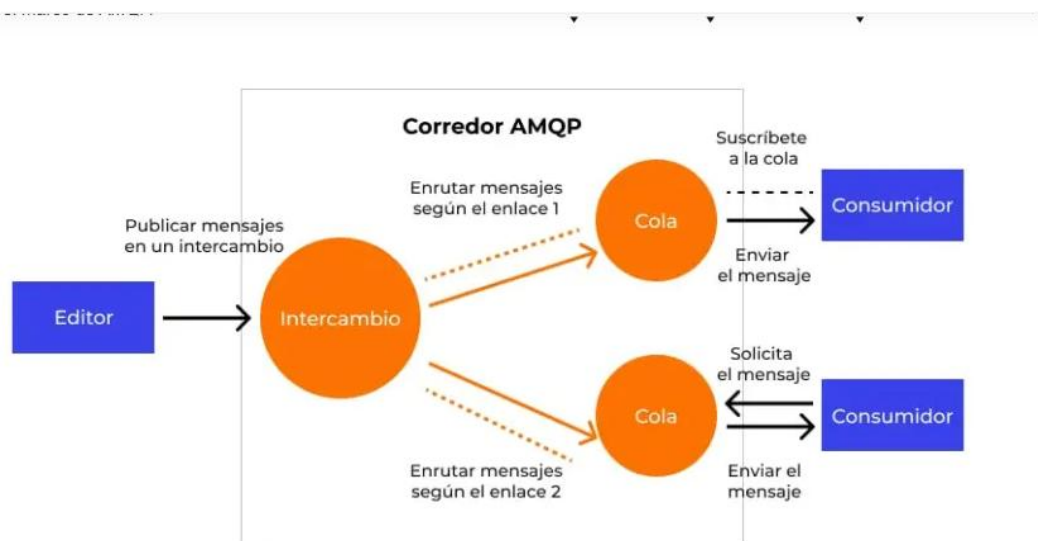


Wallarm Platform

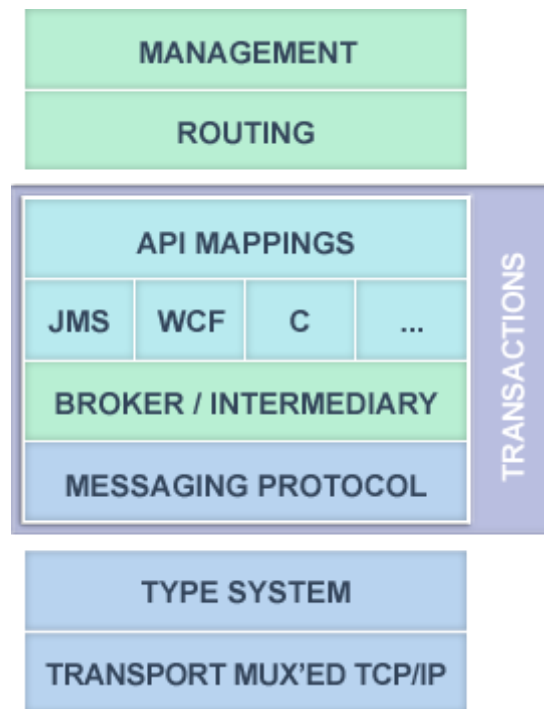
Solutions

Resources

Learn Wallarm



Consiste en un protocolo de cable eficiente que separa el transporte de red de las arquitecturas de intermediario y su gestión. **AMQP versión 1.0** admite diversas arquitecturas de intermediario que pueden utilizarse para recibir, poner en cola, enrutar y entregar mensajes, o para su uso peer-to-peer. El alcance de **AMQP 1.0** se especifica en tres partes principales: el **protocolo de red**, la **representación de los datos del sobre del mensaje** y la semántica básica **de los servicios de intermediario**.



Transporte (TCP/IP)

- Transmisión de bytes ordenada y confiable, y control del flujo de red
- Subdivisión de TCP/IP en *canales*

Protocolo de mensajería y sistema de tipos

- Representación portátil de las propiedades del mensaje
- Transporte de mensajes entre pares de transporte
- Transferencia de responsabilidad por los mensajes entre pares
- Control de flujo de mensajes

Message Broker – la capa MOM

- Intermediación, separación temporal de aplicaciones de mensajería
- Almacenamiento seguro de mensajes
- Recursos transaccionales
- “Direcciones” locales
- Funciones de distribución de mensajes; colas compartidas, temas, etc.
- Colas de respuesta

Mapeos de API

- Representación canónica de conceptos entre API
- Enviar JMS <> WCF <> C <> Python ...

Enrutamiento entre corredores: en una próxima incorporación

- Separación política de las aplicaciones de mensajería
- Reenvío de mensajes
- Traducción de dirección hacia adelante/hacia atrás
- Propagación de suscripciones

Gestión –en una próxima incorporación

- Gestión estandarizada de cualquier bróker AMQP
- Envía mensajes de gestión de AMQP al bróker para efectuar cambios
- Aproveche todas las funciones de enrutamiento y seguridad de Broker and Routing

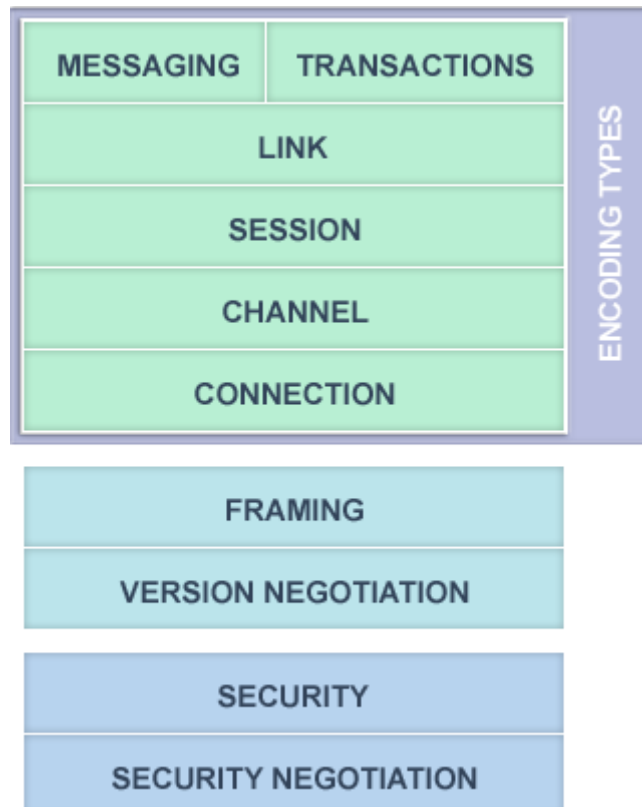
El protocolo de red AMQP

El protocolo de red AMQP define:

- Un protocolo peer to peer; aunque normalmente en AMQP un peer desempeña el papel de una aplicación cliente y el otro peer desempeña el papel de un servicio de enrutamiento y entrega de mensajes confiable, o intermediario.
- Cómo conectarse a servicios, incluido un método para conmutar por error las conexiones a servicios alternativos
- Un mecanismo para permitir que los pares descubran las capacidades de los demás
- Mecanismos de seguridad integrales, incluidos SSL y Kerberos para una confidencialidad perfecta de extremo a extremo
- Cómo multiplexar una conexión TCP/IP para que múltiples conversaciones puedan tener lugar en una sola conexión. Esto simplifica enormemente la gestión del firewall.
- Cómo direccionar una fuente de mensajes con el par de la red y especificar qué mensajes son de interés
- El ciclo de vida de un mensaje: desde la obtención, el procesamiento y la confirmación. AMQP deja muy claro cuándo se transfiere la responsabilidad de un mensaje de un par a otro, lo que mejora la confiabilidad.
- Cómo mejorar el rendimiento, si se desea, recuperando con antelación mensajes en la red, listos para que el cliente los procese sin demora
- Una forma de procesar lotes de mensajes dentro de una transacción
- Un mecanismo para permitir una transferencia completa de mensajes desde el inicio hasta el cierre de sesión en un paquete de red para aplicaciones livianas
- Control de flujo muy capaz, que permite a los consumidores de mensajes reducir la velocidad de los productores a un nivel manejable, y que permite que distintas cargas de trabajo procedan en paralelo a distintas velocidades en una conexión.
- Mecanismos para reanudar las transferencias de mensajes cuando se pierden y se restablecen las conexiones; por ejemplo, en caso de conmutación por error del servicio o conectividad intermitente

Este protocolo de red de bajo nivel es completo y eficaz, pero normalmente es invisible para los usuarios de software de mensajería. Al igual que el correo electrónico, los usuarios enviarán mensajes a target@example.com sin conocer los detalles de cómo los intermediarios procesan esa solicitud.

Dicho esto, es posible que los especialistas en redes utilicen este protocolo para otros fines, pero se anticipa que el mayor uso del protocolo será conectar clientes de mensajería con intermediarios de mensajería; los intermediarios que alojan colas y temas y se encargan del almacenamiento seguro, el enrutamiento y la entrega.



Representación del mensaje

AMQP proporciona portabilidad entre sistemas. Por lo tanto, necesita una forma de representar los datos empresariales necesarios para enrutar y entregar el mensaje de forma portátil, accesible desde diversos lenguajes de programación. El sistema de tipos AMQP 1.0 y las funciones de codificación de mensajes proporcionan una codificación portátil para satisfacer esta necesidad.

Normalmente, esta codificación solo se utiliza para añadir propiedades de enrutamiento al "envoltorio" del mensaje; el contenido dentro del envoltorio se transporta intacto. Es probable que las aplicaciones utilicen codificaciones XML, JSON o similares en el contenido de sus mensajes. Opcionalmente, una aplicación también podría optar por usar la codificación AMQP para el contenido del mensaje, pero esto es totalmente opcional.

Servicios de corretaje

La ventaja de usar intermediarios de mensajes reside en que un intermediario confiable, diseñado específicamente para este fin, gestiona las complejidades de la cola, el enrutamiento y la entrega de mensajes. Ese intermediario es el intermediario de mensajes.

AMQP define el conjunto mínimo de requisitos que se esperan de un agente de mensajes y, donde se utilizan frecuentemente facilidades más avanzadas, especifica cómo se deben exponer esas facilidades a los clientes.

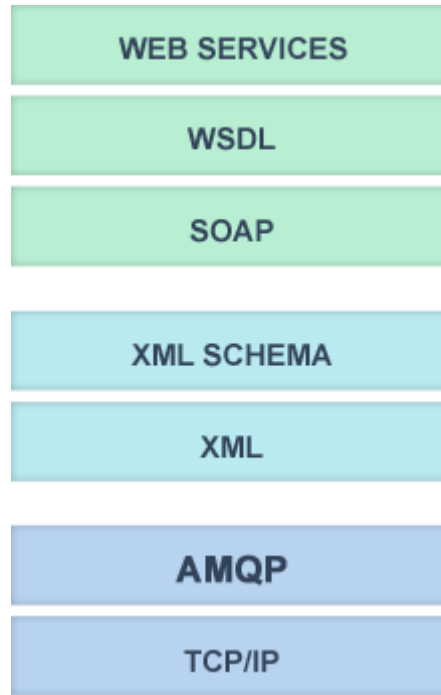
El objetivo de AMQP es permitir que las aplicaciones envíen mensajes a través de los servicios de intermediario; estos conceptos básicos son necesarios, pero no constituyen el objetivo en sí mismos. Al enviar un mensaje por mensajería, ¿no conduces el camión de reparto!

AMQP Diferenciado

AMQP es diferente de otros estándares de middleware porque es:

- **INTEROPERABLE.** Todos los clientes AMQP interoperan con todos los servidores AMQP. Diversos lenguajes de programación se comunican fácilmente. Los agentes de mensajes heredados pueden modernizarse para eliminar protocolos propietarios de su red. La mensajería puede habilitarse como servicio en la nube. Una solución sencilla para la mensajería empresarial y la integración de middleware, diseñada para satisfacer necesidades reales.
- **CONFIABLE.** Capaz de eliminar las brechas de comunicación y las ralentizaciones entre diferentes plataformas, sistemas críticos y componentes de aplicaciones, tanto dentro de la empresa como con sistemas y organizaciones externas.
- **UNIFICADO.** Proporciona un conjunto básico de patrones de mensajería a través de un único protocolo manejable.
- **COMPLETO.** JMS proporciona una API para protocolos de aplicación y se basa en Java. AMQP proporciona transporte a nivel de cable para las aplicaciones que utilizan dicha API. AMQP es ampliamente aplicable y se puede aprovechar en cualquier lenguaje, e identifica las semánticas de almacenamiento y reenvío y publicación y suscripción en una sola especificación.
- **ABIERTO.** Independiente del proveedor y la plataforma, y creado por usuarios y proveedores de tecnología que trabajan en colaboración.
- **SEGURO.** Una solución segura al problema de transportar mensajes valiosos entre organizaciones, plataformas tecnológicas y un entorno virtual de computación en la nube.

Utilice AMQP para implementar una arquitectura orientada a servicios



AMQP completa la imagen de una pila SOA abierta y pragmática al utilizar una infraestructura TCP/IP para el transporte que proporciona conectividad ubicua. El protocolo actúa como intermediario confiable para la entrega asincrónica de mensajes con características de servicio específicas. XML proporciona una codificación de datos uniforme, mientras que el esquema XML proporciona la especificación de metadatos para una representación adecuada. SOAP proporciona los patrones esperados de intercambio de mensajes, mientras que los estándares de servicios web permiten la representación de contratos comerciales.

Protocolo avanzado de cola de mensajes: AMQP

DICIEMBRE 13, 2024 13 MINS READ



Wallarm Platform

Solutions

Resources

Learn Wallarm

Protocolo avanzado de cola de mensajes: AMQP



AMQP significado

El Método Innovador de Distribución de Mensajes (AMQP, por sus siglas en inglés) proporciona un marco para la transmisión segura y fluida de datos entre un universo de sistemas mediante la utilización de una estructura de red. Este enfoque crea un flujo controlado de datos en una red, respaldando la interacción asíncrona entre los sistemas, eliminando la demanda de operatividad simultánea de cada sistema.

Descripción del funcionamiento de AMQP

AMQP adopta un enfoque comprensivo para la diseminación de datos. Los datos son transportados mediante un 'nexus', un componente que dirige los datos a múltiples canales. De esta forma, las entidades que reciben acceden a los datos desde estos canales para su análisis posterior. Este diseño facilita la interacción efectiva entre los sistemas, permitiendo reservar los datos en los canales hasta que estén listos para ser procesados.

Importancia del AMQP

AMQP es vital por diversas razones. En primer lugar, es de acceso libre, lo que indica que cualquier usuario puede implementarlo y optimizarlo, conduciendo a su adopción extensiva en varios ámbitos y circunstancias.

Además, AMQP respalda la interacción fluida entre sistemas. Los datos se pueden difundir y recuperar de forma asíncrona, es decir, los sistemas no tienen que estar operativos de manera concurrente. Esta característica es especialmente útil cuando la disponibilidad del sistema puede ser volátil o inconstante.

Por último, AMQP puede ser personalizado y adaptado a diversas aplicaciones. Su configuración es adaptable a diferentes esquemas de difusión de datos, desde modelos de emisión/adquisición, donde los datos se pueden enviar a varias entidades receptoras, hasta canales de trabajo, donde los datos se dirigen a un solo receptor.

Comparación del AMQP con otros protocolos de distribución de datos

Existen otros protocolos de distribución de datos como lo son MQTT y STOMP. Sin embargo, AMQP tiene ventajas considerables. AMQP ofrece más flexibilidad debido a los diversos esquemas de disseminación de datos que admite. Además, asegura una seguridad estable en la recepción de datos, que pueden ser reservados en los canales hasta que estén listos para su análisis. Por último, siendo de acceso libre, AMQP tiene alta adopción y es respaldado por un extenso grupo de desarrolladores.

Protocolo	Flexibilidad	Seguridad en la Recepción	Acceso Libre
AMQP	Alta	Excelente	Sí
MQTT	Moderada	Aceptable	Sí
STOMP	Limitada	Insuficiente	Sí

En conclusión, AMQP es un protocolo de distribución de datos que proporciona un medio seguro, eficiente y adaptable para la interacción entre sistemas. Su adopción en una amplia gama de entornos y la disponibilidad del código abierto, lo posicionan como una opción insuperable para un sinnúmero de aplicaciones y contextos.

Historia de AMQP

La historia de AMQP es una historia de colaboración y desarrollo continuo. En 2004, JPMorgan Chase, una de las instituciones financieras más grandes del mundo, se dio cuenta de que necesitaba una solución de mensajería más eficiente y confiable. En lugar de optar por una solución propietaria, decidió colaborar con iMatix Corporation, una empresa de software, para desarrollar un protocolo de mensajería abierto. Así nació AMQP.

El inicio de AMQP

En sus primeras etapas, AMQP fue diseñado para ser un protocolo de mensajería que pudiera manejar altos volúmenes de mensajes de manera eficiente y confiable. El objetivo era crear un protocolo que fuera fácil de implementar y que pudiera ser utilizado por cualquier empresa, independientemente de su tamaño o industria.

En 2006, se lanzó la primera versión de AMQP. Esta versión inicial ya mostraba algunas de las características clave de AMQP, como la entrega garantizada de mensajes y la capacidad de manejar diferentes tipos de mensajes.

Desarrollo y evolución de AMQP

Desde su lanzamiento inicial, AMQP ha evolucionado significativamente. En 2008, se lanzó la versión 0.9.1 de AMQP, que introdujo varias mejoras y nuevas características. Esta versión también marcó el inicio de la adopción más amplia de AMQP, con empresas como Microsoft y Red Hat comenzando a utilizar el protocolo.

En 2011, se lanzó la versión 1.0 de AMQP. Esta versión representó un cambio significativo en la arquitectura de AMQP, con un enfoque en la simplicidad y la flexibilidad. La versión 1.0 también introdujo el concepto de "intercambios", que permiten a los mensajes ser enviados a múltiples destinatarios.

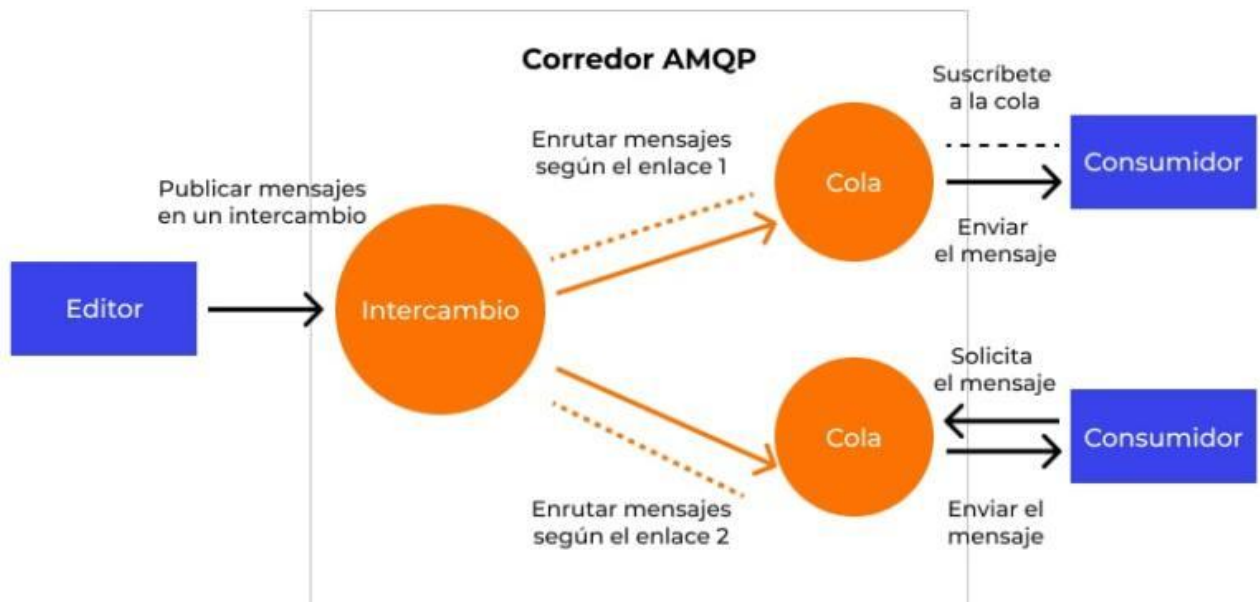
AMQP hoy

Hoy en día, AMQP es utilizado por una amplia gama de empresas e instituciones, desde gigantes tecnológicos como Microsoft y Google hasta instituciones financieras y gobiernos. El protocolo ha demostrado ser extremadamente eficaz en la entrega de mensajes de manera eficiente y confiable, y su diseño abierto ha permitido a las empresas adaptarlo a sus necesidades específicas.

En resumen, la historia de AMQP es una historia de innovación y colaboración. Desde su creación por JPMorgan Chase e iMatix Corporation, AMQP ha evolucionado para convertirse en uno de los protocolos de mensajería más utilizados en el mundo. Y con su enfoque en la eficiencia, la confiabilidad y la flexibilidad, es probable que AMQP continúe siendo una opción popular para la mensajería en el futuro.

Terminología AMQP

Antes de hundirnos en las profundidades técnicas de AMQP, es innegablemente importante entender una serie de términos clave que se emplean en este protocolo. Adelante, haremos un recorrido por algunos de las terminologías más usuales y su relevancia en el marco de AMQP.



Comunicado

En el ámbito de AMQP, un comunicado es la porción de información transportada de un punto a otro. Los comunicados se componen de dos segmentos esenciales: el preámbulo y el contenido. El preámbulo incorpora metadatos del comunicado, tales como su clase y precedencia, mientras que el contenido engloba los datos genuinos que están siendo transportados.

Generador

Un generador en este escenario es una aplicación que origina y despacha comunicados a una fila. Dicho de otro modo, es el origen de los comunicados en una infraestructura de AMQP.

Receptor

Un receptor en este sentido es una aplicación que recoge y trata los comunicados de una fila. En lenguaje sencillo, es el destino de los comunicados en una arquitectura AMQP.

Fila

La fila es una estructura de datos en la que se almacenan los comunicados a la espera de ser tratados. En el marco de AMQP, las filas son el lugar donde los generadores depositan los comunicados y donde los receptores los recogen.

Cambio

Un cambio en este orden es un distribuidor de comunicados en AMQP. Los generadores remiten comunicados a los cambios, y estos resuelven a qué filas deben ser despachados los comunicados de acuerdo a normas de distribución.

Conexión

Una conexión representa el vínculo entre una fila y un cambio. Los comunicados son transmitidos desde el cambio a la fila a través de la conexión.

Vía

La vía es un enlace virtual que se crea encima de una conexión de red. Las vías se emplean para suministrar y recibir comunicados entre generadores y receptores.

Operador

Un operador es un servidor que aplica el protocolo AMQP y ofrece la funcionalidad para suministrar y recibir comunicados.

Camino de distribución

Un camino de distribución es una llave que se utiliza para decidir a qué filas deberían ser enviados los comunicados. Los cambios usan el camino de distribución para concluir la manera de dispersar los comunicados a las filas.

Verificación

La verificación es un mecanismo que faculta a los receptores para notificar al operador de que han adquirido y procesado un comunicado con éxito.

Estas terminologías forman la base de la operativa de AMQP. En los próximos capítulos, indagaremos en cómo estos componentes interactúan entre sí para ofrecer una solución de mensajería sólida y fiable.

Componentes de AMQP

El Protocolo de Alto Rendimiento para la Gestión Avanzada de Mensajes en Cola (AMQP) se estratifica en diversas secciones fundamentales operando juntas para brindar un sistema de correspondencia confortable y versátil. Estas constituciones contemplan el eje primordial de la manipulación de mensajes, los emisores y suscriptores integrados en la plataforma, las vías de trasiego, las cajas de mensajes, las estaciones de tránsito y las pautas de vinculación.

Eje de Coordinación de Mensajes

Este eje esencial del AMQP funge como la estructura elemental del sistema. Encargado de la acogida, alojamiento y distribución de los mensajes a diferentes emisores y suscriptores integrados en la plataforma, también se inviste de la dirigencia y gestión de las cajas de mensajes y las estaciones de tránsito.

Emisores y Suscriptores Integrados en la Plataforma

Emisores y suscriptores están integrados en la plataforma y de hecho, son la misma plataforma, cuya responsabilidad yace en la creación y utilización de los mensajes. Los emisores brindan los mensajes al eje, mientras los suscriptores (usuarios) los captan. Tanto emisores como suscriptores se conectan con el eje de coordinación de mensajes a través de un enlace en la red.

Vías de Trasiego

Una vía de trasiego es una reunión virtual activa en un enlace de red entre el usuario y el eje de coordinación de mensajes. Estas vías posibilitan la concurrencia en la entrega y recepción de mensajes, evitando conexiones múltiples en la red. Cada vía tiene una designación única para las cajas y las estaciones de tránsito, permitiendo a los usuarios operar con múltiples cajas y estaciones de tránsito de manera simultánea.

Cajas de Mensajes

Las cajas de mensajes se utilizan para la conservación de los mensajes hasta su empleo. Cada mensaje es posicionado en una caja específica y los usuarios pueden subscribirse a un número variado de cajas para captar mensajes. Las cajas pueden ser permanentes (donde los mensajes persisten incluso si el sistema de mensajes se reinicia) o temporales (donde los mensajes se desvanecen al reiniciar el sistema de mensajes).

Estaciones de Tránsito

Las estaciones de tránsito son los puntos de arribo para los mensajes en el sistema AMQP. Los emisores brindan los mensajes a las estaciones de tránsito y éstas a su vez los dirigen a uno o más cajas basándose en las pautas de vinculación. Existen múltiples versiones de estaciones de tránsito, como las directas, temáticas, de salida y de encabezados, cada una con su propio conjunto de pautas de vinculación.

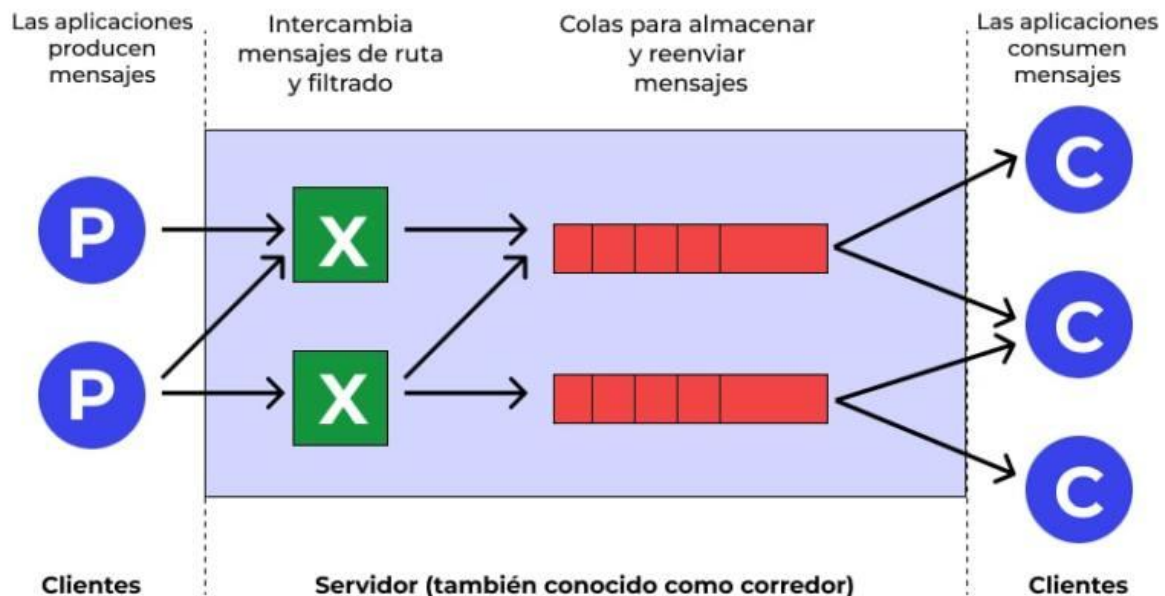
Pautas de Vinculación

Las pautas de vinculación son reglas que esbozan cómo los mensajes son dirigidos desde las estaciones de tránsito hasta las cajas. Una pauta de vinculación puede ser tan simple como una coincidencia precisa de una clave de vinculación o tan compleja como una expresión que contenga comodines y otros patrones.

Para concluir, cada segmento del AMQP colabora en conjunto para suministrar un mecanismo de mensajería eficiente y adaptable. Los emisores brindan mensajes a las estaciones de tránsito, las cuales los dirigen a las cajas adecuadas en base a las pautas de vinculación. Los usuarios después recoge los mensajes de las cajas a través de las vías de trasiego.

¿Cómo funcionan los intercambios AMQP?

Considerar AMQP y su funcionamiento de intercambios puede ser determinante para comprender la manipulación de la comunicación a través de este protocolo. Los intercambios actúan como mediadores entre los emisores de mensajes y las colas apropiadas, utilizando patrones y reglamentaciones específicas.



Variantes de Intercambios en el AMQP

Principalmente, se subdividen en cuatro categorías los intercambios del AMQP, cada uno aplicando su propia serie de normas para el direccionamiento de mensajes:

1. **Directo:** Los mensajes son redirigidos a las colas que corresponden exactamente con la clave del mensaje en este tipo de intercambio.
2. **Tema:** Esto permite la distribución de mensajes basada en patrones de correspondencia en los que las claves pueden ser especificadas con caracteres de reemplazo para permitir variadas correspondencias.
3. **Fanout:** Ignorando las claves de direccionamiento, este tipo de intercambio se encarga de distribuir el mensaje a todas las colas vinculadas a él.
4. **Cabeceras:** Este intercambio se encarga de direccionar los mensajes basándose en los atributos del encabezado del mensaje en vez de la clave de direccionamiento.

Dirección de Mensajes en los Intercambios AMQP

En los intercambios de AMQP se procede la dirección de mensajes de la siguiente manera:

1. **Generación del mensaje:** El remitente crea un mensaje específico y lo envía a un intercambio, el mensaje contiene una clave de dirección y posiblemente algunos encabezados.
2. **El intercambio recibe el mensaje:** El intercambio incorpora los mensajes y los categoriza en base a su clave de direccionamiento y/o encabezados.
3. **Distribución del Mensaje:** De acuerdo con las normas de direccionamiento y el tipo de intercambio, el mensaje es enviado a una o varias colas.
4. **Consumo del mensaje:** Los receptores ligados a las colas adquieren y procesan el mensaje.

Código de Ejemplo para trabajar con el intercambio de AMQP

A continuación, se muestra un ejemplo práctico del envío de un mensaje a través de un intercambio y la recepción de este en una cola por parte del receptor.

```
# Remitente
import pika

connection = pika.BlockingConnection(pika.ConnectionParameters('localhost'))
channel = connection.channel()

channel.exchange_declare(exchange='journal_direct', exchange_type='direct')

importance = 'error'
message = 'This is an error message.'

channel.basic_publish(exchange='journal_direct', routing_key=importance,
body=message)
print(" [x] Sent %r:%r" % (importance, message))

connection.close()

# Receptor
import pika

connection = pika.BlockingConnection(pika.ConnectionParameters('localhost'))
channel = connection.channel()

channel.exchange_declare(exchange='journal_direct', exchange_type='direct')

result = channel.queue_declare(queue='', exclusive=True)
queue_name = result.method.queue

importance = 'error'
channel.queue_bind(exchange='journal_direct', queue=queue_name,
routing_key=importance)

print(' [*] Waiting for logs. To exit press CTRL+C')

def callback(ch, method, properties, body):
    print(" [x] %r:%r" % (method.routing_key, body))

channel.basic_consume(queue=queue_name, on_message_callback=callback,
auto_ack=True)

channel.start_consuming()
```

En este código, se observa que el remitente envía un mensaje con una clave de direccionamiento 'error' a un intercambio de tipo 'directo'. El receptor sintoniza la misma cola y recibe todos los mensajes que tengan la misma clave de direccionamiento 'error'.

Los intercambios en AMQP juegan un papel esencial para la distribución adaptable de mensajes entre remitentes y receptores. Dependiendo del tipo de intercambio y sus normas de

direccionamiento, los mensajes se pueden entregar a una o más colas, lo que facilita una excelente flexibilidad y eficiencia en la gestión de la comunicación.

Desarrollo de API con AMQP

El mérito de estructurar APIs utilizando el Advanced Message Queuing Protocol (AMQP) consiste en la habilidad de configurar canales de comunicación entre diferentes sistemas de software. Mediante AMQP, se establece un roadmap consistente para intercambiar data entre distintos programas, optimizando la eficiencia y protección de dichos intercambios de información.

Beneficios de utilizar AMQP para construir APIs

Implementar AMQP en la creación de APIs desbloquea diversos beneficios. Primordialmente, AMQP es un protocolo estandarizado, facilitando la integración con múltiples plataformas y lenguajes de programación, lo que capacita para diseñar APIs que son compatibles con diversas aplicaciones, sin importar la plataforma o lenguaje de programación en uso.

Además, AMQP aporta funcionalidades que simplifican la estructuración de APIs confiables y seguras. Tal como, AMQP respalda la entrega confirmada de mensajes, asegurando que la data llegará a su destino, aun en circunstancias de fallos de red o del sistema.

Cómo crear una API empleando AMQP

Para construir una API con AMQP se requieren algunos pasos. En primer lugar, debes estructurar los mensajes que la API va a transferir. La complejidad y el tipo de data que decidas transmitir puede ser adaptable a tus necesidades.

Posteriormente, es requerido configurar las colas e intercambios que la API empleará. Las colas son los depósitos donde los mensajes esperan ser procesados, mientras que los intercambios actúan como los portales de acceso para los mensajes.

Finalmente, deberás discurrir el código que maneja el envío y recepción de mensajes mediante la API. Dicho código puede ser redactado en cualquier idioma de programación que soporte AMQP, y deberá aprovechar las herramientas proporcionadas por AMQP para el intercambio de mensajes.

Ejemplo de código

Aquí se muestra un exemplum de cómo uno puede desarrollar una API empleando AMQP con Python y la librería pika:

```
import pika

# Iniciar conexión con el servidor AMQP
conexion = pika.BlockingConnection(pika.ConnectionParameters('localhost'))

# Configuración del canal comunicativo
canal = conexion.channel()

# Declaración de la cola
canal.queue_declare(queue='hola')

# Transmisión de un mensaje
canal.basic_publish(exchange='',
                    routing_key='hola',
                    body='¡Hola Mundo!')
print(" [x] Enviado '¡Hola Mundo!'")

# Finalización de la conexión
conexion.close()
```


Este programa inicia una conexión con un servidor AMQP, instancia una cola, transfiere un mensaje a la cola, y luego termina la conexión.

Conclusión

La creación de APIs empleando AMQP es un método que puede asistirte en la creación de APIs de alta confiabilidad y seguridad. Con su amplio soporte para múltiples lenguajes de programación y plataformas, y su capacidad de entregar mensajes de forma segura, AMQP se consolida como una excelente elección para cualquier desarrollador de APIs.

La última palabra

El innovador sistema de comunicación que se apoya en el Proceso Superior de Manejo de Colas de Mensajes (AMQP) se posiciona como un recurso imprescindible para organizaciones contemporáneas. Su naturaleza de código abierto y estandarizada, facilita la fusión de variados sistemas y plataformas, dando pie a una colaboración más fluida.

Razones para decantarse por AMQP

La verdadera esencia de AMQP yace en su naturaleza de código abierto y estandarizada, facilitando la fusión sin interrupciones de variados sistemas y programas, dejando atrás cualquier preocupación por discrepancias. Su capacidad para adaptarse a una multitud de soluciones de transmisión de mensajes de diversos proveedores es su notable atributo.

Además, AMQP ofrece una gama de funciones extra que potencia la creación de sistemas de transmisión de mensajes duraderos y escalables. Gracias a su capacidad para ser compatible con varios modelos de intercambio de mensajes, como la transmisión directa, la transmisión temática y la difusión fanout, otorga a las organizaciones la posibilidad de personalizar sus soluciones de comunicación acorde a sus requerimientos particulares.

Uno de los puntos clave de AMQP es la garantía de entrega de mensajes. Esta certificación de envío y persistencia de mensajes facilita la recuperación en episodios inesperados de fallas de sistema o red, convirtiéndose en un elemento crucial para aplicaciones comerciales de relevancia.

Visiones Futuras para AMQP

En el actual entorno digital, la necesidad de implementar soluciones de transmisión de mensajes resilientes y expansibles es indiscutible. AMQP posee un papel principal en este desafío.

AMQP, respaldado por su diseño universal y normalizado, está preparado para ayudar en la consolidación de un marco efectivo para la integración y colaboración entre sistemas y plataformas dispares. Sus funciones avanzadas y la garantía en la entrega de mensajes le otorgan un lugar prominente como opción ideal para robustas y en crecimiento soluciones de transmisión de mensajes.

En resumen, AMQP es una extraordinaria y flexible herramienta de transmisión de mensajes que ayuda a las organizaciones a superar las barreras de integración y colaboración en el intrincado entorno empresarial actual. Su creciente adopción reafirma que seguirá siendo un elemento indispensable en el futuro de la transmisión de mensajes en los negocios.

Conclusión

El Proceso Superior de Manejo de Colas de Mensajes (AMQP) es un sistema de transmisión de mensajes resistente y versátil, fundamental para las corporaciones actuales. Gracias a su diseño universal y estandarizado, AMQP facilita la colaboración eficaz entre sistemas y plataformas variadas. Su singularidad y la garantía de la entrega de mensajes lo colocan como primera opción para la implementación de fuertes soluciones de transmisión de mensajes en expansión. En resumidas cuentas, si una empresa busca superar los obstáculos de integración y colaboración, AMQP sin duda debe ser considerado.

Enjoy Free API Security with Wallarm up to 500K Requests per month

Sign up for free

FAQ

Voy a abordar distintas cuestiones relativas al Protocolo Mejorado para la Administración de Filas de Mensajes, mejor conocido por sus siglas en inglés, AMQP.

¿Por qué es crucial AMQP y cómo se le emplea?

El AMQP tiene un rol fundamental como protocolo en la capa de aplicación, ya que facilita la transmisión y recepción de mensajes entre aplicaciones a través de una red. Su importancia es particularmente evidente en la mensajería corporativa, en la cual garantiza una transmisión efectiva de mensajes entre diferentes programas y sistemas.

¿De qué manera opera AMQP?

La operación de AMQP se basa en la gestión de intercambios y filas. Los mensajes se envían a un intercambio que, en base a las reglas de enrutamiento predefinidas, categoriza y dirige los mensajes hacia una o varias filas. Posteriormente, los destinatarios extraen los mensajes de las filas para su procesamiento.

¿Cuáles son los elementos esenciales de AMQP?

Los principales componentes de un sistema AMQP son el emisor, el intercambio, la fila y el receptor. El emisor remite los mensajes al intercambio, que los clasifica y reparte en las respectivas filas. Desde ahí, los receptores acceden y procesan los mensajes.

¿Cómo AMQP garantiza una entrega de mensajes segura?

AMQP utiliza varias estrategias para asegurar la entrega segura de los mensajes. Entre estas se incluyen las confirmaciones de envío, las confirmaciones de recepción y las transacciones.

¿Cómo se desarrolla una API con AMQP?

Cuando se desarrolla una API empleando AMQP, se utiliza este protocolo para facilitar la transmisión y recepción de mensajes entre aplicaciones a través de dicha interfaz de programación. Producto de esto es una interacción eficiente y segura entre las aplicaciones.

¿En qué se diferencia AMQP de otros protocolos de mensajería?

AMQP se distingue de otros protocolos de mensajería por diversas características. Como protocolo en la capa de aplicación, ofrece funcionalidades avanzadas inalcanzables en protocolos de capas inferiores. Adicionalmente, AMQP es de código libre, pudiendo ser implementado y usado sin limitaciones. Por último, es preferido en la mensajería de negocios por su habilidad para asegurar una comunicación segura de mensajes.

¿Cuáles son los pros y contras de usar AMQP?

El uso de AMQP trae consigo varias ventajas, como su capacidad para garantizar un intercambio de mensajes seguro, su condición de código libre y su uso extendido en la mensajería corporativa. No obstante, su implementación y uso pueden resultar más complejos en comparación con otros protocolos de mensajería. Además, a pesar de ser de código libre, no todos los sistemas de mensajería son compatibles con este.

¿Dónde puedo obtener más información acerca de AMQP?

Para indagar más acerca del AMQP, puedes visitar su página oficial en <http://www.amqp.org>. También existen diversos libros y tutoriales en línea que proporcionan una completa comprensión acerca del protocolo AMQP y su aplicación.

References

1. OASIS aprovecha el diseño de AMQP para simplificar el intercambio de datos.
2. RabbitMQ utiliza la variante de AMQP, intensificando la sincronización de la información.
3. El Estándar de Datos de Transmisión AMQP 1.0 fue forjado por el equipo de AMQP.
4. Microsoft Azure modera el flujo de los mensajes con el soporte del protocolo AMQP.
5. Con el uso de AMQP, IBM impulsa la conectividad robusta entre sus sistemas.
6. Red Hat asegura la integridad de mensajes transmitidos vía AMQP.
7. AMQP contribuye a sistemas de mensajes más productivos por obra de Apache Qpid.
8. Apache ActiveMQ entrelaza aplicaciones eficientemente bajo el estándar AMQP.
9. AMQP juega un papel crucial en la gestión de mensajes dentro de Pivotal Software.
10. Oracle mejora la sinergia entre sus servicios en la nube mediante AMQP.
11. Google Cloud garantiza la transmisión fiable de mensajes a través de AMQP.
12. Amazon Web Services pone en juego AMQP para enriquecer su compatibilidad.
13. VMware optimiza la armonía de sus aplicaciones y servicios a través de AMQP.
14. La mensajería rápida y segura en Salesforce se potencia con AMQP.
15. Para unificar microservicios, Cisco confía en AMQP.
16. Dell EMC apuesta en AMQP para interconexión efectiva de su infraestructura.
17. SAP se vale de AMQP para realzar la velocidad de transmisión de datos.
18. Software AG utiliza AMQP para garantizar la transmisión segura de mensajes.
19. En TIBCO Software, AMQP potencia la eficiencia en el envío de mensajes.
20. Informatica explota AMQP para intensificar la armonización de las aplicaciones.
21. AMQP sirve a MuleSoft para difundir su entramado de sistemas de integración.
22. WSO2 apuesta por AMQP para conexiones potentes y seguras entre microservicios.
23. Solace hace uso de AMQP para promover la coordinación en su plataforma de mensajería.
24. JBoss favorece la integración de aplicaciones mediante AMQP.
25. OpenText maneja grandes volúmenes de tareas integradoras gracias a AMQP.
26. Talend emplea AMQP para garantizar la seguridad en la transmisión de mensajes.
27. Progress Software utiliza AMQP para impulsar un canal de comunicación segura.
28. Axway conduce negocios con AMQP, confiando en su seguridad de transmisión de datos.
29. Fujitsu emplea AMQP para gestionar la comunicación de mensajes en tiempo real.
30. Hitachi sincroniza información de sistemas con ayuda de AMQP.
31. NEC utiliza AMQP para la gestión segura de interacciones entre interfaces.
32. Siemens emplea AMQP para optimizar la transferencia de mensajes.
33. Huawei opta por AMQP para su comunicación fluida entre una variedad de plataformas.
34. Ericsson procesa la información en tiempo real mediante AMQP.
35. ZTE hace uso de AMQP para mejorar la interoperabilidad de las aplicaciones.
36. Alcatel-Lucent se apoya en AMQP para optimizar la comunicación de sistemas.
37. En Nokia, AMQP permite el seguimiento del tráfico de mensajes.
38. Motorola Solutions utiliza AMQP para mejorar la funcionalidad comunicativa de sus sistemas.
39. Juniper Networks implementa AMQP para promover una integración sencilla de sistemas.
40. Brocade garantiza un canje de mensajes riguroso y puntual con AMQP.
41. Arista Networks actúa con AMQP para supervisar el tráfico de mensajes entre microservicios.
42. Extreme Networks recurre a AMQP para intensificar la transacción de mensajes.
43. F5 Networks aprovecha AMQP para mejorar la circulación de los mensajes.
44. Citrix ha acogido AMQP para potenciar su entrelazado de aplicaciones y servicios.
45. Fortinet aprovecha AMQP para conectar eficientemente sus bases de datos.
46. Para una encriptación de mensajes segura, Symantec invoca AMQP.
47. McAfee integra AMQP para realizar entregas de mensajes eficaces.
48. Trend Micro acelera la transmisión de mensajes mediante AMQP.
49. Kaspersky Lab recurre a AMQP para facilitar el intercambio de datos entre sistemas.
- Avast protege la transmisión de mensajes con la seguridad AMQP.

BIBLIOGRAFIA

- Arquitectura AMQP.
<https://www.amqp.org/product/architecture>
- Protocolo avanzado de cola de mensajes: AMQP
<https://lab.wallarm.com/what/protocolo-avanzado-de-cola-de-mensajes-amqp/?lang=es>