

---

# AWS Database Migration Service

## Guía del usuario

Versión de API API Version 2016-01-01



## AWS Database Migration Service: Guía del usuario

Copyright © 2019 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

## Table of Contents

¿Qué es AWS Database Migration Service? .....	1
Tareas de migración que realiza AWS DMS .....	1
Cómo funciona AWS DMS en el nivel básico .....	2
Funcionamiento de AWS DMS .....	4
Vista general de AWS DMS .....	4
Componentes .....	5
Orígenes .....	11
Destinos .....	12
Con otros servicios de AWS .....	13
Compatibilidad con AWS CloudFormation .....	13
Creación de un nombre ARN .....	14
Configuración .....	16
Inscripción en AWS .....	16
Creación de un usuario de IAM .....	16
Planificación de la migración para AWS Database Migration Service .....	18
Introducción .....	19
Iniciar una migración de bases de datos .....	19
Paso 1: Bienvenida .....	19
Paso 2: Crear una instancia de replicación .....	20
Paso 3: Especificar los puntos de enlace de origen y destino .....	25
Paso 4: Crear una tarea .....	30
Supervisar la tarea .....	34
Seguridad .....	35
Permisos de IAM necesarios .....	35
Roles de IAM para la CLI y la API .....	38
Control de acceso detallado .....	42
Uso de nombres de recursos para controlar el acceso .....	42
Uso de etiquetas para controlar el acceso .....	44
Configuración de una clave de cifrado .....	49
Seguridad de la red .....	50
Uso de SSL .....	51
Límites al uso de SSL con AWS Database Migration Service .....	52
Gestión de certificados .....	53
Activación de SSL para un punto de enlace PostgreSQL, SQL Server o compatible con MySQL .....	53
Compatibilidad con SSL para un punto de enlace de Oracle .....	55
Cambio de la contraseña de la base de datos .....	60
Límites .....	61
Límites de AWS Database Migration Service .....	61
Instancia de replicación .....	62
Instancias de replicación en profundidad .....	63
Instancias de replicación pública y privada .....	65
Mantenimiento de AWS DMS .....	65
Ventana de mantenimiento de AWS DMS .....	66
Versiones del motor de replicación .....	68
Descarte de una versión de la instancia de replicación .....	69
Actualización de la versión del motor de una instancia de replicación .....	69
Configuración de una red para una instancia de replicación .....	71
Configuraciones de red para migrar bases de datos .....	72
Creación de un grupo de subred de replicación .....	77
Configuración de una clave de cifrado .....	78
Creación una instancia de replicación .....	79
Modificación de una instancia de replicación .....	83
Reinicio de una instancia de replicación .....	86
Eliminación de una instancia de replicación .....	88

Instrucciones DDL compatibles .....	89
Puntos de enlace .....	90
Orígenes para la migración de datos .....	90
Usar Oracle como origen .....	91
Utilizar SQL Server como origen .....	108
Uso de Microsoft Azure SQL Database como origen .....	119
Utilizar PostgreSQL como origen .....	119
Utilizar MySQL como origen .....	132
Usar SAP ASE como origen .....	140
Utilizar MongoDB como origen .....	143
Uso de Amazon S3 como origen .....	148
Uso de IBM Db2 LUW como origen .....	154
Destinos para la migración de datos .....	157
Uso de Oracle como destino .....	159
Uso de SQL Server como destino .....	164
Uso de PostgreSQL como destino .....	167
Uso de MySQL como destino .....	170
Uso de Amazon Redshift como destino .....	175
Uso de SAP ASE como destino .....	187
Uso de Amazon S3 como destino .....	189
Uso de Amazon DynamoDB como destino .....	210
Uso de Amazon Kinesis Data Streams como destino .....	223
Uso de Amazon Elasticsearch Service como destino .....	229
Uso de Amazon DocumentDB como destino .....	232
Creación de puntos de enlace de origen y de destino .....	246
Tareas .....	250
Creación de un informe de evaluación de tarea .....	252
Creación de una tarea .....	254
Configuración de tarea .....	260
Establecimiento de compatibilidad con LOB .....	276
Crear varias tareas .....	277
Tareas de replicación continua .....	277
Replicación empezando desde un punto de inicio de CDC .....	278
Modificar una tarea .....	280
Volver a cargar tablas durante una tarea .....	280
Consola de administración de AWS .....	281
Mapeo de tablas .....	283
Especificación de selección de tabla y transformaciones mediante correspondencia de tabla desde la consola .....	283
Especificación de selección de tabla y transformaciones mediante correspondencia de tabla con JSON .....	288
Reglas y operaciones de table-settings .....	300
Usar filtros de origen .....	317
Monitorización de tareas .....	321
Estado de una tarea .....	321
Estado de la tabla durante las tareas .....	322
Uso de Amazon CloudWatch para la monitorización de tareas de replicación .....	323
Métricas del servicio de migración de datos .....	325
Métricas de instancia de replicación .....	326
Métricas de tareas de replicación .....	327
Administración de logs de AWS DMS .....	328
Registro de llamadas al API de AWS DMS con AWS CloudTrail .....	329
Información de AWS DMS en CloudTrail .....	330
Descripción de las entradas de archivos de registro de AWS DMS .....	330
Validación de tareas .....	334
Estadísticas de la tarea de replicación .....	335
Revalidación de tablas durante una tarea .....	337

Consola de administración de AWS .....	337
Solución de problemas .....	337
Limitaciones .....	338
Etiquetado de recursos .....	340
API .....	341
Uso de eventos y notificaciones .....	343
Categorías y mensajes de eventos de AWS DMS .....	344
Suscripción a notificaciones de eventos de AWS DMS .....	346
Consola de administración de AWS .....	347
API de AWS DMS .....	347
Migración de grandes almacenes de datos con Snowball Edge .....	348
Información general del proceso .....	349
Requisitos previos .....	350
Lista de comprobación de migración .....	350
Procedimientos paso a paso .....	352
Paso 1: Crear un trabajo de Snowball Edge .....	352
Paso 2: Descargar e instalar la Herramienta de conversión de esquemas de AWS (AWS SCT) .....	352
Paso 3: Desbloquear el dispositivo Snowball Edge .....	352
Paso 4: Configurar el host de agente de DMS con los controladores ODBC .....	354
Paso 5: Instalar el agente de DMS .....	356
Paso 6: Crear un nuevo proyecto de AWS SCT .....	358
Paso 7: Configure AWS SCT para utilizar el Snowball Edge .....	359
Paso 8: Registrar el agente de DMS en AWS SCT .....	361
Paso 9: Crear una tarea local y de DMS .....	362
Paso 10: Ejecutar y monitorizar la tarea en SCT .....	366
Limitaciones .....	368
Solución de problemas .....	369
Tareas de migración lenta .....	369
La barra de estado de la tarea no se mueve .....	370
Faltan claves externas e índices secundarios .....	370
Problemas de conexión de Amazon RDS .....	370
Mensaje de error de cadena de conexión de hilo incorrecta y valor de hilo incorrecto 0 .....	370
Problemas de red .....	371
Atasco de CDC después de migrar una carga total .....	371
Errores de violación de clave principal al reiniciar una tarea .....	371
Error al cargar por primera vez los esquemas .....	372
Fallan las tareas con un error desconocido .....	372
Al reiniciar la tarea las tablas se cargan desde el principio .....	372
Número de tablas por tarea .....	372
Solución de problemas específicos de Oracle .....	372
Obtención de datos de vistas .....	372
Migración de LOB desde Oracle 12c .....	373
Alternar entre Oracle LogMiner y Binary Reader .....	373
Error de CDC de Oracle detenido 122301 y de tope de reintentos de CDC de Oracle superado. ....	373
Adición automática de un registro complementario a un punto de enlace de origen de Oracle .....	374
No se están capturando los cambios de LOB .....	374
Error ORA-12899 de valor demasiado grande para <nombre-columna> .....	374
Malinterpretación del tipo de datos NUMBER .....	374
Solución de problemas específicos de MySQL .....	375
Error de la tarea CDC para el punto de enlace de la instancia de base de datos de Amazon RDS al haberse deshabilitado el registro binario .....	375
Las conexiones a una instancia de MySQL de destino se desconectan durante una tarea .....	375
Adición de Autocommit a un punto de enlace compatible con MySQL .....	376
Desactivación de claves externas en un punto de enlace de destino compatible con MySQL .....	376
Caracteres sustituidos por signos de interrogación .....	377
Entradas de registro "Bad event" (evento incorrecto) .....	377
Captura de datos de cambios con MySQL 5.5 .....	377

Aumento de la retención de registros binarios para instancias de base de datos de Amazon RDS .	377
Mensaje de registro que indica que algunos cambios desde la base de datos de origen no han surtido efecto al aplicarlos a la base de datos de destino. ....	377
Error de identificador demasiado largo .....	377
Error por el que un juego de caracteres incompatible generar error en la conversión de datos del campo .....	378
Error: página de códigos 1252 a UTF8 [120112] Se ha producido un error en la conversión de datos del campo .....	378
Solución de problemas específicos de PostgreSQL .....	379
Tipos de datos JSON truncados .....	379
Las columnas de un tipo de datos definido por el usuario no se migran correctamente .....	380
Error que indica que no se ha seleccionado ningún esquema de creación .....	380
No se están replicando las eliminaciones y las actualizaciones en una tabla mediante CDC .....	380
Las instrucciones TRUNCATE no se están propagando .....	380
Impedir que PostgreSQL capture instrucciones DDL .....	380
Selección del esquema donde crear los objetos de base de datos para capturar instrucciones DDL .....	380
Ausencia de tablas de Oracle después de migrar a PostgreSQL .....	381
La tarea de uso de vista como origen no tiene filas copiadas .....	381
Solución de problemas específicos de Microsoft SQL Server .....	381
Permisos especiales para que la cuenta de usuario AWS DMS utilice CDC .....	381
Errores al capturar cambios para una base de datos de SQL Server .....	381
Faltan columnas de identidad .....	382
Error debido a que SQL Server no admite publicaciones .....	382
Cambios que no aparecen en el destino .....	382
Solución de problemas específicos de Amazon Redshift .....	382
Carga en un clúster de Amazon Redshift en una región diferente de la de la instancia de replicación de AWS DMS .....	383
Error por existir ya la relación "awsdms_apply_exceptions" .....	383
Errores con tablas cuyo nombre comienza con "awsdms_changes" .....	383
Visualización de tablas en clúster con nombres como dms.awsdms_changes00000000XXXX .....	383
Permisos necesarios para trabajar con Amazon Redshift .....	383
Solución de problemas específicos de Amazon Aurora MySQL .....	384
Error por campos CHARACTER SET UTF8 terminados por ',' entre líneas "" terminadas por '\n' ...	384
Prácticas recomendadas .....	385
Mejora del desempeño .....	385
Determinación del tamaño de una instancia de replicación .....	387
Reducción de la carga de trabajo en su base de datos de origen .....	388
Uso del registro de tareas .....	388
Conversión de esquemas .....	388
Migración de objetos binarios grandes (LOB) .....	389
Uso del modo LOB limitado .....	389
Replicación continua .....	390
Cambio del usuario y esquema para un destino Oracle .....	390
Cambio de espacios de tabla de tabla e índice para un destino Oracle .....	391
Mejora del desempeño al migrar tablas de gran tamaño .....	392
Referencia .....	393
Tipos de datos de AWS DMS .....	393
Notas de la versión .....	395
Notas de la versión 3.1.3 de AWS DMS .....	395
Notas de la versión 3.1.2 de AWS DMS .....	396
Notas de la versión 3.1.1 de AWS DMS .....	397
Notas de la versión 2.4.5 de AWS DMS .....	399
Notas de la versión 2.4.4 de AWS DMS .....	400
Notas de la versión 2.4.3 de AWS DMS .....	401
Notas de la versión 2.4.2 de AWS DMS .....	402
Notas de la versión 2.4.1 de AWS DMS .....	404

Notas de la versión 2.4.0 de AWS DMS .....	405
Notas de la versión 2.3.0 de AWS DMS .....	406
Historial de revisión .....	410
Actualizaciones anteriores .....	410
AWS Glossary .....	413

# ¿Qué es AWS Database Migration Service?

AWS Database Migration Service (AWS DMS) es un servicio en la nube que facilita la migración de bases de datos relacionales, almacenes de datos, bases de datos NoSQL y otros tipos de almacenes de datos. Puede utilizar AWS DMS para migrar datos a la nube de AWS, entre instancias locales (a través de una configuración de nube de AWS) o entre combinaciones de configuraciones locales y en la nube.

Con AWS DMS, puede realizar migraciones puntuales, y puede replicar cambios en curso para mantener sincronizados los orígenes y los destinos. Si desea cambiar motores de base de datos, puede utilizar la Herramienta de conversión de esquemas de AWS (AWS SCT) para traducir su esquema de base de datos a la nueva plataforma. A continuación, utilice AWS DMS para migrar los datos. Dado que AWS DMS forma parte de la nube de AWS, obtendrá la rentabilidad, la velocidad de comercialización, la seguridad y la flexibilidad que ofrecen los servicios de AWS.

Para obtener información acerca de las regiones de AWS que admiten AWS DMS, consulte [Trabajar con una instancia de replicación de AWS DMS \(p. 62\)](#). Para obtener más información sobre el costo de migración de la base de datos utilizando , consulte la [página de precios de AWS Database Migration Service](#).

## Tareas de migración que realiza AWS DMS

AWS DMS se encarga de muchas de las tareas difíciles o tediosas relacionadas con un proyecto de migración:

- En una solución tradicional, debe realizar análisis de capacidad, procure hardware y software, instalar y administrar sistemas, y probar y depurar la instalación. administra AWS DMS automáticamente la implementación, la administración y el monitoreo de todo el hardware y el software necesarios para la migración. La migración puede empezar a funcionar en cuestión de minutos desde el inicio del proceso de configuración de AWS DMS.
- Con AWS DMS, puede ampliar (o reducir) los recursos de migración según sea necesario para adaptarlos a la carga de trabajo real. Por ejemplo, si determina que necesita almacenamiento adicional, puede aumentar fácilmente el almacenamiento asignado y reiniciar la migración, normalmente en cuestión de minutos. Por otra parte, si descubre que no se utiliza toda la capacidad de los recursos que ha configurado, puede reducirlos con facilidad para satisfacer su carga de trabajo real.
- AWS DMS utiliza un modelo de servicio de pago por uso. A diferencia de los modelos de licencia tradicionales, que tienen costos de compra iniciales y cargos de mantenimiento continuos, solo se paga por los recursos de AWS DMS mientras se utilizan.
- AWS DMS administra automáticamente toda la infraestructura necesaria para el servidor de migración, incluido el hardware y el software, la aplicación de parches de software y la notificación de errores.
- AWS DMS proporciona conmutación por error automática. Si el servidor de replicación principal tiene cualquier tipo de error, un servidor de replicación de reserva puede sustituirlo con poca o ninguna interrupción del servicio.
- AWS DMS puede ayudarle a cambiar a un motor de base de datos moderno, y posiblemente más rentable, que el que utiliza actualmente. Por ejemplo, AWS DMS puede ayudarle a aprovechar los servicios de base de datos administrados que ofrecen Amazon RDS o Amazon Aurora. O puede ayudarle a pasarse al servicio de almacén de datos administrado proporcionado por Amazon Redshift, a plataformas NoSQL como Amazon DynamoDB o a plataformas de almacenamiento de bajo costo, como Amazon Simple Storage Service (Amazon S3). Si, por el contrario, desea abandonar su infraestructura

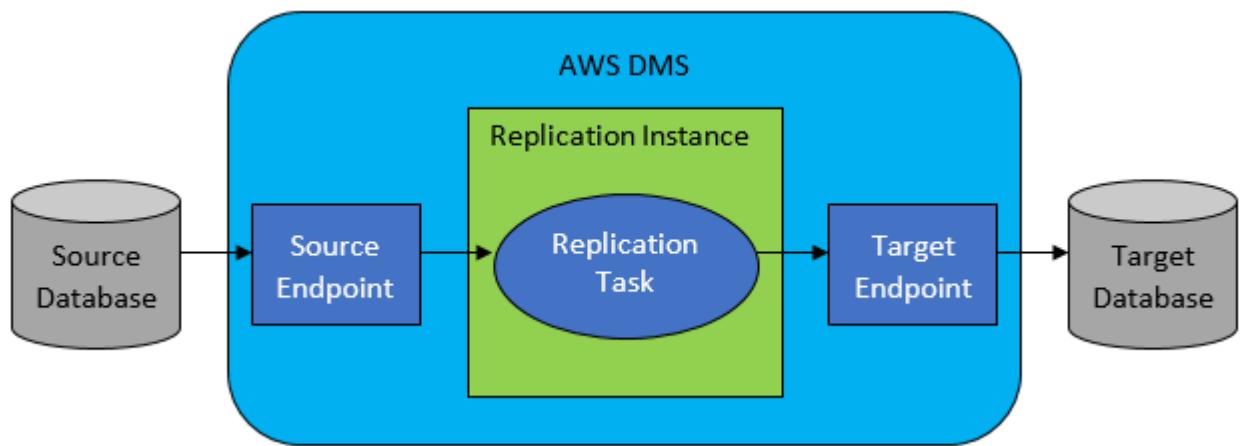
antigua, pero continuar utilizando el mismo motor de base de datos, AWS DMS también puede realizar este proceso.

- AWS DMS permite utilizar como origen de datos la mayoría de los motores de DBMS más populares actualmente, como Oracle, Microsoft SQL Server, MySQL, MariaDB, PostgreSQL, Db2 LUW, SAP, MongoDB y Amazon Aurora.
- AWS DMS dispone de una amplia gama de motores de destino, como Oracle, Microsoft SQL Server, PostgreSQL, MySQL, Amazon Redshift, SAP ASE, Amazon S3 y Amazon DynamoDB.
- Puede migrar desde cualquiera de los orígenes de datos admitidos a cualquiera de los objetivos de datos admitidos. AWS DMS admite totalmente migraciones de datos heterogéneos entre los motores compatibles.
- AWS DMS garantiza que la migración de datos es segura. Los datos en reposo se cifran con cifrado AWS Key Management Service (AWS KMS). Durante la migración, puede utilizar la capa de conexión segura (SSL) para cifrar los datos en tránsito desde el origen hasta el destino.

## Cómo funciona AWS DMS en el nivel básico

En su nivel más básico, AWS DMS es un servidor en la nube de AWS que ejecuta software de replicación. Debe crear una conexión de origen y de destino para indicar a AWS DMS de dónde debe extraer los datos y dónde debe cargarlos. A continuación, programe una tarea que se ejecuta en este servidor para mover los datos. AWS DMS crea las tablas y claves primarias asociadas si no existen en el destino. Puede crear previamente las tablas de destino de forma manual, si lo prefiere. También puede utilizar AWS SCT para crear algunos o todos los índices, tablas, vistas, disparadores, etc. de destino.

En el siguiente diagrama se ilustra el proceso de AWS DMS.



Para ejecutar el proceso de AWS DMS, de principio a fin

1. Para iniciar un proyecto de migración, identifique los almacenes de datos de origen y de destino. Estos almacenes de datos pueden residir en cualquiera de los motores de base de datos mencionados anteriormente.
2. Tanto para el origen como para el destino, configure los puntos de enlace en AWS DMS que especifiquen la información de conexión a las bases de datos. Los puntos de enlace utilizan los controladores ODBC apropiados para comunicarse con el origen y el destino.
3. Aprovisione una instancia de replicación, que es un servidor que AWS DMS configura automáticamente con software de replicación.
4. Crear una tarea de replicación, que especifica los datos reales y la transformación de datos tablas que se van a migrar reglas que se van a aplicar. AWS DMS administra la ejecución de la tarea de replicación y le proporciona el estado en el proceso de migración.

Para obtener más información, consulte lo siguiente:

- Si es la primera vez que utiliza AWS DMS, pero está familiarizado con otros servicios de AWS, comience por [Funcionamiento de AWS Database Migration Service \(p. 4\)](#). En esta sección, se explican los componentes principales de AWS DMS y el proceso global para configurar y ejecutar una migración.
- Si desea cambiar de motor de base de datos, la herramienta [AWS Schema Conversion Tool](#) puede convertir a la plataforma de destino el esquema de base de datos existente, incluidas las tablas, los índices y la mayoría del código de la aplicación.
- Para obtener más información sobre los servicios de AWS relacionados que podrían ser necesarios para diseñar la estrategia de migración, consulte [Productos de la nube de AWS](#).
- Amazon Web Services ofrece diversos servicios de base de datos. Para descubrir cuál es el mejor servicio para su entorno, consulte [Ejecución de bases de datos en AWS](#).
- Para ver información general sobre todos los productos de AWS, consulte [¿Qué es la informática en la nube?](#)

# Funcionamiento de AWS Database Migration Service

AWS Database Migration Service (AWS DMS) es un servicio web que puede utilizar para migrar datos de un almacén de datos de origen a otro de destino. Estos dos almacenes de datos se denominan puntos de enlace. Puede migrar entre puntos de enlace de origen y de destino que utilicen el mismo motor de base de datos, como desde una base de datos de Oracle a una base de datos de Oracle. También puede migrar entre puntos de enlace de origen y de destino que utilicen motores de base de datos distintos, como desde una base de datos de Oracle a una base de datos de PostgreSQL. El único requisito para utilizar AWS DMS es que uno de los puntos de enlace debe estar en un servicio de AWS. No puede usar AWS DMS para migrar desde una base de datos local a otra base de datos local.

Para obtener más información sobre el costo de migración de la base de datos utilizando , consulte la [página de precios de AWS Database Migration Service](#).

Utilice los siguientes temas para comprender mejor AWS DMS.

## Temas

- [Vista general de AWS DMS \(p. 4\)](#)
- [Componentes de AWS Database Migration Service \(p. 5\)](#)
- [Orígenes para AWS Database Migration Service \(p. 11\)](#)
- [Destinos para AWS Database Migration Service \(p. 12\)](#)
- [Uso de AWS DMS con otros servicios de AWS \(p. 13\)](#)

## Vista general de AWS DMS

Para realizar una migración de base de datos, AWS DMS se conecta con el almacén de datos de origen, lee los datos de origen y formatea los datos para que el almacén de datos de destino pueda consumirlos. A continuación, carga los datos en el almacén de datos de destino. La mayor parte de este proceso tiene lugar en la memoria, aunque es posible que las transacciones grandes necesiten almacenamiento en búfer en el disco. Las transacciones almacenadas en caché y los archivos de registro también se escriben en el disco.

A grandes rasgos, cuando utiliza AWS DMS, hace lo siguiente:

- Crear un servidor de replicación.
- Crear los puntos de enlace de origen y de destino que tienen información de conexión sobre sus almacenes de datos.
- Crear una o varias tareas de migración para migrar datos entre los almacenes de datos de origen y de destino.

Una tarea puede estar compuesta por tres fases principales:

- La carga completa de los datos existentes
- La aplicación de cambios en la memoria caché
- La replicación continua

Durante una migración de carga completa, en la que los datos existentes desde el origen se trasladan al destino, AWS DMS carga los datos de tablas en el almacén de datos de origen a tablas del almacén de datos de destino. Mientras se efectúa la carga completa, los cambios realizados en las tablas que se están cargando se almacenan en caché en el servidor de replicación. Estos son los cambios almacenados en caché. Es importante tener en cuenta que AWS DMS no captura cambios de una tabla determinada hasta que se haya iniciado la carga completa de esa tabla. En otras palabras, el punto en el que comienza la captura de cambios es diferente para cada tabla.

Cuando se ha terminado de cargar completamente una tabla determinada, AWS DMS comienza de inmediato a aplicar los cambios en la memoria caché para esa tabla. Cuando todas las tablas se han cargado, AWS DMS comienza a recopilar cambios como transacciones para la fase de replicación en curso. Una vez que AWS DMS aplique todos los cambios en la caché, las tablas serán coherentes desde el punto de vista transaccional. En este punto, AWS DMS pasa a la fase de replicación continua, en la que se aplican los cambios como transacciones.

Al iniciarse la fase de replicación continua, habrá transacciones pendientes que causarán por lo general cierto desfase entre la base de datos de origen y la de destino. La migración alcanzará finalmente un estado estable después de procesar la acumulación de transacciones pendientes. En ese punto, puede cerrar las aplicaciones, dejar que se apliquen al destino las transacciones restantes y abrir de nuevo las aplicaciones, que entonces apuntarán a la base de datos de destino.

AWS DMS crea los objetos del esquema de destino necesarios para realizar la migración. No obstante, AWS DMS adopta un enfoque minimalista y crea solo los objetos necesarios para migrar de manera eficiente los datos. En otras palabras, AWS DMS crea tablas, claves primarias y, en algunos casos, índices únicos, pero no crea ningún otro objeto que no se necesite para migrar eficientemente los datos desde el origen. Por ejemplo, no crea índices secundarios, limitaciones de claves no principales o valores predeterminados de datos.

En la mayoría de los casos, al migrar datos, también migra gran parte del esquema de origen (o su totalidad). Si se realiza una migración homogénea (entre dos bases de datos del mismo tipo de motor), el esquema se migra con las herramientas nativas del equipo para exportar e importar el esquema en sí, sin ningún dato.

Si su migración es heterogénea (entre dos bases de datos que utilizan diferentes tipos de motor), puede utilizar Herramienta de conversión de esquemas de AWS (AWS SCT) para generar un esquema de destino completo para usted. Si utiliza la herramienta, las dependencias entre tablas como, por ejemplo, limitaciones de la clave externa deben deshabilitarse durante las fases de "carga completa" y "aplicación de cambio en caché" de la migración. Si el desempeño es un problema, eliminar o desactivar los índices secundarios durante el proceso de migración ayuda. Para obtener más información sobre AWS SCT, consulte [AWS Schema Conversion Tool](#) en la documentación de AWS SCT.

## Componentes de AWS Database Migration Service

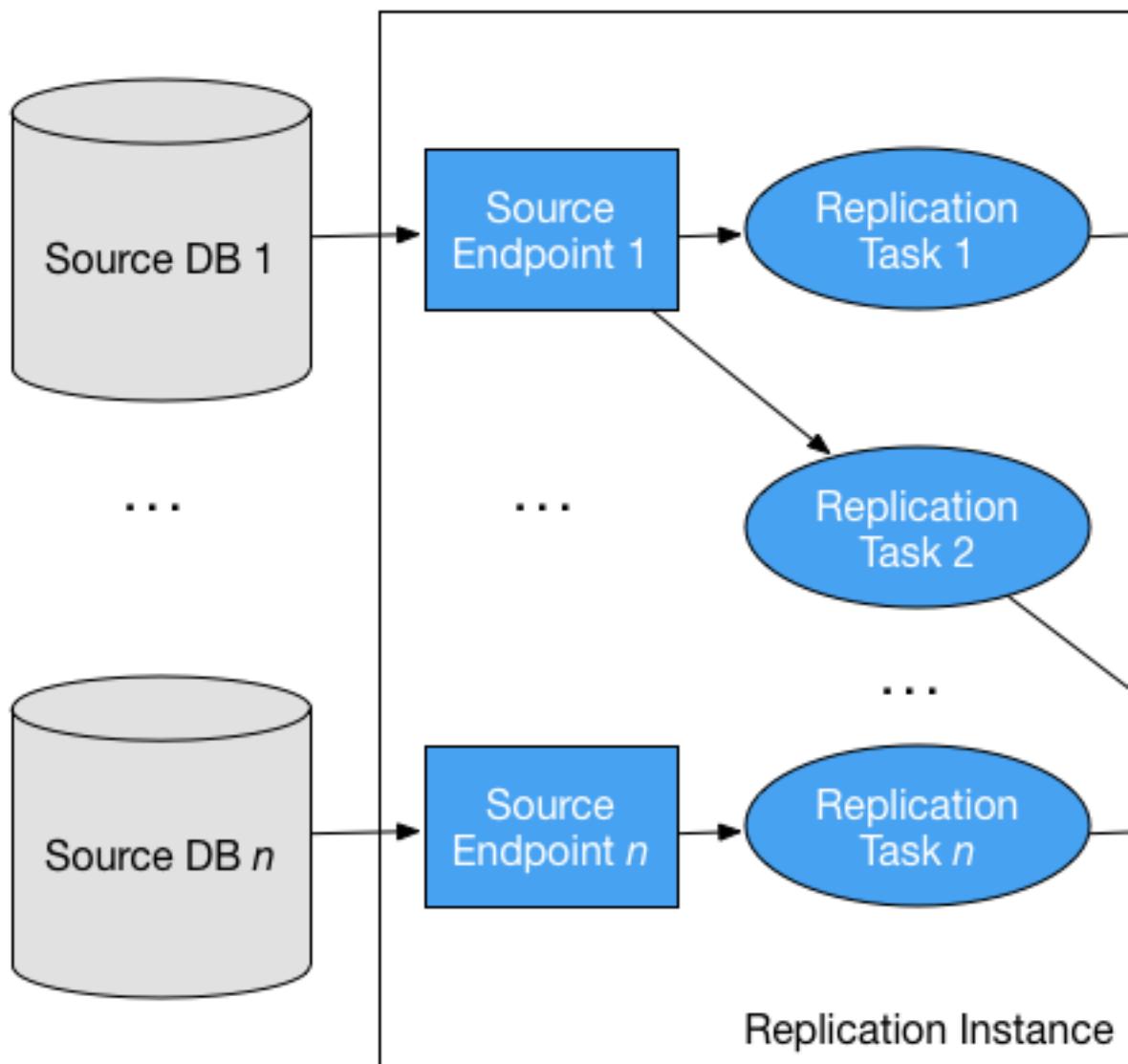
En esta sección se describen los componentes internos de AWS DMS y cómo funcionan juntos para llevar a cabo su migración de datos. Comprender los componentes subyacentes de AWS DMS puede ayudarle a migrar datos de forma más eficiente y a proporcionar una mejor visión a la hora de solucionar problemas o de investigarlos.

Una migración de AWS DMS consta de tres componentes: una instancia de replicación, puntos de enlace de origen y de destino y una tarea de replicación. Crea una migración de AWS DMS mediante la creación de la instancia de replicación necesaria, puntos de enlace y tareas en una región de AWS.

### Instancia de replicación

En líneas generales, una instancia de replicación de AWS DMS es sencillamente una instancia Amazon Elastic Compute Cloud (Amazon EC2) administrada que aloja una o varias tareas de replicación.

La figura siguiente muestra una instancia de replicación de ejemplo que ejecuta varias tareas de replicación asociadas.



Una sola instancia de replicación puede alojar una o varias tareas de replicación, en función de las características de su migración y la capacidad del servidor de replicación. AWS DMS proporciona una gran variedad de instancias de replicación de modo que puede elegir la configuración óptima para utilizar mayúsculas o minúsculas. Para obtener más información acerca de las diversas clases de instancias de replicación, consulte [Selección de la instancia de replicación de AWS DMS adecuada para su migración \(p. 63\)](#).

AWS DMS crea la instancia de replicación en una instancia de Amazon Elastic Compute Cloud (Amazon EC2). Algunas de las clases de instancias más pequeñas son suficientes para probar el

servicio o para pequeñas migraciones. Si su migración conlleva muchas tablas, o si va a ejecutar varias tareas de replicación simultáneas, debería plantearse el uso de una de las instancias más grandes. Le recomendamos este método porque AWS DMS puede consumir una cantidad significativa de memoria y de CPU.

De acuerdo con la clase de instancia Amazon EC2 que seleccione, su instancia de replicación viene con 50 GB o 100 GB de almacenamiento de datos. Esta cantidad suele ser suficiente para la mayoría de los clientes. Sin embargo, si la migración implica grandes transacciones o un alto volumen de cambios de datos, entonces es posible que desee aumentar la asignación de almacenamiento base. La captura de datos de cambio (CDC) puede provocar que los datos se escriban en el disco, en función de la rapidez con que el destino pueda escribir los cambios.

AWS DMS puede proporcionar alta disponibilidad y soporte de comutación por error mediante una implementación Multi-AZ. En una implementación Multi-AZ, AWS DMS aprovisiona y mantiene automáticamente una réplica en espera de la instancia de replicación en una zona de disponibilidad diferente. La instancia de replicación principal se replica sincrónicamente en la réplica en espera. Si la instancia de replicación principal falla o no responde, la instancia en espera reanuda cualquier tarea en ejecución con una interrupción mínima. Debido a que el nodo principal replica constantemente su estado a la espera, la implementación Multi-AZ incurre en algunos costos de desempeño.

Para obtener más información detallada acerca de la instancia de replicación de AWS DMS, consulte [Trabajar con una instancia de replicación de AWS DMS \(p. 62\)](#).

#### Puntos de enlace

AWS DMS utiliza un punto de enlace para obtener acceso a su almacén de datos de origen o de destino. La información de conexión específica es diferente, en función de su almacén de datos, pero en general proporcione la siguiente información al crear un punto de enlace.

- Tipo de punto de enlace: origen o destino.
- Tipo de motor: tipo de motor de base de datos, como, por ejemplo, Oracle, PostgreSQL o Amazon S3.
- Nombre del servidor: nombre del servidor o la dirección IP, deben poder conectarse desde AWS DMS
- Puerto: número de puerto utilizado para conexiones de servidor de base de datos.
- Cifrado: modo SSL, si se utiliza para cifrar la conexión.
- Credenciales: nombre de usuario y contraseña de una cuenta con los derechos de acceso necesarios.

Al crear un punto de enlace con la consola de AWS DMS, la consola requiere que pruebe la conexión de punto de enlace. La prueba debe realizarse correctamente antes de usar el punto de enlace en una tarea de DMS. Al igual que la información de conexión, los criterios de prueba específicos son distintos para los distintos tipos de motor. En general, AWS DMS verifica que la base de datos existe en el nombre de servidor y puerto especificado y que las credenciales suministradas se pueden utilizar para conectarse a la base de datos con los privilegios necesarios para realizar una migración. Si la prueba de conexión tiene éxito, AWS DMS descarga y almacena información de esquema, incluidas las definiciones de tablas y las definiciones de clave primaria/única, que se pueden utilizar más adelante durante la configuración de tareas.

Un único punto de enlace se puede utilizar en más de una tarea de replicación. Por ejemplo, podría tener dos aplicaciones distintas lógicamente alojadas en la misma base de datos de origen que desea migrar por separado. Debería crear dos tareas de replicación, una para cada conjunto de tablas de aplicación, pero puede utilizar el mismo punto de enlace de AWS DMS en ambas tareas.

Puede personalizar el comportamiento de un punto de enlace mediante el uso de atributos de conexión adicionales. Estos atributos pueden controlar diversos comportamientos tales como el detalle de registro, tamaño de archivo y otros parámetros. Cada tipo de motor de almacén de datos tiene distintos atributos de conexión adicionales disponibles. Puede encontrar los atributos de conexión

adicionales específicos para cada almacén de datos en la sección de origen o destino del almacén de datos. Para obtener una lista de los almacenes de datos de origen y de destino admitidos, consulte [Orígenes para AWS Database Migration Service \(p. 11\)](#) y [Destinos para AWS Database Migration Service \(p. 12\)](#).

Para obtener información detallada acerca de los puntos de enlace de AWS DMS, consulte [Trabajo con puntos de enlace de AWS DMS \(p. 90\)](#).

#### Tareas de replicación

Se utiliza una tarea de replicación de AWS DMS para mover un conjunto de datos desde el punto de enlace de origen al punto de enlace de destino. La creación de una tarea de replicación es el último paso que debe realizar antes de iniciar una migración.

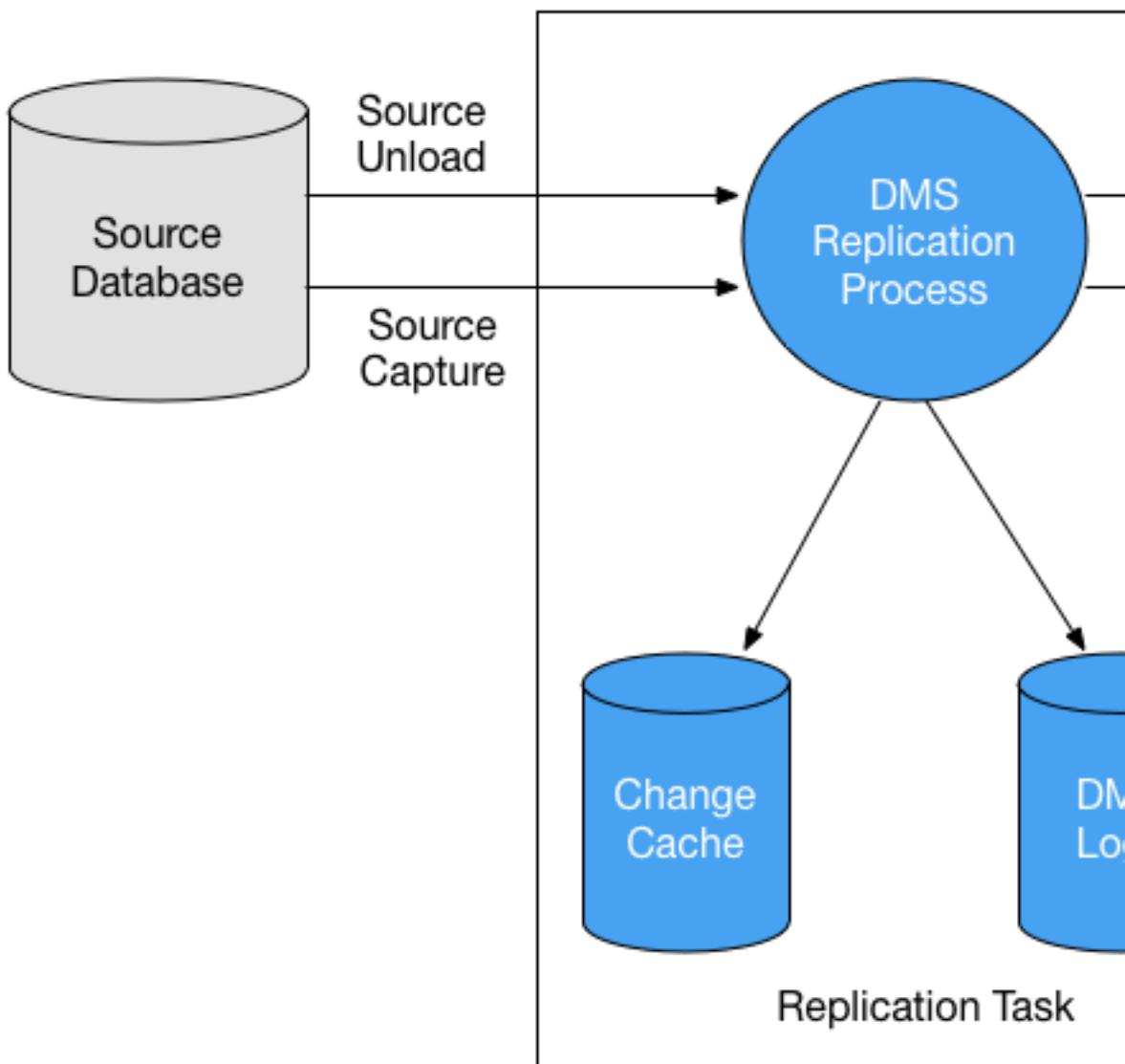
Cuando crea una tarea de replicación, puede especificar los siguientes ajustes de la tarea:

- Instancia de replicación: la instancia para alojar y ejecutar la tarea
- Punto de enlace de origen
- Punto de enlace de destino
- Opciones de tipo de migración, tal y como se indica a continuación. Para obtener una explicación completa de las opciones de tipos de migración, consulte [Creación de una tarea \(p. 254\)](#).
  - Carga completa (Migrar datos existentes): esta opción es buena si puede permitirse interrumpir la actividad durante el tiempo suficiente como para copiar sus datos existentes. Esta opción simplemente migra los datos de la base de datos de origen a la de destino y crea tablas cuando es necesario.
  - Carga completa + CDC (Migrar datos existentes y replicar los cambios en curso): esta opción realiza una carga completa de los datos mientras captura los cambios del origen. Una vez terminada la carga completa, los cambios capturados se aplican al destino. En el tiempo debido, la aplicación de los cambios alcanzará un estado de estabilidad. En ese momento, puede cerrar las aplicaciones, dejar que los cambios restantes fluyan al destino y reiniciar después las aplicaciones que apunten al destino.
  - CDC solo (Replicar solo los cambios en los datos): en algunas situaciones puede resultar más práctico copiar los datos existentes utilizando un método que no sea AWS DMS. En una migración homogénea, por ejemplo, usar herramientas nativas de exportación/importación puede ser más eficaz para cargar datos masivos. En este caso, puede utilizar AWS DMS para replicar cambios desde el momento en el que se inicia la carga masiva para que las bases de datos de origen y de destino se sincronicen y sigan sincronizadas.
- Opciones de modo de preparación de tabla de destino, tal y como se indica a continuación. Para obtener una explicación completa de los modos de tabla de destino, consulte [Creación de una tarea \(p. 254\)](#).
  - Do nothing (No hacer nada): AWS DMS presupone que las tablas de destino se han creado previamente en el destino.
  - Drop tables on target (Borrar tablas en el destino): AWS DMS borra las tablas de destino y las vuelve a crear.
  - Truncate (Truncar): si ha creado tablas en el destino, AWS DMS las trunca antes de que comience la migración. Si no existen tablas y selecciona esta opción, AWS DMS crea las tablas que faltan.
- Opciones de modo LOB, tal y como se indica a continuación. Para obtener una explicación completa de los modos de LOB, consulte [Establecimiento de compatibilidad con LOB para bases de datos de origen en la tarea de AWS DMS \(p. 276\)](#).
  - Don't include LOB columns (No incluir columnas LOB): las columnas LOB se excluyen de la migración.
  - Full LOB mode (Modo de LOB completo): se migran los LOB completos independientemente de su tamaño. AWS DMS migra los LOB por partes en porciones que se controlan con el parámetro Max LOB Size (Tamaño máximo de LOB). Este modo es más lento que utilizar el modo de LOB limitado.

- Limited LOB mode (Modo de LOB limitado): los LOB se truncan de acuerdo con el valor especificado por el parámetro Max LOB size (Tamaño máximo de LOB). Este modo es más rápido que utilizar el modo de LOB completo.
- Mapeos de tablas: indica las tablas que se van a migrar y la forma en que se migran. Para obtener más información, consulte [Uso de la correspondencia de tablas para especificar la configuración de tareas \(p. 283\)](#).
- Transformaciones de datos, tal y como se indica a continuación. Para obtener más información acerca de las transformaciones de datos, consulte [Especificación de selección de tabla y transformaciones mediante correspondencia de tabla con JSON \(p. 288\)](#).
  - Cambio de nombres de esquemas, tablas y columnas.
  - Cambio de nombres de espacios de tablas (para puntos de enlace de Oracle).
  - Definición de claves primarias e índices únicos en el destino.
- Validación de datos
- Registro de Amazon CloudWatch

Utiliza la tarea para migrar datos desde el punto de enlace de origen hasta el punto de enlace de destino y el procesamiento de tareas se realiza en la instancia de replicación. Especifica qué tablas y esquemas desea migrar y cualquier procesamiento especial, como, por ejemplo, los requisitos de registro, los datos de la tabla de control y la gestión de errores.

Conceptualmente, una tarea de replicación de AWS DMS realiza dos funciones diferentes, tal y como se muestra en el diagrama siguiente:



El proceso de carga completa es sencillo de comprender. Los datos se extraen desde el origen de forma masiva y se cargan directamente en el destino. Puede especificar el número de tablas que extraer y cargar en paralelo en la consola de AWS DMS en Advanced Settings (Configuración avanzada).

Para obtener más información sobre las tareas de AWS DMS, consulte [Trabajar con tareas de AWS DMS \(p. 250\)](#).

Replicación continua o captura de datos de cambio (CDC)

También puede utilizar una tarea AWS DMS para capturar los cambios en curso en el almacén de datos de origen durante la migración de sus datos a un destino. El proceso de captura de cambios que

AWS DMS utiliza para replicar los cambios en curso a partir de un punto de enlace de origen recopila los cambios en los registros de la base de datos mediante la API nativa del motor de la base de datos.

En el proceso de CDC, la tarea de replicación está diseñada para transmitir los cambios desde el origen al destino, utilizando búferes en memoria para almacenar datos en tránsito. Si los búferes en memoria se agotan por cualquier motivo, la tarea de replicación volcará los cambios pendientes en la caché de cambio en disco. Esto podría ocurrir, por ejemplo, si AWS DMS captura cambios desde el origen más rápido de lo que se pueden aplicar en el destino. En este caso, verá que la latencia de destino de la tarea supera la latencia de origen de la tarea.

Puede comprobar esto accediendo a su tarea en la consola de AWS DMS y abriendo la pestaña Task Monitoring (Monitorización de tareas). Los gráficos CDCLatencyTarget y CDCLatencySource se muestran en la parte inferior de la página. Si tiene una tarea que muestra una latencia de destino, entonces probablemente es necesario realizar algún ajuste en el punto de enlace de destino para aumentar la tasa de aplicación.

La tarea de replicación también utiliza el almacenamiento para registros de tareas tal y como se ha explicado anteriormente. El espacio de disco que viene preconfigurado con su instancia de replicación suele ser suficiente para el registro y los cambios de volcado. Si necesita espacio en disco adicional, por ejemplo, cuando se utiliza la depuración detallada para investigar un problema de migración, puede modificar la instancia de replicación para asignar más espacio.

#### Migración de esquemas y códigos

AWS DMS no convierte ni los esquemas ni el código. Puede utilizar herramientas como Oracle SQL Developer, MySQL Workbench o pgAdmin III para trasladar su esquema si su origen y su destino se corresponden con el mismo motor de base de datos. Si desea convertir un esquema existente en otro motor de base de datos, puede utilizar AWS SCT. Puede crear un esquema de destino y también generar y crear un esquema completo: tablas, índices, vistas y así sucesivamente. También puede utilizar la AWS SCT para convertir PL/SQL o TSQL en PgSQL y otros formatos. Para obtener más información sobre AWS SCT, consulte [AWS Schema Conversion Tool](#).

Siempre que es posible, AWS DMS crea automáticamente el esquema de destino por usted. A veces, AWS DMS no puede crear el esquema: por ejemplo, AWS DMS no crea un esquema de Oracle de destino por motivos de seguridad. En destinos de bases de datos MySQL, puede utilizar atributos de conexión adicionales para que AWS DMS migre todos los objetos al esquema y la base de datos especificados o los vaya creando a medida que encuentre el esquema en el origen.

## Orígenes para AWS Database Migration Service

Puede utilizar los siguientes almacenes de datos como puntos de enlace de origen para migrar datos con AWS Database Migration Service.

#### Bases de datos locales y de instancia EC2

- Versiones de Oracle 10.2 y posteriores, 11g y hasta 12.2, para las ediciones Enterprise, Standard, Standard One y Standard Two
- Versiones de Microsoft SQL Server 2005, 2008, 2008R2, 2012, 2014 y 2016, para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.
- Versiones de MySQL 5.5, 5.6 y 5.7.
- MariaDB (se admite como origen de datos compatible con MySQL).
- PostgreSQL, versión 9.4 y posteriores.
- Versiones 2.6.x, 3.x de MongoDB y posteriores.
- SAP Adaptive Server Enterprise (ASE) versiones 12.5, 15, 15.5, 15.7, 16 y posteriores.

- Versiones de Db2 LUW:
  - Versión 9.7, se admiten todos los Fix Packs.
  - Versión 10.1, se admiten todos los Fix Packs.
  - Version 10.5, se admiten todos los Fix Packs excepto Fix Pack 5.

#### Microsoft Azure

- Base de datos SQL Azure.

#### Bases de datos de instancias de Amazon RDS y Amazon Simple Storage Service (Amazon S3)

- Oracle versiones 11g (versiones 11.2.0.3.v1 y posteriores) y 12c para las ediciones Enterprise, Standard, Standard One y Standard Two.
- Versiones de Microsoft SQL Server 2008R2, 2012 y 2014 para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.
- Versiones de MySQL 5.5, 5.6 y 5.7.
- MariaDB (se admite como origen de datos compatible con MySQL).
- PostgreSQL 9.4 y posteriores. La captura de datos de cambios (CDC) es solo compatible con las versiones 9.4.9 y posteriores y la 9.5.4 y posteriores. El parámetro `rds.logical_replication`, necesario para CDC, se admite únicamente en estas versiones y las posteriores.
- Amazon Aurora (se admite como un origen de datos compatible con MySQL).
- Amazon S3.

## Destinos para AWS Database Migration Service

Puede utilizar los siguientes almacenes de datos como puntos de enlace de destino para migrar datos con AWS Database Migration Service.

#### Bases de datos locales y de instancias Amazon EC2

- Las versiones 10g, 11g y 12c de Oracle para las ediciones Enterprise, Standard, Standard One y Standard Two
- Versiones de Microsoft SQL Server 2005, 2008, 2008R2, 2012, 2014 y 2016, para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.
- Versiones de MySQL 5.5, 5.6 y 5.7
- MariaDB (se admite como destino de datos compatible con MySQL)
- Versiones 9.4 y posteriores de PostgreSQL
- SAP Adaptive Server Enterprise (ASE) versiones 15, 15.5, 15.7, 16 y posteriores

#### Bases de datos de instancias de Amazon RDS, Amazon Redshift, Amazon DynamoDB, Amazon S3, Amazon Elasticsearch Service, Amazon Kinesis Data Streams y Amazon DocumentDB

- Versiones de Oracle 11g (versiones 11.2.0.3.v1 y posteriores) y 12c para las ediciones Enterprise, Standard, Standard One y Standard Two.
- Versiones de Microsoft SQL Server 2008R2, 2012 y 2014, para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.
- Versiones de MySQL 5.5, 5.6 y 5.7
- MariaDB (se admite como destino de datos compatible con MySQL)

- Versiones 9.4 y posteriores de PostgreSQL
- Compatibilidad de Amazon Aurora con MySQL
- Amazon Aurora con compatibilidad con PostgreSQL
- Amazon Redshift
- Amazon S3
- Amazon DynamoDB
- Amazon Elasticsearch Service
- Amazon Kinesis Data Streams
- Amazon DocumentDB (con compatibilidad con MongoDB)

## Uso de AWS DMS con otros servicios de AWS

Puede utilizar AWS DMS con otros servicios de AWS:

- Utilice una instancia Amazon EC2 o instancia de base de datos Amazon RDS como destino de una migración de datos.
- Puede utilizar Herramienta de conversión de esquemas de AWS (AWS SCT) para convertir el código SQL y el esquema de origen en código SQL y un esquema de destino equivalentes.
- Puede utilizar Amazon S3 como lugar de almacenamiento de sus datos o como paso intermedio al migrar grandes cantidades de datos.
- Puede utilizar AWS CloudFormation para configurar sus recursos de AWS para la administración o la implementación de infraestructura. Por ejemplo, puede aprovisionar los recursos de AWS DMS, como las instancias de replicación, tareas, certificados y puntos de enlace. Puede crear una plantilla que describa todos los recursos de AWS que desea y AWS CloudFormation aprovisiona y configura estos recursos por usted.

## Compatibilidad de AWS DMS con AWS CloudFormation

Puede aprovisionar recursos de AWS Database Migration Service mediante AWS CloudFormation. AWS CloudFormation es un servicio que le ayuda a crear el modelo y configurar sus recursos de AWS para la gestión de la infraestructura o la implementación. Por ejemplo, puede aprovisionar los recursos de AWS DMS, como las instancias de replicación, tareas, certificados y puntos de enlace. Puede crear una plantilla que describa todos los recursos de AWS que desea y AWS CloudFormation aprovisiona y configura estos recursos por usted.

Como desarrollador o administrador del sistema, puede crear y administrar colecciones de estos recursos que puede utilizar para tareas de migración repetitivas o la implementación de recursos para su organización. Para obtener más información acerca de AWS CloudFormation, consulte [¿Qué es AWS CloudFormation?](#) en la Guía del usuario de AWS CloudFormation.

AWS DMS es compatible con la creación de los siguientes recursos de AWS DMS mediante AWS CloudFormation:

- [AWS::DMS::Certificate](#)
- [AWS::DMS::Endpoint](#)
- [AWS::DMS::EventSubscription](#)
- [AWS::DMS::ReplicationInstance](#)

- AWS::DMS::ReplicationSubnetGroup
- AWS::DMS::ReplicationTask

## Creación de un nombre de recurso de Amazon (ARN) para usarlo con AWS DMS

Si utiliza la AWS CLI o la API de AWS Database Migration Service para automatizar la migración de su base de datos, entonces necesita saber cómo trabajar con los nombres de recurso de Amazon (ARN). Los recursos que se crean en Amazon Web Services se identifican mediante un nombre ARN, que es un identificador único. Si utiliza la AWS CLI o la API de AWS DMS para configurar la migración de su base de datos, debe proporcionar el nombre ARN del recurso con el que desea trabajar.

Los nombres ARN para los recursos de AWS DMS emplean la sintaxis siguiente:

`arn:aws:dms:<region>:<account number>:<resourcetype>:<resourcename>`

En esta sintaxis:

- `<region>` es el ID de la región de AWS en la que se ha creado el recurso de AWS DMS, por ejemplo, `us-west-2`.

En la siguiente tabla se muestran los nombres de región de AWS y los valores que debe utilizar al crear el nombre ARN.

Región	Nombre
Región Asia Pacífico (Tokio)	ap-northeast-1
Región Asia Pacífico (Seúl)	ap-northeast-2
Región Asia Pacífico (Mumbai)	ap-south-1
Región Asia Pacífico (Singapur)	ap-southeast-1
Región Asia Pacífico (Sídney)	ap-southeast-2
Región Canadá (Central)	ca-central-1
Región UE (Fráncfort)	eu-central-1
Región UE (Irlanda)	eu-west-1
Región UE (Londres)	eu-west-2
Región América del Sur (São Paulo)	sa-east-1
Región EE.UU. Este (Norte de Virginia)	us-east-1
Región EE.UU Este (Ohio)	us-east-2
EE.UU. Oeste (Norte de California)	us-west-1
Región EE.UU. Oeste (Oregón)	us-west-2

- `<número de cuenta>` es su número de cuenta sin guiones. Para encontrar su número de cuenta, inicie sesión en su cuenta de AWS en <http://aws.amazon.com>, elija My Account/Console (Mi cuenta/Consola) y, a continuación, elija My Account (Mi cuenta).
- `<resourcetype>` es el tipo de recurso de AWS DMS.

La tabla siguiente muestra los tipos de recursos que debe utilizar al crear un nombre ARN para un recurso concreto de AWS DMS.

Tipo de recurso de AWS DMS	Formato de ARN
Instancia de replicación	<code>arn:aws:dms:&lt;region&gt;:&lt;account&gt;:rep:&lt;resourcename&gt;</code>
Punto de enlace	<code>arn:aws:dms:&lt;region&gt;:&lt;account&gt;:endpoint:&lt;resourcename&gt;</code>
Tarea de replicación	<code>arn:aws:dms:&lt;region&gt;:&lt;account&gt;:task:&lt;resourcename&gt;</code>
Grupo de subredes	<code>arn:aws:dms:&lt;region&gt;:&lt;account&gt;:subgrp:&lt;resourcename&gt;</code>

- `<resourcename>` es el nombre asignado a un recurso de AWS DMS. Es una cadena que se genera arbitrariamente.

La siguiente tabla muestra ejemplos de nombres ARN para recursos de AWS DMS con una cuenta de AWS de 123456789012, creados en la región US East (N. Virginia) y con nombre de recurso:

Tipo de recurso	Ejemplo de ARN
Instancia de replicación	<code>arn:aws:dms:us-east-1:123456789012:rep:QLXQZ64MH7CXF4QCQOMGRVYVXAI</code>
Punto de enlace	<code>arn:aws:dms:us-east-1:123456789012:endpoint:D3HMZ2IGUCGFF3NTAXUXGF6S5A</code>
Tarea de replicación	<code>arn:aws:dms:us-east-1:123456789012:task:2PVREMWNPGYJCVU2IBPTOYTIV4</code>
Grupo de subredes	<code>arn:aws:dms:us-east-1:123456789012:subgrp:test-tag-grp</code>

# Configuración para AWS Database Migration Service

Antes de usar AWS Database Migration Service (AWS DMS) por primera vez, realice las siguientes tareas:

1. [Inscripción en AWS \(p. 16\)](#)
2. [Creación de un usuario de IAM \(p. 16\)](#)
3. [Planificación de la migración para AWS Database Migration Service \(p. 18\)](#)

## Inscripción en AWS

Al inscribirse en Amazon Web Services (AWS), su cuenta de AWS se inscribe automáticamente en todos los servicios de AWS, incluido AWS DMS. Solo se le cobrará por los servicios que utilice.

Con AWS DMS, paga solo por los recursos que usa. La instancia de replicación de AWS DMS que crea se ejecuta en un entorno real (no en un entorno de prueba). Deberá pagar las tarifas de uso estándares de AWS DMS para la instancia hasta que la elimine. Para obtener más información sobre las tarifas de uso de AWS DMS, consulte la [página del producto AWS DMS](#). Si es un nuevo cliente de AWS, puede comenzar a utilizar AWS DMS de forma gratuita. Para obtener más información, consulte la sección sobre la [capa de uso gratuito de AWS](#).

Si cierra su cuenta de AWS, todos los recursos de AWS DMS y configuraciones asociadas a su cuenta se eliminan al cabo de dos días. Estos recursos incluyen todas las instancias de replicación, configuración de punto de enlace de origen y de destino, tareas de replicación y certificados SSL. Si después de dos días decide volver a utilizar AWS DMS, vuelva a crear los recursos necesarios.

Si ya dispone de una cuenta de AWS, pase a la siguiente tarea.

Si no dispone de una cuenta de AWS, utilice el siguiente procedimiento para crearla.

Para inscribirse en AWS

1. Abra <https://aws.amazon.com/> y elija Create an AWS Account.
2. Siga las instrucciones en línea.

Anote su número de cuenta de AWS porque lo necesitará en la siguiente tarea.

## Creación de un usuario de IAM

Cuando obtenga acceso a los servicios de AWS, como AWS DMS, debe proporcionar las credenciales para que estos puedan determinar si tiene permiso de acceso a sus recursos. La consola requiere que especifique la contraseña. Puede crear claves de acceso para su cuenta de AWS para tener acceso a la interfaz de línea de comandos o API. No obstante, no recomendamos que acceda a AWS con las credenciales de su cuenta de AWS, es mejor que utilice AWS Identity and Access Management (IAM) en su lugar. Cree un usuario de IAM y agréguelo a un grupo de IAM con permisos administrativos, o conceda al usuario permisos administrativos. De este modo podrá tener acceso a AWS mediante una dirección URL especial y las credenciales del usuario de IAM.

Si se ha inscrito en AWS, pero no ha creado un usuario de IAM para usted, puede crearlo en la consola de IAM.

Para crearse usted mismo un usuario administrador y agregarlo a un grupo de administradores (consola)

1. Utilice la dirección de correo electrónico y la contraseña de su cuenta de AWS para iniciar sesión como [Usuario de la cuenta raíz de AWS](https://console.aws.amazon.com/iam/) en la consola de IAM en <https://console.aws.amazon.com/iam/>.

Note

Le recomendamos que siga la práctica recomendada de utilizar el usuario de IAM **Administrator** como se indica a continuación y guardar de forma segura las credenciales de usuario raíz. Inicie sesión como usuario raíz únicamente para realizar algunas [tareas de administración de servicios y de cuentas](#).

2. En el panel de navegación, elija Users (Usuarios) y, a continuación, elija Add user (Añadir usuario).
3. En User name (Nombre de usuario), escriba **Administrator**.
4. Marque la casilla situada junto a Consola de administración de AWS access (Acceso a la Consola de administración de AWS). A continuación, seleccione Custom password (Contraseña personalizada) y luego escriba la nueva contraseña en el cuadro de texto.
5. (Opcional) De forma predeterminada, AWS requiere al nuevo usuario que cree una nueva contraseña la primera vez que inicia sesión. Puede quitar la marca de selección de la casilla de verificación situada junto a User must create a new password at next sign-in (El usuario debe crear una nueva contraseña en el siguiente inicio de sesión) para permitir al nuevo usuario restablecer su contraseña después de iniciar sesión.
6. Elija Next: Permissions.
7. En Set permissions (Establecer permisos), elija Add user to group (Añadir usuario a grupo).
8. Elija Create group (Crear grupo).
9. En el cuadro de diálogo Create group (Crear grupo), en Group name (Nombre del grupo) escriba **Administrators**.
10. Elija Filter policies (Filtrar políticas) y, a continuación, seleccione AWS managed -job function (Función de trabajo administrada por AWS) para filtrar el contenido de la tabla.
11. En la lista de políticas, active la casilla de verificación AdministratorAccess. A continuación, elija Create group (Crear grupo).

Note

Debe activar el acceso de usuarios y roles de IAM a Facturación para poder utilizar los permisos **AdministratorAccess** para el acceso a la consola de AWS Billing and Cost Management. Para ello, siga las instrucciones que se indican en el [paso 1 del tutorial sobre cómo delegar el acceso a la consola de facturación](#).

12. Retroceda a la lista de grupos y active la casilla de verificación del nuevo grupo. Elija Refresh si es necesario para ver el grupo en la lista.
13. Elija Next: Tags (Siguiente: Etiquetas).
14. (Opcional) Añadir metadatos al rol asociando las etiquetas como pares de clave-valor. Para obtener más información sobre el uso de etiquetas en IAM, consulte [Etiquetado de entidades de IAM](#) en la Guía del usuario de IAM.
15. Elija Next: Review para ver la lista de suscripciones a grupos que se van a añadir al nuevo usuario. Cuando esté listo para continuar, elija Create user (Crear usuario).

Puede usar este mismo proceso para crear más grupos y usuarios y para conceder a los usuarios acceso a los recursos de la cuenta de AWS. Para obtener información sobre cómo usar las políticas que restringen

los permisos de los usuarios a recursos de AWS específicos, consulte [Administración de acceso y Políticas de ejemplo](#).

Para iniciar sesión como este nuevo usuario de IAM, cierre la sesión de la consola de AWS y después utilice la siguiente dirección URL, donde `su_id_de_cuenta_de_aws` es su número de cuenta de AWS sin los guiones (por ejemplo, si su número de cuenta de AWS es 1234-5678-9012, su ID de cuenta de AWS será 123456789012):

```
https://your_aws_account_id.signin.amazonaws.com/console/
```

Escriba el nombre y la contraseña del usuario de IAM que acaba de crear. Cuando haya iniciado sesión, en la barra de navegación se mostrará "su\_nombre\_de\_usuario @ su\_id\_de\_cuenta\_de\_aws".

Si no desea que la dirección URL de la página de inicio de sesión contenga el ID de su cuenta de AWS, puede crear un alias de cuenta. En el panel de IAM, elija Customize (Personalizar) y escriba un alias, como el nombre de su empresa. Para iniciar sesión después de crear un alias de cuenta, use la siguiente dirección URL.

```
https://your_account_alias.signin.amazonaws.com/console/
```

Para comprobar el enlace de inicio de sesión de los usuarios de IAM de su cuenta, abra la consola de IAM y compruebe el campo AWS Account Alias (Alias de cuenta de AWS) en el panel.

## Planificación de la migración para AWS Database Migration Service

A la hora de planificar la migración de una base de datos con AWS Database Migration Service, tenga en cuenta lo siguiente:

- Tendrá que configurar una red que conecte las bases de datos de origen y de destino con una instancia de replicación de AWS DMS. Esto puede ser tan sencillo como conectar dos recursos de AWS en la misma VPC como la instancia de replicación con configuraciones más complejas, tales como conectar una base de datos local a una instancia de base de datos de Amazon RDS a través de una VPN. Para obtener más información, consulte [Configuraciones de red para migrar bases de datos \(p. 72\)](#)
- Puntos de enlace de origen y de destino: deberá saber qué información y tablas de la base de datos de origen deben migrarse a la base de datos de destino. AWS DMS admite la migración básica del esquema, incluida la creación de tablas y claves primarias. Sin embargo, AWS DMS no crea automáticamente índices secundarios, claves externas, cuentas de usuario, etc. en la base de datos de destino. Tenga en cuenta que, en función de su motor de base de datos de origen y de destino, es posible que tenga que configurar el registro complementario o modificar otra configuración para una base de datos de destino o de origen. Consulte las secciones [Orígenes para la migración de datos \(p. 90\)](#) y [Destinos para la migración de datos \(p. 157\)](#) para obtener más información.
- Migración de esquemas/código: AWS DMS no realiza conversiones de esquema o de código. Puede utilizar herramientas como Oracle SQL Developer, MySQL Workbench o pgAdmin III para convertir el esquema. Si desea convertir un esquema existente en otro motor de base de datos, puede utilizar AWS Schema Conversion Tool. Puede crear un esquema de destino y también generar y crear un esquema completo: tablas, índices, vistas y así sucesivamente. También puede utilizar la herramienta para convertir PL/SQL o TSQL en PostgreSQL y otros formatos. Para obtener más información sobre AWS Schema Conversion Tool, consulte la sección sobre [AWS Schema Conversion Tool](#).
- Tipos de datos no soportados: algunos tipos de datos de origen deben convertirse en los tipos de datos equivalentes para la base de datos de destino. Consulte la sección de origen o de destino para su almacén de datos si desea obtener más información sobre los tipos de datos admitidos.

# Introducción a AWS Database Migration Service

AWS Database Migration Service (AWS DMS) le ayuda a migrar bases de datos a AWS de manera sencilla y segura. Puede migrar datos hacia y desde la mayoría de las bases de datos comerciales y de código abierto más utilizadas, como Oracle, MySQL y PostgreSQL. Este servicio admite migraciones homogéneas como de Oracle a Oracle, así como migraciones heterogéneas entre diferentes plataformas de bases de datos, como de Oracle a PostgreSQL o de MySQL a Oracle.

Para obtener más información sobre el costo de migración de la base de datos utilizando AWS Database Migration Service, consulte la [página de precios de AWS Database Migration Service](#).

## Temas

- [Iniciar una migración de bases de datos con AWS Database Migration Service \(p. 19\)](#)
- [Paso 1: Bienvenida \(p. 19\)](#)
- [Paso 2: Crear una instancia de replicación \(p. 20\)](#)
- [Paso 3: Especificar los puntos de enlace de origen y destino \(p. 25\)](#)
- [Paso 4: Crear una tarea \(p. 30\)](#)
- [Supervisar la tarea \(p. 34\)](#)

## Iniciar una migración de bases de datos con AWS Database Migration Service

Hay varias formas de empezar a migrar una base de datos. Puede seleccionar el asistente de la consola de AWS DMS que le guiará en el proceso o puede realizar cada uno de los pasos si selecciona la tarea correspondiente en el panel de navegación. También puede utilizar la CLI de AWS; para obtener información sobre cómo utilizar la CLI con AWS DMS, consulte [CLI de AWS para AWS DMS](#).

Para utilizar el asistente, seleccione Getting started (Introducción) en el panel de navegación de la consola de AWS DMS. Puede utilizar el asistente para crear su primera migración de datos. Cuando termine el asistente, podrá asignar una instancia de replicación que realiza todos los procesos de la migración, especificar una base de datos de origen y de destino y, a continuación, creará una tarea o conjunto de tareas para definir qué tablas y procesos de replicación desea utilizar. A continuación, AWS DMS crea la instancia de replicación y se encarga de realizar las tareas en los datos que se van a migrar.

También puede crear cada uno de los componentes de una migración de base de datos de AWS DMS si selecciona los elementos en panel de navegación. Para migrar una base de datos, debe hacer lo siguiente:

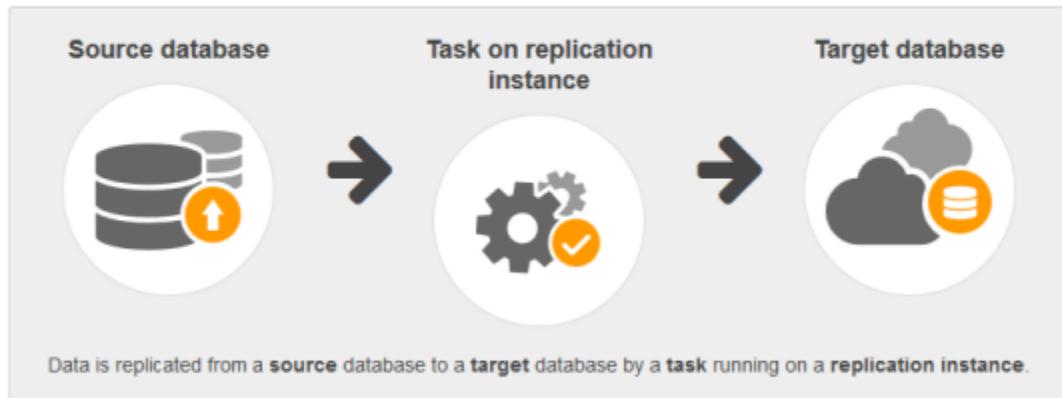
- Completar las tareas que se describen en [Configuración para AWS Database Migration Service \(p. 16\)](#)
- Asignar una instancia de replicación que efectúe todos los procesos para la migración
- Especificar un punto de enlace de base de datos de origen y de destino
- Crear una tarea o conjunto de tareas para definir qué tablas y procesos de replicación desea utilizar

## Paso 1: Bienvenida

Si inicia la migración de la base de datos con el asistente de la consola de AWS DMS, aparecerá la página de bienvenida, donde se explica el proceso de migración de bases de datos con AWS DMS.

## Welcome to AWS Database Migration Service

AWS Database Migration Service tasks require at least a source, a target, and a replication instance. Your source is the database you wish to move data from and the target is the database you're moving data to. The replication instance processes the migration tasks and requires access to your source and target endpoints inside your VPC. Replication instances come in different sizes depending on your performance needs. If you're migrating to a different database engine, AWS Schema Conversion Tool can generate the new schema for you. [Download AWS Schema Conversion Tool](#)



[Cancel](#) [Back](#) [Next](#)

Para iniciar la migración de una base de datos desde la página de bienvenida de la consola

- Seleccione Next (Siguiente).

## Paso 2: Crear una instancia de replicación

La primera tarea a la hora de migrar una base de datos consiste en crear una instancia de replicación que tenga almacenamiento y capacidad de procesamiento suficientes para realizar las tareas que se asignan y migrar datos desde la base de datos de origen a la base de datos de destino. El tamaño necesario para esta instancia varía en función de la cantidad de datos que deba migrar y las tareas que necesita que efectúe la instancia. Para obtener más información sobre las instancias de replicación, consulte [Trabajar con una instancia de replicación de AWS DMS \(p. 62\)](#).

En el procedimiento siguiente, se presupone que va a utilizar el asistente de la consola de AWS DMS. Tenga en cuenta que también puede efectuar este paso si selecciona Replication instances (Instancias de replicación) en el panel de navegación de la consola de AWS DMS y, a continuación, selecciona Create replication instance (Crear instancia de replicación).

Para crear una instancia de replicación con la consola de AWS

1. En el panel de navegación, haga clic en Replication instances.
2. Seleccione Create replication instance.
3. En la página Create replication instance especifique la información de la instancia de replicación. La tabla siguiente describe la configuración.

## Create replication instance

A replication instance initiates the connection between the source and target databases, transfers the data, and caches any changes that occur on the source database during the initial data load. Use the fields below to configure the parameters of your new replication instance including network and security information, encryption details, and performance characteristics. We suggest you shut down the replication instance once your migration is complete to prevent further usage charges.

Name*	e.g. production-replication-server	
Description*	e.g. migrates prod data	
Instance class*	dms.t2.medium	
Replication engine version*	2.3.0	
VPC*	vpc-	
Multi-AZ	No	
Publicly accessible	<input checked="" type="checkbox"/>	

Para esta opción	Haga lo siguiente
Nombre	Escriba un nombre para la instancia de replicación que contenga entre 8 y 16 caracteres ASCII imprimibles (excluidos /, " y @). El nombre de la cuenta debe ser único en la región que haya seleccionado. Puede optar por asignar un nombre descriptivo, como incluir la región y la tarea que esté realizando, por ejemplo <b>west2-mysql2mysql-instance1</b> .
Descripción	Escriba una breve descripción para la instancia de replicación.
Instance class	Elija una clase de instancia con la configuración que necesita para la migración. Tenga en cuenta que la instancia debe tener suficiente capacidad de almacenamiento, red y procesamiento para realizar correctamente la migración. Para obtener más información sobre cómo determinar qué clase de instancia es la mejor opción para la migración, consulte <a href="#">Trabajar con una instancia de replicación de AWS DMS (p. 62)</a> .
Replication engine version	De forma predeterminada, la instancia de replicación ejecuta la última versión del software del motor de replicación de AWS DMS. Le recomendamos que acepte este valor predeterminado; no obstante, puede elegir una versión de motor anterior en caso de que sea necesario.

Para esta opción	Haga lo siguiente
VPC	Seleccione la Amazon Virtual Private Cloud (Amazon VPC) que desee utilizar. Si la base de datos de origen o de destino se encuentra en una VPC, seleccione esa VPC. Si las bases de datos de origen y de destino se encuentran en diferentes VPC, asegúrese de que ambas se encuentren en subredes públicas y son de acceso público y, a continuación, seleccione la VPC donde se va a ubicar la instancia de replicación. Es preciso que la instancia de replicación tenga acceso a los datos en la VPC de origen. Si las bases de datos de origen o destino no se encuentran en una VPC, seleccione la VPC donde se vaya a ubicar la instancia de replicación.
Varias AZ	Use este parámetro opcional para crear una réplica en espera de la instancia de replicación en otra zona de disponibilidad para el soporte de comutación por error. Si va a utilizar la captura de datos de cambios (CDC) o la replicación continua, debe habilitar esta opción.
Publicly accessible	Seleccione esta opción si desea que la instancia de replicación esté accesible desde Internet.

4. Seleccione la pestaña Advanced que se muestra más abajo para establecer valores para la configuración de red y cifrado de la red en caso de que lo necesite. La tabla siguiente describe la configuración.

▼ Advanced

Allocated storage (GB)\*

Replication Subnet Group\*

Availability zone\*

VPC Security Group(s)   
rds-launch-wizard-1  
rds-launch-wizard  
rds-[REDACTED]

KMS master key

Description Default master key that protects my  
DMS replication instance volumes when  
no other key is defined

Account

Key ARN

Para esta opción	Haga lo siguiente
Allocated storage (GB)	<p>La capacidad de almacenamiento la consumen básicamente los archivos de registro y las transacciones que se almacenan en la memoria caché. En el caso de las transacciones en la memoria caché, el almacenamiento se utiliza únicamente cuando las transacciones en memoria caché se deben escribir en el disco. Por lo tanto, AWS DMS no utiliza una cantidad significativa de almacenamiento, salvo en determinadas excepciones como las siguientes:</p> <ul style="list-style-type: none"> <li>• Tablas muy grandes que conllevan una carga de transacciones importante. Cargar una tabla muy grande puede llevar su tiempo, por tanto hay más probabilidades de que las transacciones almacenadas en la memoria caché se escriban en el disco cuando se carga una tabla de gran tamaño.</li> <li>• Las tareas se configuran para detenerse antes de cargar las transacciones en caché. En este caso, todas las transacciones se almacenan en la memoria caché hasta que finaliza la carga completa de todas las tablas. Con esta configuración, es posible que las transacciones en la memoria caché consuman una cantidad considerable de espacio de almacenamiento.</li> <li>• Tareas configuradas con tablas que se cargan en Amazon Redshift. Sin embargo, esta configuración no comporta problema alguno si Amazon Aurora es el destino.</li> </ul> <p>En la mayoría de los casos, la asignación de almacenamiento es suficiente. Sin embargo, es siempre aconsejable prestar atención a las métricas relativas al almacenamiento y ampliar la capacidad de almacenamiento si determina que el consumo es mayor de lo que se ha asignado de forma predeterminada.</p>
Replication Subnet Group	Elija el grupo de subred de replicación en la VPC que haya seleccionado si quiere que se cree la instancia de replicación. Si la base de datos de origen está en una VPC, seleccione el grupo de subred que contiene la base de datos de origen como ubicación para la instancia de replicación. Para obtener más información sobre los grupos de subred para la replicación, consulte <a href="#">Creación de un grupo de subred de replicación (p. 77)</a> .
Availability zone	Seleccione la zona de disponibilidad en la que se encuentra la base de datos de origen.
VPC Security group(s)	La instancia de replicación se crea en una VPC. Si la base de datos de origen está en una VPC, seleccione el grupo de seguridad de la VPC que proporciona acceso a la instancia de base de datos donde reside la base de datos.

Para esta opción	Haga lo siguiente
Clave maestra de KMS	Elija la clave de cifrado que se utilizará para cifrar el almacenamiento de la replicación y la información de la conexión. Si elige (Default) aws/dms, se utilizará la clave de AWS Key Management Service (AWS KMS) asociada con su cuenta y región. Se muestran una descripción y su número de cuenta junto con el ARN de la clave. Para obtener más información sobre cómo utilizar la clave de cifrado, consulte <a href="#">Configuración de una clave de cifrado y especificación de permisos KMS (p. 49)</a> .

5. Especifique la configuración Maintenance. La tabla siguiente describe la configuración. Para obtener más información sobre la configuración de mantenimiento, consulte [Ventana de mantenimiento de AWS DMS \(p. 66\)](#).

▼ Maintenance

The screenshot shows the 'Maintenance' configuration section of the AWS DMS 'Create replication instance' wizard. It includes a checkbox for 'Auto minor version upgrade' which is checked. Below it is a 'Maintenance window' configuration with dropdowns for day ('Tuesday'), hour ('03'), and minutes ('0.5 (UTC-7)'). A note indicates a 53-hour window. At the bottom right are 'Cancel' and 'Create replication instance' buttons.

Para esta opción	Haga lo siguiente
Auto minor version upgrade	Seleccione esta opción para que las actualizaciones de motor secundarias se apliquen automáticamente a la instancia de replicación durante el periodo de mantenimiento.
Maintenance window	Seleccione el intervalo de tiempo semanal durante el cual puede llevarse a cabo el mantenimiento del sistema, en tiempo universal coordinado (UTC).  Predeterminado: una ventana de 30 minutos seleccionada al azar de un bloque de 8 horas de tiempo por región, que suceden un día de la semana al azar.

6. Elija Create replication instance.

## Paso 3: Especificar los puntos de enlace de origen y destino

Mientras se está creando la instancia de replicación, puede especificar los almacenes de datos de origen y de destino. Los almacenes de datos de origen y de destino pueden residir en una instancia de Amazon

Elastic Compute Cloud (Amazon EC2), una instancia de base de datos de Amazon Relational Database Service (Amazon RDS) o en una base de datos local.

En el procedimiento siguiente, se presupone que va a utilizar el asistente de la consola de AWS DMS. Tenga en cuenta que también puede efectuar este paso si selecciona Endpoints (Puntos de enlace) en el panel de navegación de la consola de AWS DMS y, a continuación, selecciona Create endpoint (Crear punto de enlace). Cuando se utiliza el asistente de la consola, debe crear los puntos de enlace de origen y de destino en la misma página. Si no utiliza el asistente de la consola, debe crear cada uno de los puntos de enlace por separado.

Para especificar los puntos de enlace de la base de datos de origen o de destino a con la consola de AWS

1. En la página Connect source and target database endpoints, especifique la información de conexión para la base de datos de origen o destino. La tabla siguiente describe la configuración.

## Connect source and target database endpoints

Your replication instance is being created. You can start entering your endpoint connection details now. Once created, you can test your endpoint connections and move on to task creation.

Your database endpoint can be on-premise, in EC2, RDS or in the cloud. Define the connection details below. It is recommended that you test your endpoint connections here to avoid errors later.

Source database connection details	Target database connection details
Endpoint identifier* <input type="text" value="ProdEndpoint"/>	Endpoint identifier* <input type="text" value="TestEndpoint"/>
Source engine* <input type="button" value="Select One -"/>	Target engine* <input type="button" value="Select One -"/>
Server name* <input type="text"/>	Server name* <input type="text"/>
Port* <input type="button"/>	Port* <input type="button"/>
SSL mode* <input type="button"/>	SSL mode* <input type="button"/>
User name* <input type="text"/>	User name* <input type="text"/>
Password* <input type="text"/>	Password* <input type="text"/>
» Advanced	
<input type="button" value="Run test"/>	
<input type="button" value="Cancel"/> <input type="button" value="Previous"/> <input type="button" value="Next"/>	

Para esta opción	Haga lo siguiente
Endpoint identifier	Escriba el nombre que desea utilizar para identificar el punto de enlace. Es posible que le interese incluir en el nombre del tipo de punto de enlace, como <b>oracle-source</b> o <b>PostgreSQL-target</b> . El nombre debe ser único para todas las instancias de replicación.

Para esta opción	Haga lo siguiente
Source engine y Target engine	Elija el tipo de motor de base de datos que vaya a ser el punto de enlace.
Server name	Escriba el nombre del servidor. Para las bases de datos locales, puede ser la dirección IP o el nombre de host público. Para una instancia de base de datos Amazon RDS, puede ser el punto de enlace (también denominado nombre de DNS) de la instancia de base de datos, como <b>mysqlsvrinst.abcd12345678.us-west-2.rds.amazonaws.com</b> .
Puerto	Escriba el puerto que utiliza la base de datos.
SSL mode	Elija un modo SSL si desea habilitar el cifrado de conexión para este punto de enlace. En función del modo seleccionado, es posible que se le solicite proporcionar información sobre el certificado y el certificado de servidor.
Nombre de usuario	Escriba el nombre de usuario con los permisos necesarios para permitir la migración de datos. Para obtener información sobre los permisos necesarios, consulte la sección de seguridad para el motor de base de datos de origen o destino en esta guía del usuario.
Contraseña	Escriba la contraseña para la cuenta con los permisos necesarios. Si desea utilizar caracteres especiales en la contraseña, como "+" o "&", incluya la totalidad de la contraseña entre llaves "{}".

2. Seleccione la pestaña Advanced que se muestra a continuación para establecer los valores para la cadena de conexión y la clave de cifrado en caso de que sea necesario. Puede probar la conexión del punto de enlace si selecciona Run test.

<p><b>▼ Advanced</b></p> <p>Extra connection attributes <input type="text"/></p> <p>KMS master key <input type="text" value="Default aws/dms"/> ⓘ</p> <p><b>Description</b> Default master key that protects my DMS replication instance volumes when no other key is defined</p> <p>Account <input type="text" value="XXXXXXXXXXXXXX"/></p> <p>Key ARN <input type="text" value="arn:aws:kms:us-west-2:XXXXXXXXXXXXXX:key/7944f0ec-XXXX-XXXX-XXXX-XXXX-XXXXXX"/></p> <p><b>Run test</b></p>	<p><b>▼ Advanced</b></p> <p>Extra connection attributes <input type="text"/></p> <p>KMS master key <input type="text" value="Default aws/dms"/> ⓘ</p> <p><b>Description</b> Default master key that protects my DMS replication instance volumes when no other key is defined</p> <p>Account <input type="text" value="XXXXXXXXXXXXXX"/></p> <p>Key ARN <input type="text" value="arn:aws:kms:us-west-2:XXXXXXXXXXXXXX:key/7944f0ec-XXXX-XXXX-XXXX-XXXXXX"/></p> <p><b>Run test</b></p>
<input type="button" value="Cancel"/> <input type="button" value="Previous"/> <input style="background-color: #0070C0; color: white; font-weight: bold; border-radius: 5px; border: none; padding: 2px 10px;" type="button" value="Next"/>	

Para esta opción	Haga lo siguiente
Atributos de conexión adicionales	Escriba los parámetros de conexión adicionales aquí. Para obtener más información sobre los atributos de conexión adicionales, consulte la sección de documentación relacionada con su almacén de datos.
Clave maestra de KMS	Elija la clave de cifrado que se utilizará para cifrar el almacenamiento de la replicación y la información de la conexión. Si elige (Default) aws/dms , se utilizará la clave de AWS Key Management Service (AWS KMS) asociada con su cuenta y región. Para obtener más información sobre cómo utilizar la clave de cifrado, consulte <a href="#">Configuración de una clave de cifrado y especificación de permisos KMS (p. 49)</a> .

## Paso 4: Crear una tarea

Cree una tarea para especificar qué tablas se van a migrar, asignar los datos mediante un esquema de destino y crear tablas nuevas en la base de datos de destino. Cuando cree las tareas podrá elegir el tipo de migración: migrar datos existentes, migrar datos existentes y replicar los cambios continuos o replicar solamente los cambios en los datos.

Con AWS DMS, puede especificar la asignación precisa de los datos entre la base de datos de origen y la de destino. Antes de especificar las asignaciones, asegúrese de revisar la sección de la documentación sobre las asignaciones de tipos de datos para las bases de datos de origen y de destino.

Puede optar por comenzar una tarea tan pronto como termine de especificar la información para esa tarea en la página Create task o puede comenzar la tarea de la página Dashboard una vez que termine de especificar la información de la tarea.

En el procedimiento siguiente, se presupone que ha elegido el asistente de la consola de AWS DMS y ha especificado la información de la instancia de replicación y los puntos de enlace con el asistente de la consola. Tenga en cuenta que también puede efectuar este paso si selecciona Tasks (Tareas) en el panel de navegación de la consola de AWS DMS y, a continuación, selecciona Create Task (Crear tarea).

Para crear una tarea de migración

1. En la página Create Task especifique las opciones de la tarea. La tabla siguiente describe la configuración.

### Create Task

A task can contain one or more table mappings which define what data is moved from the source to the target. If a table does not exist on the target, it can be created automatically.

<b>Task name</b>	ProdEndpoint-TestEndpoint	
<b>Task description</b>	e.g. migrate customer tables to RD	
<b>Source endpoint</b>	rdsoracle-endpoint	
<b>Target endpoint</b>	rdspostgres-endpoint	
<b>Replication instance</b>	replinstance1-26	
<b>Migration type</b>	Migrate existing data	
<b>Start task on create</b>	<input checked="" type="checkbox"/>	
<a href="#">Task Settings</a>		
<a href="#">Table mappings</a>		

Para esta opción	Haga lo siguiente
Task name	Escriba un nombre para la tarea.
Task description	Escriba una descripción para la tarea.
Punto de enlace de origen	Muestra el punto de enlace de origen que se va a utilizar.
Punto de enlace de destino	Muestra el punto de enlace de destino que se va a utilizar.
Instancia de replicación	Muestra la instancia de replicación que se va a utilizar.
Migration type	Elija el método de migración que desea utilizar. Puede elegir que solo se migren los datos existentes a la base de datos de destino o que se envíen los cambios en curso a la base de datos de destino, además de los datos migrados.
Start task on create	Cuando esta opción esté seleccionada, la tarea comenzará tan pronto como se cree.

2. Seleccione la pestaña Task Settings que se muestra a continuación y especifique los valores para su tabla de destino, la compatibilidad con LOB y habilite el registro. La configuración de la tarea que se muestra dependerá del valor Migration type que seleccione. Por ejemplo, al seleccionar Migrate existing data, se muestran las siguientes opciones:

Task Settings

Target table preparation mode

- Do nothing
- Drop tables on target
- Truncate

Include LOB columns in replication

- Don't include LOB columns
- Full LOB mode
- Limited LOB mode

Max LOB size (kb)\*

32

Enable logging

Para esta opción	Haga lo siguiente
Target table preparation mode	<p>Do nothing: los datos y los metadatos de las tablas de destino no cambiarán.</p> <p>Drop tables on target: las tablas existentes se eliminan y se crean otras para sustituirlas.</p> <p>Truncate: las tablas se truncan sin que los metadatos de la tabla se vean afectados.</p>
Include LOB columns in replication	Don't include LOB columns: las columnas de LOB se excluirán de la migración.

Para esta opción	Haga lo siguiente
	<p>Full LOB mode: se migran los LOB completos independientemente de su tamaño. Los LOB se migran por partes, en porciones que se controlan según el tamaño de la porción del LOB. Este método es más lento que utilizar el modo LOB limitado.</p> <p>Limited LOB mode (Modo LOB limitado): se truncan los LOB al tamaño máximo permitido. Este método es más rápido que utilizar el modo LOB completo.</p>
Max LOB size (kb)	En Limited LOB Mode, las columnas de LOB que superen el valor configurado en Max LOB Size se truncarán según el tamaño máximo especificado para los LOB.
Enable logging (Habilitar el registro)	Habilita los registros de Amazon CloudWatch.

Al seleccionar Migrate existing data and replicate para Migration type, se muestran las siguientes opciones:

▼ Task Settings

**Target table preparation mode**  Do nothing i  
 Drop tables on target i  
 Truncate i

**Stop task after full load completes**  Don't stop i  
 Stop Before Applying Cached Changes i  
 Stop After Applying Cached Changes i

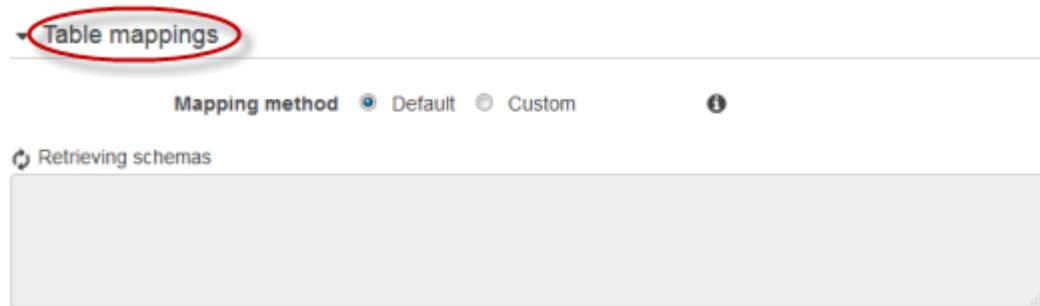
**Include LOB columns in replication**  Don't include LOB columns i  
 Full LOB mode i  
 Limited LOB mode i

**Max LOB size (kb)\***  i

**Enable logging**

Para esta opción	Haga lo siguiente
Target table preparation mode	<p>Do nothing: los datos y los metadatos de las tablas de destino no cambiarán.</p> <p>Drop tables on target: las tablas existentes se eliminan y se crean otras para sustituirlas.</p> <p>Truncate: las tablas se truncan sin que los metadatos de la tabla se vean afectados.</p>
Stop task after full load completes	<p>Don't stop: no interrumpe la tarea, se aplican inmediatamente los cambios en la memoria caché y se continúa.</p> <p>Stop before applying cached changes: se detiene la tarea antes de aplicar los cambios de la memoria caché. Esto le permitirá añadir índices secundarios que pueden agilizar la aplicación de cambios.</p> <p>Stop after applying cached changes: se detiene la tarea después de aplicar los cambios de la memoria caché. Esto le permitirá añadir claves externas, disparadores, etc. si está utilizando el modo de aplicación transaccional.</p>
Include LOB columns in replication	<p>Don't include LOB columns: las columnas de LOB se excluirán de la migración.</p> <p>Full LOB mode: se migran los LOB completos independientemente de su tamaño. Los LOB se migran por partes, en porciones que se controlan según el tamaño de la porción del LOB. Este método es más lento que utilizar el modo LOB limitado.</p> <p>Limited LOB mode (Modo LOB limitado): se truncan los LOB al tamaño máximo permitido. Este método es más rápido que utilizar el modo LOB completo.</p>
Max LOB size (kb)	En Limited LOB Mode, las columnas de LOB que superen el valor configurado en Max LOB Size se truncarán según el tamaño máximo especificado para los LOB.
Enable logging (Habilitar el registro)	Habilita los registros de Amazon CloudWatch.

3. Seleccione la pestaña Table mappings que se muestra a continuación para establecer valores para el mapeo de esquemas y el método de mapeo. Si elige Custom, puede especificar el esquema de destino y los valores de la tabla. Para obtener más información sobre la asignación de tablas, consulte [Uso de la correspondencia de tablas para especificar la configuración de tareas \(p. 283\)](#).



4. Una vez que haya completado la configuración de las tareas, seleccione Create task.

## Supervisar la tarea

Si selecciona Start task on create al crear una tarea, la tarea comienza inmediatamente a migrar datos cuando seleccione Create Task. Puede consultar las estadísticas y supervisar la información para la tarea si selecciona la tarea en ejecución en la consola de administración de AWS. La siguiente captura de pantalla muestra las estadísticas de la tabla de una migración de base de datos. Para obtener más información sobre las supervisiones, consulte [Monitorización de tareas de AWS DMS \(p. 321\)](#)

The screenshot shows the 'Table statistics' tab for a task named 'task-fkvacppsulxnqyv'. The tab is highlighted with a red oval. Below the tab, there's a table with columns: Schema, Table, State, Inserts, Deletes, Updates, DDLs, Full Load Rows, Total, and Last. The table data is as follows:

Schema	Table	State	Inserts	Deletes	Updates	DDLs	Full Load Rows	Total	Last
aat	T20	Full load	0	0	0	0	16,080,394	16,080,394	3/1
aat	T21	Full load	0	0	0	0	16,079,437	16,079,437	3/1
aat	T22	Full load	0	0	0	0	15,804,000	15,804,000	3/1

# Seguridad para AWS Database Migration Service

AWS Database Migration Service (AWS DMS) utiliza varios procesos para proteger los datos durante la migración. Este servicio se encarga de cifrar el almacenamiento utilizado por la instancia de replicación y la información de conexión del punto de enlace utilizando una clave AWS Key Management Service (AWS KMS) exclusiva para su cuenta de AWS. Es compatible con la capa de conexión segura (SSL). AWS Database Migration Service también requiere que tenga los permisos pertinentes si inicia sesión como usuario de AWS Identity and Access Management (IAM).

La VPC basada en el servicio Amazon Virtual Private Cloud (Amazon VPC) que utiliza con su instancia de replicación debe asociarse a un grupo de seguridad que disponga de normas que permitan que todo el tráfico en todos los puertos salga de la VPC. Este enfoque permite la comunicación entre la instancia de replicación y los puntos de enlace de su base de datos de origen y de destino, siempre que en dichos puntos de enlace se haya activado la entrada correcta.

Si desea ver los registros de migración de la base de datos, necesita los permisos de registros de Amazon CloudWatch para el rol de IAM que está utilizando.

## Temas

- [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#)
- [Creación de los roles de IAM para usar con la AWS CLI y la API de AWS DMS \(p. 38\)](#)
- [Control de acceso detallado mediante nombres de recursos y etiquetas \(p. 42\)](#)
- [Configuración de una clave de cifrado y especificación de permisos KMS \(p. 49\)](#)
- [Seguridad de red para AWS Database Migration Service \(p. 50\)](#)
- [Uso de SSL con AWS Database Migration Service \(p. 51\)](#)
- [Cambio de la contraseña de la base de datos \(p. 60\)](#)

## Permisos de IAM necesarios para utilizar AWS DMS

Para utilizar AWS DMS se usan determinados permisos y roles de IAM. Si está registrado como usuario de IAM y desea utilizar AWS DMS, el administrador de la cuenta debe adjuntar la política tratada en esta sección al usuario, grupo o rol de IAM que utilice para ejecutar AWS DMS. Para obtener más información sobre los permisos de IAM, consulte la [Guía del usuario de IAM](#).

El siguiente conjunto de permisos le da acceso a AWS DMS, así como permisos para determinadas acciones necesarias de otros servicios de Amazon, como AWS KMS, IAM, Amazon Elastic Compute Cloud (Amazon EC2) y Amazon CloudWatch. CloudWatch monitoriza su migración de AWS DMS en tiempo real y recopila y realiza un seguimiento de las métricas que indican el progreso de su migración. Puede utilizar los registros de CloudWatch para depurar problemas con una tarea.

### Note

Puede restringir aún más el acceso a recursos de AWS DMS mediante el etiquetado. Para obtener más información sobre la restricción del acceso a recursos de AWS DMS mediante el etiquetado, consulte [Control de acceso detallado mediante nombres de recursos y etiquetas \(p. 42\)](#)

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "dms:*",  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "kms>ListAliases",  
                "kms>DescribeKey"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "iam>GetRole",  
                "iam>PassRole",  
                "iam>CreateRole",  
                "iam>AttachRolePolicy"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2>DescribeVpcs",  
                "ec2>DescribeInternetGateways",  
                "ec2>DescribeAvailabilityZones",  
                "ec2>DescribeSubnets",  
                "ec2>DescribeSecurityGroups",  
                "ec2>ModifyNetworkInterfaceAttribute",  
                "ec2>CreateNetworkInterface",  
                "ec2>DeleteNetworkInterface"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "cloudwatch>Get*",  
                "cloudwatch>List*"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "logs>DescribeLogGroups",  
                "logs>DescribeLogStreams",  
                "logs>FilterLogEvents",  
                "logs>GetLogEvents"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "redshift>Describe*",  
                "redshift>ModifyClusterIamRoles"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

]

Un desglose de estos permisos podría ayudarle a entender mejor por qué cada uno de ellos es necesario.

La sección siguiente es necesaria para permitir al usuario llamar a las operaciones de la API de AWS DMS.

```
{  
    "Effect": "Allow",  
    "Action": "dms:*",  
    "Resource": "*"  
}
```

La siguiente sección es necesaria para permitir al usuario enumerar sus claves de AWS KMS y alias disponibles para su visualización en la consola. Esta entrada no es necesaria si conoce el nombre de recurso de Amazon (ARN) para la clave de KMS y está utilizando solo los AWS Command Line Interface (AWS CLI).

```
{  
    "Effect": "Allow",  
    "Action": [  
        "kms>ListAliases",  
        "kms>DescribeKey"  
    ],  
    "Resource": "*"  
}
```

La sección siguiente es necesaria para determinados tipos de punto de enlace que requieren que se pase un ARN del rol con el punto de enlace. Además, si no se crean los roles de AWS DMS por adelantado, la consola AWS DMS tiene la capacidad de crear el rol. Si todas las funciones se configuran con antelación, todo eso es necesario en `iam:GetRole` e `iam:PassRole`. Para obtener más información acerca de los roles de , consulte [Creación de los roles de IAM para usar con la AWS CLI y la API de AWS DMS \(p. 38\)](#).

```
{  
    "Effect": "Allow",  
    "Action": [  
        "iam:GetRole",  
        "iam:PassRole",  
        "iam>CreateRole",  
        "iam:AttachRolePolicy"  
    ],  
    "Resource": "*"  
}
```

La sección siguiente es necesaria dado que AWS DMS tiene que crear la instancia Amazon EC2 y configurar la red para la instancia de replicación que se crea. Estos recursos existen en la cuenta del cliente, por lo que la capacidad para realizar estas acciones en nombre del cliente es necesaria.

```
    "Effect": "Allow",
    "Action": [
        "ec2:DescribeVpcs",
        "ec2:DescribeInternetGateways",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
```

```
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2>CreateNetworkInterface",
        "ec2>DeleteNetworkInterface"
    ],
    "Resource": "*"
}
```

La siguiente sección es necesaria para permitir que el usuario pueda ver las métricas de instancia de replicación.

```
{
    "Effect": "Allow",
    "Action": [
        "cloudwatch:Get*",
        "cloudwatch>List*"
    ],
    "Resource": "*"
}
```

Esta sección es necesaria para permitir que el usuario vea los registros de replicación.

```
{
    "Effect": "Allow",
    "Action": [
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams",
        "logs:FilterLogEvents",
        "logs:GetLogEvents"
    ],
    "Resource": "*"
}
```

La sección siguiente es necesaria cuando se utiliza Amazon Redshift como destino. Permite a AWS DMS validar que el clúster de Amazon Redshift está configurado correctamente para AWS DMS.

```
{
    "Effect": "Allow",
    "Action": [
        "redshift:Describe*",
        "redshift:ModifyClusterIamRoles"
    ],
    "Resource": "*"
}
```

La consola de AWS DMS crea varios roles que se adjuntan automáticamente a su cuenta de AWS al utilizar la consola de AWS DMS. Si utiliza la AWS Command Line Interface (AWS CLI) o la API de AWS DMS para la migración, tendrá que añadir dichos roles a su cuenta. Para obtener más información sobre cómo añadir roles, consulte [Creación de los roles de IAM para usar con la AWS CLI y la API de AWS DMS \(p. 38\)](#).

## Creación de los roles de IAM para usar con la AWS CLI y la API de AWS DMS

Si utiliza la AWS CLI o la API de AWS DMS para migrar su base de datos, debe añadir tres roles de IAM a su cuenta de AWS para poder utilizar las funciones de AWS DMS. Dos de los tres roles son `dms-vpc-role` y `dms-cloudwatch-logs-role`. Si utiliza Amazon Redshift como base de datos de destino, deberá añadir el rol de IAM `dms-access-for-endpoint` a su cuenta de AWS.

Las actualizaciones de las políticas administradas son automáticas. Si utiliza una política personalizada con los roles de IAM, asegúrese de comprobar de forma periódica las actualizaciones de la política administrada en esta documentación. Puede ver los detalles de la política administrada usando una combinación de los comandos `get-policy` y `get-policy-version`.

Por ejemplo, el siguiente comando `get-policy` obtiene información acerca del rol.

```
aws iam get-policy --policy-arn arn:aws:iam::aws:policy/service-role/  
AmazonDMSVPCManagementRole
```

El comando devuelve la siguiente información.

```
{  
    "Policy": {  
        "PolicyName": "AmazonDMSVPCManagementRole",  
        "Description": "Provides access to manage VPC settings for AWS managed customer  
configurations",  
        "CreateDate": "2015-11-18T16:33:19Z",  
        "AttachmentCount": 1,  
        "IsAttachable": true,  
        "PolicyId": "ANPAJHKIGMBQI4AEFFSYO",  
        "DefaultVersionId": "v3",  
        "Path": "/service-role/",  
        "Arn": "arn:aws:iam::aws:policy/service-role/AmazonDMSVPCManagementRole",  
        "UpdateDate": "2016-05-23T16:29:57Z"  
    }  
}
```

El siguiente comando `get-policy-version` obtiene información de políticas.

```
aws iam get-policy-version --policy-arn arn:aws:iam::aws:policy/service-role/  
AmazonDMSVPCManagementRole --version-id v3
```

El comando devuelve la siguiente información.

```
{  
    "PolicyVersion": {  
        "CreateDate": "2016-05-23T16:29:57Z",  
        "VersionId": "v3",  
        "Document": {  
            "Version": "2012-10-17",  
            "Statement": [  
                {  
                    "Action": [  
                        "ec2:CreateNetworkInterface",  
                        "ec2:DescribeAvailabilityZones",  
                        "ec2:DescribeInternetGateways",  
                        "ec2:DescribeSecurityGroups",  
                        "ec2:DescribeSubnets",  
                        "ec2:DescribeVpcs",  
                        "ec2:DeleteNetworkInterface",  
                        "ec2:ModifyNetworkInterfaceAttribute"  
                    ],  
                    "Resource": "*"  
                }  
            ]  
        }  
    }  
}
```

```
        "Effect": "Allow"
    }
],
"IsDefaultVersion": true
}
```

Se pueden utilizar los mismos comandos para obtener información sobre `AmazonDMSCloudWatchLogsRole` y la política administrada `AmazonDMSRedshiftS3Role`.

**Note**

Si utiliza la consola de AWS DMS para migrar la base de datos, estos roles se añadirán a su cuenta de AWS automáticamente.

Los siguientes procedimientos crean los roles de IAM `dms-vpc-role`, `dms-cloudwatch-logs-role` y `dms-access-for-endpoint`.

Para crear el rol de IAM `dms-vpc-role` para usarlo en la AWS CLI o la API de AWS DMS

1. Cree un archivo JSON con la siguiente política de IAM. Asigne el nombre al archivo JSON `dmsAssumeRolePolicyDocument.json`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "dms.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Cree el rol en la AWS CLI utilizando el siguiente comando.

```
aws iam create-role --role-name dms-vpc-role --assume-role-policy-document file://
dmsAssumeRolePolicyDocument.json
```

2. Adjunte la política `AmazonDMSVPCManagementRole` a `dms-vpc-role` utilizando el siguiente comando.

```
aws iam attach-role-policy --role-name dms-vpc-role --policy-arn
arn:aws:iam::aws:policy/service-role/AmazonDMSVPCManagementRole
```

Para crear el rol de IAM `dms-cloudwatch-logs-role` para usarlo en la AWS CLI o la API de AWS DMS

1. Cree un archivo JSON con la siguiente política de IAM. Asigne el nombre al archivo JSON `dmsAssumeRolePolicyDocument2.json`.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": {  
                "Service": "dms.amazonaws.com"  
            },  
            "Action": "sts:AssumeRole"  
        }  
    ]  
}
```

Cree el rol en la AWS CLI utilizando el siguiente comando.

```
aws iam create-role --role-name dms-cloudwatch-logs-role --assume-role-policy-document  
file://dmsAssumeRolePolicyDocument2.json
```

2. Adjunte la política `AmazonDMSCloudWatchLogsRole` a `dms-cloudwatch-logs-role` utilizando el siguiente comando.

```
aws iam attach-role-policy --role-name dms-cloudwatch-logs-role --policy-arn  
arn:aws:iam::aws:policy/service-role/AmazonDMSCloudWatchLogsRole
```

Si utiliza Amazon Redshift como base de datos de destino, debe crear el rol de IAM `dms-access-for-endpoint` para proporcionar acceso a Amazon Simple Storage Service (Amazon S3).

Para crear el rol de IAM `dms-access-for-endpoint` para usarlo en Amazon Redshift como base de datos de destino

1. Cree un archivo JSON con la siguiente política de IAM. Asigne el nombre al archivo JSON `dmsAssumeRolePolicyDocument3.json`.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "1",  
            "Effect": "Allow",  
            "Principal": {  
                "Service": "dms.amazonaws.com"  
            },  
            "Action": "sts:AssumeRole"  
        },  
        {  
            "Sid": "2",  
            "Effect": "Allow",  
            "Principal": {  
                "Service": "redshift.amazonaws.com"  
            },  
            "Action": "sts:AssumeRole"  
        }  
    ]  
}
```

```
    ]  
}
```

- Cree el rol en la AWS CLI utilizando el siguiente comando.

```
aws iam create-role --role-name dms-access-for-endpoint --assume-role-policy-document  
file://dmsAssumeRolePolicyDocument3.json
```

- Adjunte la política AmazonDMSRedshiftS3Role al rol dms-access-for-endpoint utilizando el siguiente comando.

```
aws iam attach-role-policy --role-name dms-access-for-endpoint \  
--policy-arn arn:aws:iam::aws:policy/service-role/AmazonDMSRedshiftS3Role
```

Ahora debería disponer de las políticas de IAM necesarias para utilizar la AWS CLI o la API de AWS DMS.

## Control de acceso detallado mediante nombres de recursos y etiquetas

Puede utilizar nombres de recursos basados en ARN y etiquetas de recursos para administrar el acceso a recursos de AWS DMS. Para ello, define la acción permitida o incluye declaraciones condicionales en las políticas de IAM.

### Uso de nombres de recursos para controlar el acceso

Puede crear una cuenta de usuario de IAM y asignar una política basada en el Nombre de recurso de Amazon (ARN) del recurso de AWS DMS.

La siguiente política deniega el acceso a la instancia de replicación AWS DMS con el ARN arn:aws:dms:us-east-1:152683116:rep:DOH67ZTOXGLIXMIHKITV:

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Action": [  
                "dms:*"  
            ],  
            "Effect": "Deny",  
            "Resource": "arn:aws:dms:us-east-1:152683116:rep:DOH67ZTOXGLIXMIHKITV"  
        }  
    ]  
}
```

Por ejemplo, los siguientes comandos fallarían cuando la política está en vigor:

```
$ aws dms delete-replication-instance  
--replication-instance-arn "arn:aws:dms:us-east-1:152683116:rep:DOH67ZTOXGLIXMIHKITV"
```

```
A client error (AccessDeniedException) occurred when calling the DeleteReplicationInstance operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform: dms>DeleteReplicationInstance on resource: arn:aws:dms:us-east-1:152683116:rep:DOH67ZTOXGLIXMIHKITV

$ aws dms modify-replication-instance
--replication-instance-arn "arn:aws:dms:us-east-1:152683116:rep:DOH67ZTOXGLIXMIHKITV"

A client error (AccessDeniedException) occurred when calling the ModifyReplicationInstance operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform: dms:ModifyReplicationInstance on resource: arn:aws:dms:us-east-1:152683116:rep:DOH67ZTOXGLIXMIHKITV
```

También puede especificar políticas de IAM que limiten el acceso a los puntos de enlace de AWS DMS y las tareas de replicación.

La siguiente política limita el acceso a puntos de enlace de AWS DMS utilizando el ARN del punto de enlace:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "dms:*"
            ],
            "Effect": "Deny",
            "Resource": "arn:aws:dms:us-east-1:152683116:endpoint:D6E37YBXTNHOA6XRQSZCUGX"
        }
    ]
}
```

Por ejemplo, los siguientes comandos fallarían cuando la política que utiliza el ARN del punto de enlace está en vigor:

```
$ aws dms delete-endpoint
--endpoint-arn "arn:aws:dms:us-east-1:152683116:endpoint:D6E37YBXTNHOA6XRQSZCUGX"

A client error (AccessDeniedException) occurred when calling the DeleteEndpoint operation:
User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms>DeleteEndpoint
on resource: arn:aws:dms:us-east-1:152683116:endpoint:D6E37YBXTNHOA6XRQSZCUGX

$ aws dms modify-endpoint
--endpoint-arn "arn:aws:dms:us-east-1:152683116:endpoint:D6E37YBXTNHOA6XRQSZCUGX"

A client error (AccessDeniedException) occurred when calling the ModifyEndpoint operation:
User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms:ModifyEndpoint
on resource: arn:aws:dms:us-east-1:152683116:endpoint:D6E37YBXTNHOA6XRQSZCUGX
```

La siguiente política limita el acceso a una tarea de AWS DMS utilizando el ARN de la tarea:

```
{
    "Version": "2012-10-17",
```

```
"Statement": [
    {
        "Action": [
            "dms:*"
        ],
        "Effect": "Deny",
        "Resource": "arn:aws:dms:us-east-1:152683116:task:UO3YR4N47DXH3ATT4YMWOIT"
    }
]
```

Por ejemplo, los siguientes comandos fallarían cuando la política que utiliza el ARN de la tarea está en vigor:

```
$ aws dms delete-replication-task
--replication-task-arn "arn:aws:dms:us-east-1:152683116:task:UO3YR4N47DXH3ATT4YMWOIT"

A client error (AccessDeniedException) occurred when calling the DeleteReplicationTask
operation:
User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms:DeleteReplicationTask
on resource: arn:aws:dms:us-east-1:152683116:task:UO3YR4N47DXH3ATT4YMWOIT
```

## Uso de etiquetas para controlar el acceso

AWS DMS define un conjunto de pares clave/valor comunes que están disponibles para su uso en las políticas definidas por el cliente sin necesidad de requisitos de etiquetado adicionales. Para obtener más información acerca del etiquetado de recursos de AWS DMS, consulte [Etiquetado de recursos en AWS Database Migration Service \(p. 340\)](#).

A continuación se enumeran las etiquetas estándar disponibles para su uso con AWS DMS:

- `aws:CurrentTime`: representa la fecha y la hora de la solicitud, lo que permite la restricción del acceso en función de criterios temporales.
- `aws:EpochTime`: esta etiqueta es similar a la etiqueta `aws:CurrentTime` de arriba, salvo que la hora actual se representa como el número de segundos transcurridos desde la fecha de inicio de Unix.
- `aws:MultiFactorAuthPresent`: se trata de una etiqueta booleana que indica si la solicitud se firmó o no mediante la autenticación multifactor.
- `aws:MultiFactorAuthAge`: proporciona acceso a la antigüedad del token de autenticación multifactor (en segundos).
- `aws:principaltype`: proporciona acceso al tipo de principal (usuario, cuenta, usuario federado, etc.) para la solicitud actual.
- `aws:SourceIP`: representa la dirección IP de origen para el usuario que emite la solicitud.
- `aws:UserAgent`: ofrece información acerca de la aplicación cliente que solicita un recurso.
- `aws:userid`: proporciona acceso al ID del usuario que emite la solicitud.
- `aws:username`: proporciona acceso al nombre del usuario que emite la solicitud.
- `dms:InstanceClass`: proporciona acceso al tamaño del cómputo de los host de la instancia de replicación.
- `dms:StorageSize`: proporciona acceso al tamaño del volumen de almacenamiento (en GB).

También puede definir sus propias etiquetas. Las etiquetas definidas por el cliente son pares sencillos de clave-valor que se conservan en el servicio de etiquetado de AWS. Puede añadirlos a recursos de AWS DMS, incluidas las instancias de replicación, los puntos de enlace y las tareas. Estas etiquetas se asocian mediante declaraciones "condicionales" de IAM en las políticas y se hace referencia a ellas mediante una

etiqueta condicional específica. Las claves de etiquetas tienen el prefijo "dms", el tipo de recurso y el prefijo "tag". Se muestra a continuación el formato de la etiqueta.

```
dms:{resource type}-tag/{tag key}={tag value}
```

Por ejemplo, suponga que desea definir una política que permita únicamente que una llamada a la API tenga éxito para una instancia de replicación que contiene la etiqueta "stage=production". La siguiente declaración condicional se asocia a un recurso con la etiqueta proporcionada.

```
"Condition":  
{  
    "streq":  
    {  
        "dms:rep-tag/stage": "production"  
    }  
}
```

Añada la siguiente etiqueta a una instancia de replicación que coincide con esta condición de la política.

```
stage production
```

Además de las etiquetas ya asignadas a recursos de AWS DMS, también se pueden escribir políticas para limitar las claves de etiquetas y los valores que se pueden aplicar a un recurso determinado. En este caso, el prefijo de la etiqueta es "req".

Por ejemplo, la siguiente declaración de la política limita las etiquetas que un usuario puede asignar a un recurso determinado a una lista específica de valores permitidos.

```
"Condition":  
{  
    "streq":  
    {  
        "dms:req-tag/stage": [ "production", "development", "testing" ]  
    }  
}
```

Los siguientes ejemplos de políticas limitan el acceso a un recurso de AWS DMS en función de las etiquetas de recursos.

La siguiente política limita el acceso a una instancia de replicación donde el valor de la etiqueta es "Desktop" y la clave de la etiqueta es "Env":

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Action": [  
                "dms:*"  
            ],  
            "Effect": "Deny",  
            "Resource": "*",  
            "Condition": {  
                "StringEquals": {  
                    "dms:rep-tag/Env": [  
                        "Desktop"  
                    ]  
                }  
            }  
        }  
    ]  
}
```

```
        }
    ]  
}
```

Los siguientes comandos tienen éxito o se produce un error en función de la política de IAM que restringe el acceso cuando el valor de etiqueta es "Desktop" y la clave de etiqueta es "Env".

```
$ aws dms list-tags-for-resource
--resource-name arn:aws:dms:us-east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN
--endpoint-url http://localhost:8000
{
    "TagList": [
        {
            "Value": "Desktop",
            "Key": "Env"
        }
    ]
}

$ aws dms delete-replication-instance
--replication-instance-arn "arn:aws:dms:us-east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN"
A client error (AccessDeniedException) occurred when calling the DeleteReplicationInstance
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms:DeleteReplicationInstance on resource: arn:aws:dms:us-
east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN

$ aws dms modify-replication-instance
--replication-instance-arn "arn:aws:dms:us-east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN"

A client error (AccessDeniedException) occurred when calling the ModifyReplicationInstance
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms:ModifyReplicationInstance on resource: arn:aws:dms:us-
east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN

$ aws dms add-tags-to-resource
--resource-name arn:aws:dms:us-east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN
--tags Key=CostCenter,Value=1234

A client error (AccessDeniedException) occurred when calling the AddTagsToResource
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms:AddTagsToResource on resource: arn:aws:dms:us-
east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN

$ aws dms remove-tags-from-resource
--resource-name arn:aws:dms:us-east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN
--tag-keys Env

A client error (AccessDeniedException) occurred when calling the RemoveTagsFromResource
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms:RemoveTagsFromResource on resource: arn:aws:dms:us-
east-1:152683116:rep:46DHOU7JOJYOJXWDOZNFEN
```

La siguiente política limita el acceso a un punto de enlace de AWS DMS donde el valor de la etiqueta es "Desktop" y la clave de la etiqueta es "Env".

```
{
    "Version": "2012-10-17",
    "Statement": [
        {

```

```
    "Action": [
        "dms:*"
    ],
    "Effect": "Deny",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "dms:endpoint-tag/Env": [
                "Desktop"
            ]
        }
    }
}
```

Los siguientes comandos tienen éxito o se produce un error en función de la política de IAM que restringe el acceso cuando el valor de etiqueta es "Desktop" y la clave de etiqueta es "Env".

```
$ aws dms list-tags-for-resource
--resource-name arn:aws:dms:us-east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I
{
    "TagList": [
        {
            "Value": "Desktop",
            "Key": "Env"
        }
    ]
}

$ aws dms delete-endpoint
--endpoint-arn "arn:aws:dms:us-east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I"

A client error (AccessDeniedException) occurred when calling the DeleteEndpoint
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms>DeleteEndpoint on resource: arn:aws:dms:us-
east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I

$ aws dms modify-endpoint
--endpoint-arn "arn:aws:dms:us-east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I"

A client error (AccessDeniedException) occurred when calling the ModifyEndpoint
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms>ModifyEndpoint on resource: arn:aws:dms:us-
east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I

$ aws dms add-tags-to-resource
--resource-name arn:aws:dms:us-east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I
--tags Key=CostCenter,Value=1234

A client error (AccessDeniedException) occurred when calling the AddTagsToResource
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
dms>AddTagsToResource on resource: arn:aws:dms:us-
east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I

$ aws dms remove-tags-from-resource
--resource-name arn:aws:dms:us-east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I
--tag-keys Env

A client error (AccessDeniedException) occurred when calling the RemoveTagsFromResource
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:
```

```
dms:RemoveTagsFromResource on resource: arn:aws:dms:us-east-1:152683116:endpoint:J2YCZPNGOLFY52344IZWA6I
```

La siguiente política limita el acceso a una tarea de replicación donde el valor de la etiqueta es "Desktop" y la clave de la etiqueta es "Env".

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Action": [  
                "dms:*"  
            ],  
            "Effect": "Deny",  
            "Resource": "*",  
            "Condition": {  
                "StringEquals": {  
                    "dms:task-tag/Env": [  
                        "Desktop"  
                    ]  
                }  
            }  
        }  
    ]  
}
```

Los siguientes comandos tienen éxito o se produce un error en función de la política de IAM que restringe el acceso cuando el valor de etiqueta es "Desktop" y la clave de etiqueta es "Env".

```
$ aws dms list-tags-for-resource  
--resource-name arn:aws:dms:us-east-1:152683116:task:RB7N24J2XBUPS3RFABZTG3  
{  
    "TagList": [  
        {  
            "Value": "Desktop",  
            "Key": "Env"  
        }  
    ]  
}  
  
$ aws dms delete-replication-task  
--replication-task-arn "arn:aws:dms:us-east-1:152683116:task:RB7N24J2XBUPS3RFABZTG3"  
  
A client error (AccessDeniedException) occurred when calling the DeleteReplicationTask  
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:  
dms>DeleteReplicationTask on resource: arn:aws:dms:us-  
east-1:152683116:task:RB7N24J2XBUPS3RFABZTG3  
  
$ aws dms add-tags-to-resource  
--resource-name arn:aws:dms:us-east-1:152683116:task:RB7N24J2XBUPS3RFABZTG3  
--tags Key=CostCenter,Value=1234  
  
A client error (AccessDeniedException) occurred when calling the AddTagsToResource  
operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform:  
dms>AddTagsToResource on resource: arn:aws:dms:us-  
east-1:152683116:task:RB7N24J2XBUPS3RFABZTG3  
  
$ aws dms remove-tags-from-resource  
--resource-name arn:aws:dms:us-east-1:152683116:task:RB7N24J2XBUPS3RFABZTG3
```

```
--tag-keys Env
```

```
A client error (AccessDeniedException) occurred when calling the RemoveTagsFromResource operation: User: arn:aws:iam::152683116:user/dmstestusr is not authorized to perform: dms:RemoveTagsFromResource on resource: arn:aws:dms:us-east-1:152683116:task:RB7N24J2XBUPS3RFABZTG3
```

## Configuración de una clave de cifrado y especificación de permisos KMS

AWS DMS cifra el almacenamiento utilizado por una instancia de replicación y la información de conexión del punto de enlace. Para cifrar los datos almacenados que utiliza una instancia de replicación, AWS DMS se sirve de una clave de AWS Key Management Service (AWS KMS) que es exclusiva en su cuenta de AWS. Puede ver y administrar esta clave con AWS KMS. Puede utilizar la clave KMS predeterminada en su cuenta (`aws/dms`) o crear una clave KMS personalizada. Si ya posee una clave KMS, también la puede utilizar para el cifrado.

La clave KMS predeterminada (`aws/dms`) se crea la primera vez que lanza una instancia de replicación, si no ha seleccionado ninguna clave maestra KMS personalizada en la sección Advanced (Avanzado) de la página Create Replication Instance (Crear instancia de replicación). Si utiliza la clave KMS predeterminada, los únicos permisos que debe otorgar a la cuenta de usuario de IAM que utilice para la migración son `kms>ListAliases` y `kms:DescribeKey`. Para obtener más información sobre el uso de la clave KMS predeterminada, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).

Para utilizar una clave KMS personalizada, asigne permisos a la clave KMS personalizada utilizando una de las siguientes opciones:

- Añada la cuenta de usuario de IAM utilizada para la migración como administrador de claves o usuario de clave a la clave KMS personalizada. Esto le permitirá que se concedan los permisos de KMS necesarios a la cuenta de usuario de IAM. Esta acción se suma a los permisos de IAM que otorga a la cuenta de usuario de IAM para utilizar AWS DMS. Para obtener más información sobre la concesión de permisos a un usuario clave, consulte [Permite a los usuarios de claves utilizar la CMK](#) en la AWS Key Management Service Developer Guide.
- Si no desea añadir la cuenta de usuario de IAM como administrador de claves o usuario de claves a su clave KMS personalizada, añada los siguientes permisos adicionales a los permisos de IAM que debe conceder a la cuenta de usuario de IAM para utilizar AWS DMS.

```
{
    "Effect": "Allow",
    "Action": [
        "kms>ListAliases",
        "kms:DescribeKey",
        "kms>CreateGrant",
        "kms:Encrypt",
        "kms:ReEncrypt*"
    ],
    "Resource": "*"
},
```

AWS DMS no funciona con alias de clave de KMS. Sin embargo, puede utilizar el número de recurso de Amazon (ARN) de la clave de KMS al especificar la información de clave de KMS. Para obtener más

información sobre cómo crear sus propias claves KMS y dar a los usuarios acceso a una clave KMS, consulte la [Guía para desarrolladores de KMS](#).

Si no especifica un identificador de clave KMS, AWS DMS utilizará su clave de cifrado predeterminada. KMS crea la clave de cifrado predeterminada para AWS DMS para su cuenta de AWS. Cada cuenta de AWS dispone de una clave de cifrado predeterminada diferente para cada región de AWS.

Para gestionar las claves de KMS que se utilizan para cifrar sus recursos de AWS DMS debe utilizar KMS. Encontrará KMS en la Consola de administración de AWS eligiendo Identity & Access Management (Administración de identidad y acceso) en la página de inicio de la consola y luego Encryption Keys (Claves de cifrado) en el panel de navegación. KMS combina hardware y software seguros y altamente disponibles para ofrecer un sistema de gestión de claves escalado a la nube. Si utiliza KMS, puede crear claves de cifrado y definir las políticas que controlan cómo se pueden utilizar dichas claves. KMS es compatible con AWS CloudTrail, lo que permite auditar el uso de claves para comprobar que las claves se utilizan de forma adecuada. Las claves de KMS se pueden utilizar en combinación con AWS DMS y admitir servicios de AWS tales como Amazon RDS, Amazon S3, Amazon Redshift y Amazon EBS.

También puede crear claves de KMS personalizadas específicamente para cifrar datos de destino para los siguientes puntos de enlace de AWS DMS:

- Amazon Redshift: para obtener más información, consulte [Creación y uso de claves de AWS KMS para cifrar datos de destino de Amazon Redshift \(p. 178\)](#).
- Amazon S3: para obtener más información, consulte [Creación de claves de AWS KMS para cifrar objetos de destino de Amazon S3 \(p. 197\)](#).

Una vez que haya creado sus recursos de AWS DMS con una clave KMS, ya no podrá cambiar la clave de cifrado de dichos recursos. Asegúrese de determinar los requisitos de clave de cifrado antes de crear sus recursos de AWS DMS.

## Seguridad de red para AWS Database Migration Service

Los requisitos de seguridad de la red que crea al utilizar AWS Database Migration Service dependerán de cómo configure la red. Las reglas generales para la seguridad de la red para AWS DMS son las siguientes:

- La instancia de replicación debe tener acceso a los puntos de enlace de origen y de destino. El grupo de seguridad para la instancia de replicación debe tener las ACL de red o las reglas que permiten dar salida desde la instancia del puerto de la base de datos a los puntos de enlace de la base de datos.
- Los puntos de enlace de la base de datos deben incluir las ACL de red y las reglas del grupo de seguridad que permiten el acceso entrante de la instancia de replicación. Puede hacerlo utilizando el grupo de seguridad de la instancia de replicación, la dirección IP privada, la dirección IP pública o la dirección pública de la gateway NAT, en función de su configuración.
- Si su red utiliza un túnel de VPN, la instancia Amazon EC2, que actúa como la gateway NAT, debe utilizar un grupo de seguridad que disponga de reglas que permitan a la instancia de replicación enviar tráfico a través de dicho túnel.

De forma predeterminada, el grupo de seguridad de VPC que utiliza la instancia de replicación AWS DMS dispone de reglas que permiten salir a 0.0.0.0/0 en todos los puertos. Si modifica este grupo de seguridad o utiliza su propio grupo de seguridad, la salida debe estar permitida al menos a los puntos de enlace de origen y de destino en los puertos de la base de datos respectivos.

Las configuraciones de red que puede utilizar para la migración de bases de datos requieren consideraciones de seguridad específicas:

- [Configuración con todos los componentes de migración de bases de datos en una VPC \(p. 72\)](#): el grupo de seguridad utilizado por los puntos de enlace debe permitir la entrada al puerto de la base de datos desde la instancia de replicación. Asegúrese de que el grupo de seguridad utilizado por la instancia de replicación entra a los puntos de enlace. Otra opción es crear una regla en el grupo de seguridad que utilizan los puntos de enlace que otorgue acceso a la dirección IP privada de la instancia de replicación.
- [Configuración con dos VPC \(p. 72\)](#): el grupo de seguridad utilizado por la instancia de replicación debe tener una regla para el rango de VPC y el puerto de la base de datos.
- [Configuración de una red a una VPC mediante AWS Direct Connect o una VPN \(p. 73\)](#): es un túnel de VPN que permite el tráfico a través del túnel desde la VPC a una VPN local. En esta configuración, la VPC incluye una regla de direccionamiento que envía el tráfico destinado a una dirección IP o a un rango específico a un host que puede conectar el tráfico de la VPC con la VPN local. En este caso, el host de NAT incluye su propia configuración del grupo de seguridad que debe permitir el tráfico desde la dirección IP privada o el grupo de seguridad de la instancia de replicación a la instancia de NAT.
- [Configuración de una red a una VPC mediante Internet \(p. 73\)](#): el grupo de seguridad de VPC debe incluir reglas de direccionamiento que envíen el tráfico no destinado a la VPC a la gateway de Internet. En esta configuración, la conexión con el punto de enlace parece provenir de la dirección IP pública de la instancia de replicación.
- [Configuración con una instancia de base de datos de Amazon RDS que no está en una VPC a una instancia de base de datos en una VPC con ClassicLink \(p. 74\)](#): cuando la instancia de base de datos de Amazon RDS de origen o destino no está en una VPC y no comparte un grupo de seguridad con la VPC en la que se encuentra la instancia de replicación, puede configurar un servidor proxy y utilizar ClassicLink para conectar las bases de datos de origen y destino.
- El punto de enlace de origen está fuera de la VPC que utiliza la instancia de replicación y emplea una gateway NAT: puede configurar una gateway de conversión de las direcciones de red (NAT) empleando una única dirección IP elástica asociada a una única interfaz de red elástica. Esta interfaz de red elástica después recibe un identificador NAT (nat- #####). Si la VPC incluye una ruta predeterminada a dicho NAT en lugar de la gateway de Internet, la instancia de replicación aparece para ponerse en contacto con el punto de enlace de la base de datos mediante la dirección IP pública de la gateway de Internet. En este caso, la entrada al punto de enlace de la base de datos fuera de la VPC debe permitir la entrada de la dirección NAT en lugar de la dirección IP pública de la instancia de replicación.

## Uso de SSL con AWS Database Migration Service

Puede cifrar las conexiones para los puntos de enlace de origen y de destino utilizando la capa de conexión segura (SSL). Para hacerlo, puede utilizar la consola de administración de AWS DMS o la API de AWS DMS para asignar un certificado a un punto de enlace. También puede utilizar la consola de AWS DMS para administrar los certificados.

No todas las bases de datos utilizan SSL de la misma forma. Compatibilidad de Amazon Aurora con MySQL utiliza el nombre de servidor, el punto de enlace de la instancia principal del clúster, como punto de enlace de SSL. Un punto de enlace de Amazon Redshift ya utiliza una conexión SSL y no requiere una conexión SSL configurada por AWS DMS. Un punto de enlace de Oracle requiere pasos adicionales. Para obtener más información, consulte [Compatibilidad con SSL para un punto de enlace de Oracle \(p. 55\)](#).

### Temas

- [Límites al uso de SSL con AWS Database Migration Service \(p. 52\)](#)
- [Gestión de certificados \(p. 53\)](#)
- [Activación de SSL para un punto de enlace PostgreSQL, SQL Server o compatible con MySQL \(p. 53\)](#)
- [Compatibilidad con SSL para un punto de enlace de Oracle \(p. 55\)](#)

Para asignar un certificado a un punto de enlace, proporcione el certificado raíz o la cadena de certificados CA intermedios que llevan hacia la raíz (como un paquete de certificados) y que se utilizó para firmar el certificado SSL del servidor desplegado en su punto de enlace. Los certificados se aceptan solo como archivos X509 en formato PEM. Al importar un certificado, recibe un nombre de recurso de Amazon (ARN) que puede utilizar para especificar dicho certificado para un punto de enlace. Si utiliza Amazon RDS, puede descargar el paquete de CA y certificados raíz proporcionado por Amazon RDS en <https://s3.amazonaws.com/rds-downloads/rds-combined-ca-bundle.pem>.

Puede elegir entre varios modos de SSL para verificar su certificado SSL.

- none: la conexión no está cifrada. Esta opción no es segura, pero es menos costosa.
- require: la conexión se cifra mediante SSL (TLS) pero no se hace ninguna verificación de CA. Esta opción es más segura y más costosa.
- verify-ca: la conexión está cifrada. Esta opción es más segura y más costosa. Esta opción verifica el certificado de servidor.
- verify-full: la conexión está cifrada. Esta opción es más segura y más costosa. Esta opción verifica el certificado de servidor y que el nombre de host del servidor coincida con el atributo del nombre de host para el certificado.

No todos los modos SSL funcionan con todos los puntos de enlace de la base de datos. En la siguiente tabla se indica qué modos de SSL son compatibles con qué motor de base de datos.

Motor de base de datos	ninguno	require	verify-ca	verify-full
MySQL/MariaDB/ Amazon Aurora MySQL	Default (Predeterminada)	No admitido	Soportado	Soportado
Microsoft SQL Server	Valor predeterminado	Soportado	No es compatible	Soportado
PostgreSQL	Valor predeterminado	Soportado	Soportado	Soportado
Amazon Redshift	Valor predeterminado	SSL no activado	SSL no activado	SSL no activado
Oracle	Valor predeterminado	No admitido	Soportado	No es compatible
SAP ASE	Valor predeterminado	SSL no activado	SSL no activado	Soportado
MongoDB	Valor predeterminado	Soportado	No es compatible	Soportado
Db2 LUW	Valor predeterminado	No es compatible	Soportado	No es compatible

## Límites al uso de SSL con AWS Database Migration Service

- No se admiten las conexiones SSL con puntos de enlace Amazon Redshift de destino. AWS DMS utiliza un bucket de Amazon S3 para transferir datos a la base de datos de Redshift. Amazon Redshift cifra esta transmisión de forma predeterminada.

- Al realizar tareas CDC con puntos de enlace de Oracle compatibles con SSL se pueden producir tiempos de espera en SQL. Si tiene este problema, en que los contadores CDC no reflejan los números previstos, defina el parámetro `MinimumTransactionSize` desde la sección `ChangeProcessingTuning` de la configuración de tareas con un valor inferior, empezando por un valor de tan solo 100. Para obtener más información sobre el parámetro `MinimumTransactionSize`, consulte [Configuración de los ajustes del procesamiento de cambios \(p. 269\)](#).
- Los certificados solo se pueden importar en los formatos .PEM y .SSO (wallet de Oracle).
- Si el certificado SSL del servidor está firmado por un CA intermedio, asegúrese de que toda la cadena de certificados que lleva desde el CA intermedio hasta el CA de raíz se importe como un archivo .PEM individual.
- Si utiliza certificados autofirmados en su servidor, elija require como el modo SSL. El modo SSL requiere confía de forma implícita en el certificado SSL del servidor y no intentará comprobar que el certificado lo ha firmado un CA.

## Gestión de certificados

Puede utilizar la consola de DMS para ver y gestionar sus certificados SSL. También puede importar sus certificados utilizando la consola de DMS.

Identifier	Signing algorithm	Owner	Key size	Valid from
my-cert-1	SHA1withRSA	[redacted]	2048	Thu Feb 05 01:11:
my-cert-2	SHA1withRSA	[redacted]	2048	Thu Feb 05 01:11:

## Activación de SSL para un punto de enlace PostgreSQL, SQL Server o compatible con MySQL

Puede añadir una conexión SSL a un punto de enlace recién creado o a un punto de enlace existente.

Para crear un punto de enlace de AWS DMS con SSL

1. Inicie sesión en la consola de administración de AWS y seleccione AWS Database Migration Service.

### Note

Si está registrado como usuario de AWS Identity and Access Management (IAM), debe disponer de los permisos adecuados para acceder a AWS DMS. Para obtener más información sobre los permisos necesarios para migrar bases de datos, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).

2. En el panel de navegación, elija Certificates.
3. Elija Import Certificate.
4. Cargue el certificado que desea usar para cifrar la conexión a un punto de enlace.

**Note**

También puede cargar un certificado con la consola de AWS DMS al crear o modificar un punto de enlace seleccionando Add new CA certificate (Agregar nuevo certificado de CA) en la página Create database endpoint (Crear punto de enlace de base de datos).

5. Cree un punto de enlace tal y como se describe en [Paso 3: Especificar los puntos de enlace de origen y destino \(p. 25\)](#)

Para modificar un punto de enlace de AWS DMS existente para utilizar SSL:

1. Inicie sesión en la consola de administración de AWS y seleccione AWS Database Migration Service.

**Note**

Si está registrado como usuario de AWS Identity and Access Management (IAM), debe disponer de los permisos adecuados para acceder a AWS DMS. Para obtener más información sobre los permisos necesarios para migrar bases de datos, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).

2. En el panel de navegación, elija Certificates.
3. Elija Import Certificate.
4. Cargue el certificado que desea usar para cifrar la conexión a un punto de enlace.

**Note**

También puede cargar un certificado con la consola de AWS DMS al crear o modificar un punto de enlace seleccionando Add new CA certificate (Agregar nuevo certificado de CA) en la página Create database endpoint (Crear punto de enlace de base de datos).

5. En el panel de navegación, elija Endpoints, seleccione el punto de enlace que desea modificar y elija Modify.
6. Elija un modo SSL.

Si selecciona el modo verify-ca o el modo verify-full, debe especificar el certificado de CA que desea utilizar, tal y como se muestra a continuación.

## Create database endpoint

A database endpoint is used by the replication server to connect to a database. The database specified in the endpoint can be on-prem. Details should be specified in the form below. It is recommended that you test your endpoint connections here to avoid errors during pri

Endpoint type\*  Source  Target ?

Endpoint identifier\*  ?

Source engine\*  ?

Server name\*

Port\*

SSL mode\*  ?

CA certificate\*  ?  
[Add new CA certificate](#)

User name\*

Password\*

» Advanced

7. Elija Modify.
8. Una vez modificado el punto de enlace, selecciónelo y elija Test connection para determinar si la conexión SSL funciona.

Después de crear los puntos de enlace de origen y de destino, cree una tarea que utilice estos puntos de enlace. Para obtener más información sobre la creación de una tarea, consulte [Paso 4: Crear una tarea \(p. 30\)](#).

## Compatibilidad con SSL para un punto de enlace de Oracle

Los puntos de enlace de Oracle en AWS DMS admiten los modos SSL none y verify-ca. Para utilizar SSL con un punto de enlace de Oracle, debe cargar el wallet de Oracle para el punto de enlace en lugar de archivos de certificado .pem.

### Temas

- [Uso de un certificado existente para Oracle SSL \(p. 56\)](#)
- [Uso de un certificado SSL autofirmado para Oracle SSL \(p. 56\)](#)

## Uso de un certificado existente para Oracle SSL

Para utilizar una instalación de cliente Oracle existente para crear el archivo wallet de Oracle desde el archivo de certificado CA, siga los pasos que se indican a continuación.

Para utilizar una instalación de cliente Oracle para Oracle SSL con AWS DMS

1. Ejecute el siguiente comando para especificar la ubicación del directorio dbhome\_1 en la variable del sistema ORACLE\_HOME:

```
prompt>export ORACLE_HOME=/home/user/app/user/product/12.1.0/dbhome_1
```

2. Añada \$ORACLE\_HOME/lib a la variable del sistema LD\_LIBRARY\_PATH.

```
prompt>export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib
```

3. Cree un directorio para el wallet de Oracle en \$ORACLE\_HOME/ssl\_wallet.

```
prompt>mkdir $ORACLE_HOME/ssl_wallet
```

4. Coloque el archivo .pem del certificado CA en el directorio ssl\_wallet. Los clientes de Amazon RDS pueden descargar el archivo de certificados CA de RDS desde <https://s3.amazonaws.com/rds-downloads/rds-ca-2015-root.pem>.
5. Ejecute los siguientes comandos para crear el wallet de Oracle.

```
prompt>orapki wallet create -wallet $ORACLE_HOME/ssl_wallet -auto_login_only
prompt>orapki wallet add -wallet $ORACLE_HOME/ssl_wallet -trusted_cert -cert
$ORACLE_HOME/ssl_wallet/ca-cert.pem -auto_login_only
```

Cuando haya completado los pasos anteriores, podrá importar el archivo wallet con la API ImportCertificate especificando el parámetro certificate-wallet. A continuación, podrá utilizar el certificado wallet importado al seleccionar verify-ca como el modo SSL al crear o modificar su punto de enlace de Oracle.

### Note

Los wallets de Oracle son archivos binarios. AWS DMS acepta estos archivos tal y como son.

## Uso de un certificado SSL autofirmado para Oracle SSL

Para utilizar un certificado autofirmado para Oracle SSL, haga lo siguiente.

Para utilizar un certificado autofirmado para Oracle SSL con AWS DMS

1. Cree un directorio que utilizará para trabajar con el certificado autofirmado.

```
mkdir <SELF_SIGNED_CERT_DIRECTORY>
```

2. Cambie al directorio que ha creado en el paso anterior.

```
cd <SELF_SIGNED_CERT_DIRECTORY>
```

- Cree una clave raíz.

```
openssl genrsa -out self-rootCA.key 2048
```

- Firme usted mismo un certificado raíz con la clave raíz que ha creado en el paso anterior.

```
openssl req -x509 -new -nodes -key self-rootCA.key  
-sha256 -days 1024 -out self-rootCA.pem
```

- Cree un directorio wallet de Oracle para la base de datos de Oracle.

```
mkdir $ORACLE_HOME/self_signed_ssl_wallet
```

- Cree un nuevo wallet de Oracle.

```
orapki wallet create -wallet $ORACLE_HOME/self_signed_ssl_wallet  
-pwd <password> -auto_login_local
```

- Añada el certificado raíz al wallet de Oracle.

```
orapki wallet add -wallet $ORACLE_HOME/self_signed_ssl_wallet  
-trusted_cert -cert self-rootCA.pem -pwd <password>
```

- Enumere el contenido del wallet de Oracle. La lista debe incluir el certificado raíz.

```
orapki wallet display -wallet $ORACLE_HOME/self_signed_ssl_wallet
```

- Genere la solicitud de firma del certificado (CSR) mediante la utilidad ORAPKI.

```
orapki wallet add -wallet $ORACLE_HOME/self_signed_ssl_wallet  
-dn "CN=dms" -keysize 2048 -sign_alg sha256 -pwd <password>
```

- Ejecute el comando siguiente.

```
openssl pkcs12 -in ewallet.p12 -nodes -out nonoracle_wallet.pem
```

- Inserte 'dms' como nombre común.

```
openssl req -new -key nonoracle_wallet.pem -out certrequest.csr
```

- Obtenga la firma del certificado.

```
openssl req -noout -text -in self-signed-oracle.csr | grep -i signature
```

- Si la salida del paso 12 es sha256WithRSAEncryption, entonces ejecute el código siguiente.

```
openssl x509 -req -in self-signed-oracle.csr -CA self-rootCA.pem  
-CAkey self-rootCA.key -CAcreateserial  
-out self-signed-oracle.crt -days 365 -sha256
```

- Si la salida del paso 12 es md5WithRSAEncryption, entonces ejecute el código siguiente.

```
openssl x509 -req -in certrequest.csr -CA self-rootCA.pem  
-CAkey self-rootCA.key -CAcreateserial
```

```
-out certrequest.crt -days 365 -sha256
```

15. Añada el certificado al wallet.

```
orapki wallet add -wallet $ORACLE_HOME/self_signed_ssl_wallet -user_cert  
-cert certrequest.crt -pwd <password>
```

16. Configure el archivo sqlnet.ora (\$ORACLE\_HOME/network/admin/sqlnet.ora).

```
WALLET_LOCATION =  
  (SOURCE =  
    (METHOD = FILE)  
    (METHOD_DATA =  
      (DIRECTORY = <ORACLE_HOME>/self_signed_ssl_wallet)  
    )  
  )  
  
SQLNET.AUTHENTICATION_SERVICES = (NONE)  
SSL_VERSION = 1.0  
SSL_CLIENT_AUTHENTICATION = FALSE  
SSL_CIPHER_SUITES = (SSL_RSA_WITH_AES_256_CBC_SHA)
```

17. Detenga el listener de Oracle.

```
lsnrctl stop
```

18. Añada entradas para SSL en el archivo listener.ora (\$ORACLE\_HOME/network/admin/listener.ora).

```
SSL_CLIENT_AUTHENTICATION = FALSE  
WALLET_LOCATION =  
  (SOURCE =  
    (METHOD = FILE)  
    (METHOD_DATA =  
      (DIRECTORY = <ORACLE_HOME>/self_signed_ssl_wallet)  
    )  
  )  
  
SID_LIST_LISTENER =  
  (SID_LIST =  
    (SID_DESC =  
      (GLOBAL_DBNAME = <SID>)  
      (ORACLE_HOME = <ORACLE_HOME>)  
      (SID_NAME = <SID>)  
    )  
  )  
  
LISTENER =  
  (DESCRIPTION_LIST =  
    (DESCRIPTION =  
      (ADDRESS = (PROTOCOL = TCP)(HOST = localhost.localdomain)(PORT = 1521))  
      (ADDRESS = (PROTOCOL = TCPS)(HOST = localhost.localdomain)(PORT = 1522))  
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))  
    )  
  )
```

19. Configure el archivo tnsnames.ora (\$ORACLE\_HOME/network/admin/tnsnames.ora).

```
<SID>=  
(DESCRIPTION=  
  (ADDRESS_LIST =  
    (ADDRESS=(PROTOCOL = TCP)(HOST = localhost.localdomain)(PORT = 1521))  
  )
```

```
(CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = <SID>)
)
)

<SID>_ssl=
(DESCRIPTION=
    (ADDRESS_LIST =
        (ADDRESS=(PROTOCOL = TCPS)(HOST = localhost.localdomain)(PORT = 1522))
    )
    (CONNECT_DATA =
        (SERVER = DEDICATED)
        (SERVICE_NAME = <SID>)
    )
)
```

20. Reinicie el listener de Oracle.

```
lsnrctl start
```

21. Muestre el estado de listener de Oracle.

```
lsnrctl status
```

22. Pruebe la conexión SSL a la base de datos desde localhost utilizando sqlplus y la entrada tnsnames SSL.

```
sqlplus -L <ORACLE_USER>@<SID>_ssl
```

23. Compruebe que se ha conectado correctamente mediante SSL.

```
SELECT SYS_CONTEXT('USERENV', 'network_protocol') FROM DUAL;
SYS_CONTEXT('USERENV', 'NETWORK_PROTOCOL')
-----
tcps
```

24. Cambie de directorio al directorio con el certificado autofirmado.

```
cd <SELF_SIGNED_CERT_DIRECTORY>
```

25. Cree un nuevo wallet de cliente de Oracle que utilice AWS DMS.

```
orapki wallet create -wallet ./ -auto_login_only
```

26. Añada el certificado raíz autofirmado al wallet de Oracle.

```
orapki wallet add -wallet ./ -trusted_cert -cert rootCA.pem -auto_login_only
```

27. Enumere el contenido del wallet de Oracle que utilice AWS DMS. La lista debe incluir el certificado raíz autofirmado.

```
orapki wallet display -wallet ./
```

28. Cargue el wallet de Oracle que acaba de crear en AWS DMS.

## Cambio de la contraseña de la base de datos

En la mayoría de los casos, cambiar la contraseña de la base de datos del punto de enlace de origen o de destino es un paso sencillo. Pero si necesita cambiar la contraseña de la base de datos de un punto de enlace que utiliza actualmente en una tarea de replicación o de migración, el proceso es algo más complejo. El procedimiento siguiente muestra cómo hacerlo.

Para cambiar la contraseña de la base de datos de un punto de enlace en una tarea de replicación o de migración

1. Inicie sesión en la consola de administración de AWS y seleccione AWS DMS. Tenga en cuenta que si está registrado como usuario de AWS Identity and Access Management (IAM), debe disponer de los permisos adecuados para acceder a AWS DMS. Para obtener más información sobre los permisos que se necesitan, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).
2. En el panel de navegación, elija Tasks.
3. Elija la tarea que utiliza el punto de enlace cuya contraseña de la base de datos desea cambiar y, a continuación, elija Stop.
4. Mientras la tarea está parada, puede cambiar la contraseña de la base de datos del punto de enlace utilizando las herramientas nativas que utiliza para trabajar con la base de datos.
5. Vuelva a la consola de administración de DMS y elija Endpoints en el panel de navegación.
6. Elija el punto de enlace de la base de datos del que ha cambiado la contraseña y, luego, elija Modify.
7. Escriba la nueva contraseña en la casilla Password y, a continuación, elija Modify.
8. Elija Tasks en el panel de navegación.
9. Elija la tarea que ha detenido anteriormente y elija Start/Resume.
10. Elija Start o Resume, en función de cómo desee continuar la tarea y, a continuación, elija Start task.

# Límites de AWS Database Migration Service

A continuación, puede encontrar los límites de recursos y las restricciones de nomenclatura para AWS Database Migration Service (AWS DMS).

El tamaño máximo de una base de datos que AWS DMS puede migrar depende del entorno de origen, la distribución de los datos en la base de datos de origen y lo ocupado que esté su sistema de origen. La mejor forma de saber si su sistema puede funcionar bien con AWS DMS es probarlo. Empiece despacio para comprobar que la configuración funciona, agregue algunos objetos complejos y, finalmente, intente realizar una carga completa de prueba.

## Límites de AWS Database Migration Service

Cada cuenta de AWS tiene límites del número de recursos de AWS DMS que se pueden crear por región. Una vez que se alcance el límite de un recurso, las llamadas adicionales para crear ese recurso dejarán de funcionar con una excepción.

El límite de 6 TB de almacenamiento se aplica a la instancia de replicación de DMS. Este almacenamiento se utiliza para guardar los cambios en la memoria caché si el destino no puede mantener el ritmo del origen y para almacenar información de registro. Este límite no se aplica al tamaño de destino. Los puntos de enlace de destino pueden superar los 6 TB.

En la siguiente tabla se muestran los recursos de AWS DMS y sus límites por región.

Recurso	Límite predeterminado
Instancias de replicación	20
Cantidad total de almacenamiento	6 TB
Suscripciones de eventos	20
Grupos de subredes de replicación	20
Subredes por grupo de subredes de replicación	20
Puntos de enlace	100
Tareas	200
Puntos de enlace por instancia	20

# Trabajar con una instancia de replicación de AWS DMS

Cuando se crea una instancia de replicación de AWS DMS, AWS DMS crea la instancia de replicación en una instancia Amazon Elastic Compute Cloud (Amazon EC2) en una VPC basada en el servicio Amazon Virtual Private Cloud (Amazon VPC). Esta instancia de replicación es la que se usa para migrar sus bases de datos. La instancia de replicación proporciona alta disponibilidad y soporte de comutación por error mediante una implementación Multi-AZ cuando selecciona la opción Multi-AZ.

En una implementación Multi-AZ, AWS DMS aprovisiona y mantiene automáticamente una réplica en espera síncrona de la instancia de replicación en una zona de disponibilidad diferente. La instancia de replicación principal se replica sincrónicamente en las zonas de disponibilidad en la réplica en espera. Este enfoque proporciona redundancia de datos, elimina los bloqueos de E/S y minimiza los picos de latencia.



AWS DMS utiliza una instancia de replicación para conectarse con su almacén de datos de origen, leer los datos de origen y formatear los datos para que el almacén de datos de destino pueda consumirlos. Una instancia de replicación también carga los datos en el almacén de datos de destino. La mayor parte de este procesamiento ocurre en la memoria. No obstante, es posible que en las transacciones de mayor volumen se precise almacenar en la memoria búfer del disco. Las transacciones almacenadas en caché y los archivos de registro también se escriben en el disco.

Puede crear una instancia de replicación de AWS DMS en las siguientes regiones de AWS.

Región	Nombre
Región Asia Pacífico (Tokio)	ap-northeast-1
Región Asia Pacífico (Seúl)	ap-northeast-2
Región Asia Pacífico (Mumbai)	ap-south-1
Región Asia Pacífico (Singapur)	ap-southeast-1
Región Asia Pacífico (Sídney)	ap-southeast-2
Región Canadá (Central)	ca-central-1
Región UE (Fráncfort)	eu-central-1
Región UE (Irlanda)	eu-west-1

Región	Nombre
Región UE (Londres)	eu-west-2
Región América del Sur (São Paulo)	sa-east-1
Región EE.UU. Este (Norte de Virginia)	us-east-1
Región EE.UU Este (Ohio)	us-east-2
EE.UU. Oeste (Norte de California)	us-west-1
Región EE.UU. Oeste (Oregón)	us-west-2

AWS DMS admite una región de AWS especial llamada AWS GovCloud (EE.UU.), que se ha diseñado para permitir a los clientes y los organismos gubernamentales de Estados Unidos mover cargas de trabajo más confidenciales a la nube. AWS GovCloud (EE.UU.) aborda algunos de los requisitos normativos y de conformidad específicos del gobierno de Estados Unidos. Para obtener más información acerca de AWS GovCloud (EE.UU.), consulte [¿Qué es AWS GovCloud \(EE.UU.\)?](#)

A continuación, encontrará más detalles acerca de las instancias de replicación.

#### Temas

- [Selección de la instancia de replicación de AWS DMS adecuada para su migración \(p. 63\)](#)
- [Instancias de replicación pública y privada \(p. 65\)](#)
- [Mantenimiento de AWS DMS \(p. 65\)](#)
- [Trabajo con las versiones de motor de replicación \(p. 68\)](#)
- [Configuración de una red para una instancia de replicación \(p. 71\)](#)
- [Establecimiento de una clave de cifrado para una instancia de replicación \(p. 78\)](#)
- [Creación una instancia de replicación \(p. 79\)](#)
- [Modificación de una instancia de replicación \(p. 83\)](#)
- [Reinicio de una instancia de replicación \(p. 86\)](#)
- [Eliminación de una instancia de replicación \(p. 88\)](#)
- [Instrucciones DDL compatibles con AWS DMS \(p. 89\)](#)

## Selección de la instancia de replicación de AWS DMS adecuada para su migración

AWS DMS crea la instancia de replicación en una instancia de Amazon Elastic Compute Cloud (Amazon EC2). AWS DMS admite actualmente las clases de instancia de Amazon EC2 T2, C4 y R4 para instancias de replicación:

- Las clases de instancias T2 son instancias estándar de bajo costo diseñadas para proporcionar un nivel básico de desempeño de la CPU con la capacidad de ampliar ese nivel básico. Estas instancias son adecuadas para desarrollar, configurar y probar el proceso de migración de la base de datos. También funcionan bien para realizar tareas periódicas de migración de datos que pueden beneficiarse de la capacidad de ráfaga de la CPU.
- Las clases de instancia C4 se han diseñado para proporcionar el mayor nivel de desempeño del procesador para cargas de trabajo que hacen uso intensivo del equipo. Logran un desempeño muy superior en cuanto a paquetes por segundo (PPS), menor inestabilidad de la red y menor latencia

de la red. AWS DMS puede hacer un uso intensivo de la CPU, en especial al realizar migraciones heterogéneas y replicaciones como, por ejemplo, la migración desde Oracle a PostgreSQL. Las instancias C4 pueden ser una buena opción para estas situaciones.

- Las clases de instancia R4 tienen optimizada la memoria para cargas de trabajo que hacen un uso intensivo de la memoria. Las migraciones continuas o las replicaciones de sistemas de transacción de alto desempeño que utilizan DMS pueden, a veces, consumir gran cantidad de CPU y de memoria. Las instancias R4 incluyen más memoria por vCPU.

Cada instancia de replicación tiene una configuración específica de memoria y de vCPU. La siguiente tabla muestra la configuración de cada tipo de instancia de replicación. Para obtener información sobre precios, vaya a la [página de precios de AWS Database Migration Service](#).

Tipo de instancia de replicación	vCPU	Memoria (GB)
Uso general		
dms.t2.micro	1	1
dms.t2.small	1	2
dms.t2.medium	2	4
dms.t2.large	2	8
Computación optimizada		
dms.c4.large	2	3.75
dms.c4.xlarge	4	7.5
dms.c4.2xlarge	8	15
dms.c4.4xlarge	16	30
Optimizada para memoria		
dms.r4.large	2	15.25
dms.r4.xlarge	4	30.5
dms.r4.2xlarge	8	61
dms.r4.4xlarge	16	122
dms.r4.8xlarge	32	244

Para ayudarle a determinar qué clase de instancia de replicación funcionaría mejor para la migración, veamos el proceso de captura de datos de cambio (CDC) que utiliza la instancia de replicación de AWS DMS.

Supongamos que está ejecutando una tarea de carga completa más CDC (carga masiva más replicación continua). En este caso, la tarea tiene su propio repositorio SQLite para almacenar los metadatos y otra información. Antes de que AWS DMS inicie una carga completa, estos pasos tienen lugar:

- AWS DMS comienza a capturar los cambios para las tablas que está migrando desde el log de transacciones del motor de origen (los denominamos los cambios en la memoria caché). Después de que se haya realizado la carga completa, estos cambios en caché se recopilan y se aplican en el destino. En función del volumen de los cambios en la memoria caché, estos cambios se pueden aplicar directamente desde la memoria, donde se recopilan en primer lugar, hasta un umbral definido. De forma

alternativa, también pueden aplicarse desde el disco, donde los cambios se escriben cuando no se pueden mantener en memoria.

- Despues de que se apliquen cambios en la memoria caché, de forma predeterminada AWS DMS inicia una aplicación transaccional en la instancia de destino.

Durante la fase cambios en la memoria caché aplicados y la fase de replicaciones en curso, AWS DMS utiliza dos búferes de transmisión, uno para datos entrantes y otro para datos salientes. AWS DMS también utiliza un componente importante que se conoce como clasificador, que es otro búfer de memoria. A continuación se muestran dos usos importantes del componente clasificador (que tiene otros):

- Realiza un seguimiento de todas las transacciones y se asegura de que reenvía únicamente las transacciones pertinentes al búfer de salida.
- Se asegura de que las transacciones se reenvían en el mismo orden de confirmación que en el origen.

Como puede ver, tenemos tres importantes búferes de memorias en esta arquitectura para CDC en AWS DMS. Si cualquiera de estos búferes experimenta presión de memoria, la migración puede tener problemas de desempeño que podrían llegar a producir errores.

Cuando conecte cargas de trabajo pesadas con un elevado número de transacciones por segundo (TPS) en esta arquitectura, puede encontrar la memoria adicional proporcionada por instancias R4 útiles. Puede utilizar instancias R4 para almacenar un gran número de transacciones en memoria y evitar problemas de presión de memoria durante las replicaciones en curso.

## Instancias de replicación pública y privada

Puede especificar si una instancia de replicación tiene una dirección IP pública o privada que utiliza para conectarse a las bases de datos de origen y de destino.

Las instancias de replicación privadas tienen una dirección IP privada a la que no puede obtener acceso desde fuera de la red de replicación. Las instancias de replicación deben tener una dirección IP privada cuando tanto la base de datos de origen como la de destino están en la misma red que está conectada a la VPC de la instancia de replicación mediante VPN, AWS Direct Connect o interconexión de VPC.

La interconexión de VPC es una conexión en red entre dos VPC que permite direccionar a través de las direcciones IP privadas de cada VPC, como si estuviesen en la misma red. Para obtener más información acerca de las interconexiones de VPC, consulte la sección sobre [interconexión de VPC](#) en la Guía del usuario de Amazon VPC.

## Mantenimiento de AWS DMS

AWS DMS realiza tareas de mantenimiento periódicas en los recursos de AWS DMS. Estas tareas de mantenimiento suelen conllevar actualizaciones de la instancia de replicación o del sistema operativo de la instancia de replicación. Puede administrar el intervalo de tiempo para el periodo de mantenimiento y ver las actualizaciones de mantenimiento utilizando la API de AWS CLI o la API de AWS DMS. La consola de AWS DMS no permite actualmente hacer estas tareas.

Los elementos de mantenimiento necesitan que AWS DMS desconecte su instancia de replicación durante un breve plazo de tiempo. Entre las tareas de mantenimiento que precisan que un recurso esté desconectado está la de aplicar parches obligatorios al sistema operativo o a la instancia. Los parches obligatorios que tienen que ver con la seguridad y la fiabilidad de la instancia son los únicos que se programan automáticamente. Estos parches se aplican con poca frecuencia (normalmente, una o dos veces al año) y raramente llevarán más de una pequeña parte del periodo de mantenimiento. Puede hacer

que se le realicen actualizaciones de versión menores automáticas al elegir la opción de la consola Auto minor version upgrade (Actualización de versión menor automática).

## Ventana de mantenimiento de AWS DMS

Cada instancia de replicación de AWS DMS tiene un periodo de mantenimiento semanal durante el cual se aplican los cambios del sistema disponibles. Puede considerar un periodo de mantenimiento como una oportunidad para controlar cuándo se producirán las modificaciones y los parches de software.

Si AWS DMS determina que se requiere un mantenimiento durante una semana determinada, el mantenimiento se llevará a cabo en el periodo de mantenimiento de 30 minutos que usted eligió al crear la instancia de replicación. AWS DMS completa la mayor parte de mantenimiento durante el periodo de mantenimiento de 30 minutos. Sin embargo, puede que se necesite más tiempo para los cambios más grandes.

El periodo de mantenimiento de 30 minutos que seleccionó al crear la instancia de replicación es de un bloque de 8 horas de tiempo asignado para cada región de AWS. Si no especifica un periodo de mantenimiento preferido al crear su instancia de replicación, AWS DMS le asignará uno en un día de la semana seleccionado al azar. Para una instancia de replicación que utiliza una implementación Multi-AZ, podría ser necesaria una conmutación por error para completar el mantenimiento.

La siguiente tabla muestra el periodo de mantenimiento para cada región de AWS compatible con AWS DMS.

Región	Bloque de tiempo
Región Asia Pacífico (Sídney)	12:00–20:00 UTC
Región Asia Pacífico (Tokio)	13:00–21:00 UTC
Región Asia Pacífico (Mumbai)	17:30–01:30 UTC
Región Asia Pacífico (Seúl)	13:00–21:00 UTC
Región Asia Pacífico (Singapur)	14:00–22:00 UTC
Región Canadá (Central)	06:29–14:29 UTC
Región UE (Fráncfort)	23:00–07:00 UTC
Región UE (Irlanda)	22:00–06:00 UTC
Región UE (Londres)	06:00–14:00 UTC
Región América del Sur (São Paulo)	00:00–08:00 UTC
Región EE.UU. Este (Norte de Virginia)	03:00–11:00 UTC
Región EE.UU. Este (Ohio)	03:00–11:00 UTC
EE.UU. Oeste (Norte de California)	06:00–14:00 UTC
Región EE.UU. Oeste (Oregón)	06:00–14:00 UTC

Región	Bloque de tiempo
AWS GovCloud (EE.UU. Oeste)	06:00–14:00 UTC

## Efecto del mantenimiento en las tareas de migración existentes

Cuando una tarea de migración de AWS DMS se ejecuta en una instancia, los siguientes eventos se producen cuando se aplica un parche:

- Si las tablas de la tarea de migración se encuentran en la fase de replicación de cambios en curso (CDC), AWS DMS pone en pausa la tarea por un momento mientras se aplica el parche. Despues la migración continúa a partir del punto en que se interrumpió cuando se aplicó el parche.
- Si AWS DMS estaba migrando una tabla cuando se aplicó el parche, AWS DMS reiniciará la migración de la tabla.

## Cambio de la configuración del periodo de mantenimiento

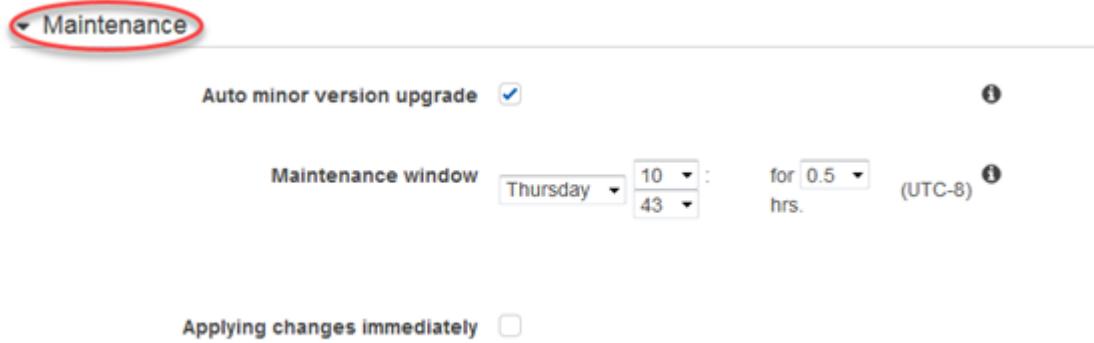
Puede cambiar el marco temporal del periodo de mantenimiento mediante la Consola de administración de AWS, la AWS CLI o la API de AWS DMS.

### Cambio de la configuración del periodo de mantenimiento mediante la consola de AWS

Puede cambiar el marco temporal del periodo de mantenimiento mediante la Consola de administración de AWS.

Para cambiar el periodo de mantenimiento preferido mediante la consola de AWS

1. Inicie sesión en la Consola de administración de AWS y elija AWS DMS.
2. En el panel de navegación, elija Replication instances.
3. Seleccione la instancia de replicación que desea modificar y elija Modify.
4. Amplíe la sección Maintenance y elija una fecha y hora para el periodo de mantenimiento.



5. Seleccione Apply changes immediately.
6. Elija Modify.

### Cambio de la configuración del periodo de mantenimiento mediante la CLI

Para ajustar la ventana de mantenimiento preferida, use el comando `modify-replication-instance` de la AWS CLI con los siguientes parámetros.

- `--replication-instance-identifier`
- `--preferred-maintenance-window`

#### Example

En el siguiente ejemplo de la AWS CLI, el periodo de mantenimiento se define para los martes de 4:00 a 4:30 a. m. UTC.

```
aws dms modify-replication-instance \
--replication-instance-identifier myrepinstance \
--preferred-maintenance-window Tue:04:00-Tue:04:30
```

#### Cambio de la configuración del periodo de mantenimiento mediante la API

Para ajustar la ventana de mantenimiento preferida, use la acción `ModifyReplicationInstance` de la API de AWS DMS con los siguientes parámetros.

- `ReplicationInstanceIdentifier = myrepinstance`
- `PreferredMaintenanceWindow = Tue:04:00-Tue:04:30`

#### Example

En el siguiente ejemplo de código, el periodo de mantenimiento se define para los martes de 4:00 a 4:30 a. m. UTC.

```
https://dms.us-west-2.amazonaws.com/
?Action=ModifyReplicationInstance
&DBInstanceIdentifier=myrepinstance
&PreferredMaintenanceWindow=Tue:04:00-Tue:04:30
&SignatureMethod=HmacSHA256
&SignatureVersion=4
&Version=2014-09-01
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIADQKE4SARGYLE/20140425/us-east-1/dms/aws4_request
&X-Amz-Date=20140425T192732Z
&X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
&X-Amz-Signature=1dc9dd716f4855e9bdf188c70f1cf9f6251b070b68b81103b59ec70c3e7854b3
```

## Trabajo con las versiones de motor de replicación

El motor de replicación es el software de AWS DMS básico que se ejecuta en la instancia de replicación y se encarga de realizar las tareas de migración que especifique. AWS lanza periódicamente nuevas versiones del software del motor de replicación de AWS DMS, con nuevas características y mejoras de desempeño. Cada versión del software del motor de replicación tiene su propio número de versión para diferenciarlo de otras versiones.

Cuando lanza una nueva instancia de replicación, esta ejecuta la última versión del motor de AWS DMS, a menos que especifique lo contrario. Para obtener más información, consulte [Trabajar con una instancia de replicación de AWS DMS \(p. 62\)](#).

Si tiene una instancia de replicación que se está ejecutando en ese momento, puede actualizarla a una versión del motor más reciente. (AWS DMS no es compatible con versiones del motor de nivel inferior). Para obtener más información, incluida una lista de versiones del motor de replicación, consulte la sección siguiente.

## Descarte de una versión de la instancia de replicación

En algunas ocasiones, AWS DMS descarta las versiones más antiguas de la instancia de replicación. A partir del 2 de abril de 2018, AWS DMS inhabilitará la creación de cualquier instancia de replicación de la versión 1.9.0. Esta versión fue soportado inicialmente AWS DMS en 15 de marzo de 2016, y se ha sustituido por las versiones posteriores que contiene mejoras a la funcionalidad, seguridad y fiabilidad.

A partir del 5 de agosto de 2018 a las 00:00 UTC, todas las instancias de replicación de DMS que ejecuten la versión 1.9.0 se programarán para una actualización automática a la última versión disponible durante el período de mantenimiento especificado para cada instancia. Le recomendamos que actualice sus instancias antes de ese momento, cuando le resulte más conveniente.

Puede iniciar una actualización de una instancia de replicación mediante las instrucciones de la sección siguiente, [Actualización de la versión del motor de una instancia de replicación](#) (p. 69).

Para las tareas de migración que se estén ejecutando cuando elija actualizar la instancia de replicación, las tablas que estén en la fase de carga completa en el momento de la actualización se vuelven a cargar desde el principio una vez que se haya completado la actualización. La replicación de todas las demás tablas debería reanudarse sin interrupción una vez que finalice la actualización. Le recomendamos que pruebe todas las tareas de migración actuales en la última versión disponible de la instancia de replicación de AWS DMS antes de actualizar las instancias desde la versión 1.9.0.

## Actualización de la versión del motor de una instancia de replicación

AWS lanza periódicamente nuevas versiones del software del motor de replicación de AWS DMS, con nuevas características y mejoras de desempeño. A continuación, se ofrece un resumen de las versiones del motor de AWS DMS disponibles.

Versión	Resumen de la versión
3.1.x	<ul style="list-style-type: none"><li>• <a href="#">Notas de la versión 3.1.3 de AWS Database Migration Service (AWS DMS)</a> (p. 395)</li><li>• <a href="#">Notas de la versión 3.1.2 de AWS Database Migration Service (AWS DMS)</a> (p. 396)</li><li>• <a href="#">Notas de la versión 3.1.1 de AWS Database Migration Service (AWS DMS)</a> (p. 397)</li></ul>
2.4.x	<ul style="list-style-type: none"><li>• <a href="#">Notas de la versión 2.4.5 de AWS Database Migration Service (AWS DMS)</a> (p. 399)</li><li>• <a href="#">Notas de la versión 2.4.4 de AWS Database Migration Service (AWS DMS)</a> (p. 400)</li><li>• <a href="#">Notas de la versión 2.4.3 de AWS Database Migration Service (AWS DMS)</a> (p. 401)</li><li>• <a href="#">Notas de la versión 2.4.2 de AWS Database Migration Service (AWS DMS)</a> (p. 402)</li><li>• <a href="#">Notas de la versión 2.4.1 de AWS Database Migration Service (AWS DMS)</a> (p. 404)</li><li>• <a href="#">Notas de la versión 2.4.0 de AWS Database Migration Service (AWS DMS)</a> (p. 405)</li></ul>
2.3.x	<ul style="list-style-type: none"><li>• <a href="#">Notas de la versión 2.3.0 de AWS Database Migration Service (AWS DMS)</a> (p. 406)</li></ul>

Versión	Resumen de la versión
2.2.x	<ul style="list-style-type: none"><li>Compatible con Microsoft SQL Server 2016 como origen de AWS DMS o destino de AWS DMS.</li><li>Compatible con SAP ASE 16, como origen de AWS DMS o destino de AWS DMS.</li><li>Compatible con la ejecución de Microsoft SQL Server en Microsoft Azure, solo como origen de AWS DMS. Puede realizar una migración completa de datos existentes; sin embargo, la captura de datos de cambios (CDC) no está disponible.</li></ul>
1.9.x	Versión acumulativa del software del motor de replicación de AWS DMS.

#### Note

Puede actualizar las instancias de replicación directamente desde la versión 2.4.x a la más reciente.

## Actualización de la versión del motor mediante la consola

Puede actualizar una instancia de replicación de AWS DMS utilizando la Consola de administración de AWS.

Para actualizar una instancia de replicación con la consola

1. Abra la consola de AWS DMS en <https://console.aws.amazon.com/dms/>.
2. En el panel de navegación, elija Replication instances.
3. Elija su motor de replicación y, a continuación, seleccione Modify.
4. Para Replication engine version (Versión del motor de replicación), elija el número de la versión que quiere y, a continuación, elija Modify (Modificar).

#### Note

La actualización de la instancia de replicación tarda varios minutos. Cuando la instancia esté lista, su estado cambiará a available.

## Actualización de la versión del motor mediante la CLI

Puede actualizar una instancia de replicación de AWS DMS utilizando la AWS CLI de la siguiente manera.

Para actualizar una instancia de replicación con la AWS CLI

1. Determine el Nombre de recurso de Amazon (ARN) de la instancia de replicación mediante el siguiente comando.

```
aws dms describe-replication-instances \
--query "ReplicationInstances[*].
[ReplicationInstanceIdentifier,ReplicationInstanceArn,ReplicationInstanceClass]"
```

En la salida, tome nota del Nombre de recurso de Amazon (ARN) de la instancia de replicación que quiere actualizar, por ejemplo: arn:aws:dms:us-east-1:123456789012:rep:6EFQQO6U6EDPRCPKLNPL2SCEEY

2. Determine qué versiones de instancias de replicación están disponibles mediante el siguiente comando.

```
aws dms describe-orderable-replication-instances \
--query "OrderableReplicationInstances[*].[ReplicationInstanceClass,EngineVersion]"
```

En la salida, tome nota de los números de versión del motor que están disponibles para la clase de instancia de replicación. Debería ver esta información en la salida del paso 1.

3. Actualice la instancia de replicación utilizando el siguiente comando.

```
aws dms modify-replication-instance \
--replication-instance-arn arn \
--engine-version n.n.n
```

Sustituya *arn* en el ejemplo anterior por el ARN de instancia de replicación real del paso anterior.

Sustituya *n.n.n* por el número de versión del motor que desee, por ejemplo: 2.2.1

#### Note

La actualización de la instancia de replicación tarda varios minutos. Puede ver el estado de la instancia de replicación utilizando el siguiente comando.

```
aws dms describe-replication-instances \
--query "ReplicationInstances[*]. \
[ReplicationIdentifier,ReplicationInstanceStatus]"
```

Cuando la instancia de replicación esté lista, su estado cambiará a available.

## Configuración de una red para una instancia de replicación

AWS DMS siempre crea la instancia de replicación en una VPC basada en Amazon Virtual Private Cloud (Amazon VPC). Especifique la VPC donde que se encuentra la instancia de replicación. Puede utilizar su VPC predeterminada para su cuenta y región de AWS, o crear una nueva VPC. La VPC debe tener dos subredes en al menos una zona de disponibilidad.

La interfaz de red elástica (ENI) asignada a la instancia de replicación de una VPC debe asociarse a un grupo de seguridad con normas que permitan que todo el tráfico, en todos los puertos, salga de la VPC. Este enfoque permite que haya comunicación entre la instancia de replicación y los puntos de enlace de sus bases de datos de origen y de destino, siempre que en dichos puntos de enlace se hayan activado las reglas de salida correctas. Le recomendamos que utilice la configuración predeterminada para los puntos de enlace, la cual permite la salida en todos los puertos y a todas las direcciones.

Los puntos de enlace de origen y de destino acceden a la instancia de replicación que está dentro de la VPC conectando a la VPC o por estar dentro de la VPC. Los puntos de enlace de las bases de datos deben incluir listas de control de acceso (ACL) de red y reglas del grupo de seguridad (si es el caso) que permitan un acceso entrante desde la instancia de replicación. Dependiendo de la configuración de red que esté utilizando, puede utilizar el grupo de seguridad de la VPC de la instancia de replicación, la dirección IP privada o pública de la instancia de replicación o la dirección IP pública de la gateway NAT. Estas conexiones forman una red que se utiliza para migrar datos.

## Configuraciones de red para migrar bases de datos

Con AWS Database Migration Service puede utilizar varias configuraciones de red diferentes. Las siguientes son configuraciones comunes para una red utilizada para migrar bases de datos.

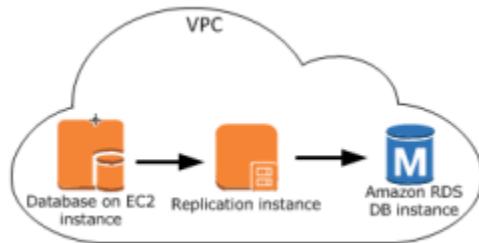
### Temas

- [Configuración con todos los componentes de migración de bases de datos en una VPC \(p. 72\)](#)
- [Configuración con dos VPC \(p. 72\)](#)
- [Configuración de una red a una VPC mediante AWS Direct Connect o una VPN \(p. 73\)](#)
- [Configuración de una red a una VPC mediante Internet \(p. 73\)](#)
- [Configuración con una instancia de base de datos de Amazon RDS que no está en una VPC a una instancia de base de datos en una VPC con ClassicLink \(p. 74\)](#)

## Configuración con todos los componentes de migración de bases de datos en una VPC

La red más sencilla para migrar las bases de datos es aquella en la que el punto de enlace de origen, la instancia de replicación y el punto de enlace de destino están todos en la misma VPC. Esta configuración es buena si los puntos de enlace de origen y de destino están en una instancia de base de datos de Amazon RDS o en una instancia Amazon EC2.

La siguiente ilustración muestra una configuración en la que una base de datos de una instancia Amazon EC2 se conecta a la instancia de replicación y los datos se migran a una instancia de base de datos de Amazon RDS.



El grupo de seguridad VPC utilizado en esta configuración debe permitir la entrada al puerto de la base de datos desde la instancia de replicación. Para ello, asegúrese de que el grupo de seguridad utilizado por la instancia de replicación entra a los puntos de enlace, o permita explícitamente la dirección IP privada de la instancia de replicación.

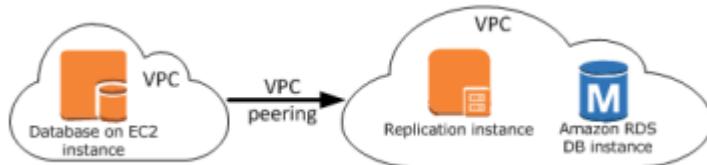
## Configuración con dos VPC

Si el punto de enlace de origen y los puntos de enlace de destino están en VPC diferentes, puede crear la instancia de replicación en una de las VPC y enlazar ambas mediante interconexión de VPC.

La interconexión de VPC es una conexión en red entre dos VPC que permite direccionar a través de las direcciones IP privadas de cada VPC, como si estuviesen en la misma red. Recomendamos este método para conectar las VPC que estén dentro de una región de AWS. Puede crear interconexiones de VPC entre sus propias VPC o con una VPC de otra cuenta de AWS dentro de la misma región de AWS. Para obtener más información acerca de las interconexiones de VPC, consulte la sección sobre [interconexión de VPC](#) en la Guía del usuario de Amazon VPC.

En la siguiente ilustración se muestra una configuración de ejemplo con interconexión de VPC. Aquí, la base de datos de origen en una instancia Amazon EC2 en una VPC se conecta por la interconexión de

VPC para una VPC. Esta VPC contiene la instancia de replicación y la base de datos de destino en una instancia de base de datos de Amazon RDS.

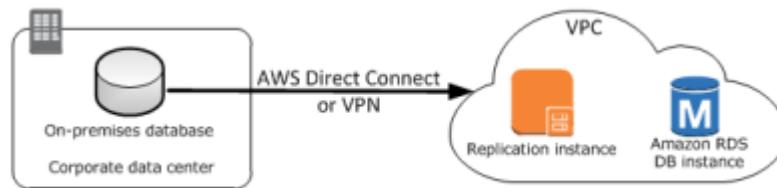


Los grupos de seguridad de VPC utilizados en esta configuración deben permitir la entrada al puerto de la base de datos desde la instancia de replicación.

## Configuración de una red a una VPC mediante AWS Direct Connect o una VPN

Las redes remotas pueden conectarse a una VPC a través de varias opciones, como AWS Direct Connect o una conexión VPN de software o de hardware. Estas opciones a menudo sirven para integrar servicios locales existentes, como los de monitorización, de autenticación, de seguridad, de datos u de otros sistemas, gracias a la ampliación de una red interna hacia la nube de AWS. Este tipo de extensión de red permite conectarse sin problemas a recursos alojados en AWS, como una VPC.

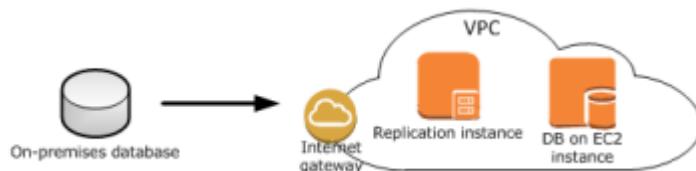
La siguiente ilustración muestra una configuración en la que el punto de enlace de origen es una base de datos local en un centro de datos corporativo. Se conecta mediante AWS Direct Connect o una VPN a una VPC que contiene la instancia de replicación y una base de datos de destino en una instancia de base de datos de Amazon RDS.



En esta configuración, el grupo de seguridad de la VPC debe incluir una regla de direccionamiento que envía el tráfico destinado a una dirección o un rango de direcciones IP concretos a un host. Este host debe ser capaz de conectar el tráfico de la VPC con la VPN local. En este caso, el host NAT incluye su propia configuración de grupo de seguridad para permitir el tráfico desde la dirección IP privada o el grupo de seguridad desde la instancia de replicación a la instancia NAT.

## Configuración de una red a una VPC mediante Internet

Si no utiliza una conexión VPN o AWS Direct Connect para conectarse a recursos de AWS, puede utilizar Internet para migrar una base de datos a una instancia Amazon EC2 o una instancia de base de datos de Amazon RDS. Esta configuración supone usar una instancia de replicación pública en una VPC con una gateway de Internet que contenga el punto de enlace de destino y la instancia de replicación.



Para añadir una gateway de Internet a la VPC, consulte la sección [Asociar una gateway de Internet](#) en la Guía del usuario de Amazon VPC.

El grupo de seguridad de la VPC debe incluir reglas de direccionamiento que envíen de forma predeterminada el tráfico no destinado a la VPC a la gateway de Internet. En esta configuración, la conexión al punto de enlace parece proceder de la dirección IP pública de la instancia de replicación, no de la dirección IP privada.

## Configuración con una instancia de base de datos de Amazon RDS que no está en una VPC a una instancia de base de datos en una VPC con ClassicLink

Puede utilizar ClassicLink con un servidor proxy para conectar una instancia de base de datos de Amazon RDS que no esté en una VPC a un servidor de replicación AWS DMS y una instancia de base de datos que residan en una VPC.

ClassicLink le permite enlazar una instancia de base de datos EC2-Classic a una VPC que esté en su cuenta, dentro de la misma región de AWS. Una vez que haya creado el enlace, la instancia de la base de datos de origen se podrá comunicar con la instancia de replicación dentro de la VPC a través de sus direcciones IP privadas.

Dado que la instancia de replicación en la VPC no puede obtener acceso directamente a la instancia de base de datos de origen en la plataforma EC2-Classic utilizando ClassicLink, debe utilizar un servidor proxy. El servidor proxy conecta la instancia de base de datos de origen a la VPC que contiene la instancia de replicación y la instancia de base de datos de destino. El servidor proxy utiliza ClassicLink para conectarse a la VPC. El reenvío de puertos en el servidor proxy permite que la instancia de base de datos de origen y la instancia de base de datos de destino en la VPC puedan comunicarse.



## Uso de ClassicLink con AWS Database Migration Service

Puede utilizar ClassicLink, junto con un servidor proxy, para conectar una instancia de base de datos de Amazon RDS que no esté en una VPC a un servidor de replicación AWS DMS y una instancia de base de datos que residan en una VPC.

El siguiente procedimiento muestra cómo utilizar ClassicLink para conectar una instancia de base de datos de origen de Amazon RDS que no esté en una VPC a una VPC y que incluya una instancia de replicación de AWS DMS y una instancia de base de datos de destino.

- Cree una instancia de replicación de AWS DMS en una VPC. (Todas las instancias de replicación se crean en una VPC).
- Asocie un grupo de seguridad de VPC a la instancia de replicación y a la instancia de base de datos de destino. Cuando dos instancias comparten un grupo de seguridad de VPC, pueden comunicarse entre sí de forma predeterminada.
- Configure un servidor proxy en una instancia EC2 Classic.
- Cree una conexión con ClassicLink entre el servidor proxy y la VPC.
- Cree puntos de enlace de AWS DMS para las bases de datos de origen y de destino.
- Cree una tarea de AWS DMS.

Para utilizar ClassicLink con el fin de migrar una base de datos en una instancia de base de datos que no esté en una VPC a una base de datos en una instancia de base de datos que está en una VPC

1. Paso 1: Crear una instancia de replicación de AWS DMS.

Para crear una instancia de replicación de AWS DMS y asignar un grupo de seguridad de VPC:

- a. Inicie sesión en la consola de administración de AWS y seleccione AWS Database Migration Service. Tenga en cuenta que si está registrado como usuario de AWS Identity and Access Management (IAM), debe disponer de los permisos adecuados para acceder a AWS DMS. Para obtener más información sobre los permisos necesarios para migrar bases de datos, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).
- b. En la página Dashboard, elija Replication Instance. Siga las instrucciones que aparecen en [Paso 2: Crear una instancia de replicación \(p. 20\)](#) para crear una instancia de replicación.
- c. Una vez que haya creado la instancia de replicación de AWS DMS, abra la consola del servicio de EC2. Seleccione Network Interfaces en el panel de navegación.
- d. Seleccione DMSNetworkInterface y elija Change Security Groups en el menú Actions.
- e. Seleccione el grupo de seguridad que desea utilizar para la instancia de replicación y la instancia de base de datos de destino.

2. Paso 2: Asociar el grupo de seguridad del último paso con la instancia de base de datos de destino.

Para asociar un grupo de seguridad con una instancia de base de datos

- a. Abra la consola del servicio de Amazon RDS. Seleccione Instances en el panel de navegación.
  - b. Seleccione la instancia de base de datos de destino. En Instance Actions seleccione Modify.
  - c. Para el parámetro Security Group, seleccione el grupo de seguridad que ha utilizado en el paso anterior.
  - d. Seleccione Continue y, a continuación, elija Modify DB Instance.
3. Paso 3: Configurar un servidor proxy en una instancia EC2 Classic con NGINX. Utilice una AMI de su elección para lanzar una instancia EC2 Classic. El ejemplo siguiente se basa en la AMI Ubuntu Server 14.04 LTS (HVM).

Para configurar un servidor proxy en una instancia EC2 Classic

- a. Establezca una conexión con la instancia EC2 Classic e instale NGINX con los siguientes comandos:

```
Prompt> sudo apt-get update
Prompt> sudo wget http://nginx.org/download/nginx-1.9.12.tar.gz
Prompt> sudo tar -xvzf nginx-1.9.12.tar.gz
Prompt> cd nginx-1.9.12
Prompt> sudo apt-get install build-essential
Prompt> sudo apt-get install libpcre3 libpcre3-dev
Prompt> sudo apt-get install zlib1g-dev
Prompt> sudo ./configure --with-stream
Prompt> sudo make
Prompt> sudo make install
```

- b. Edite el archivo NGINX daemon, /etc/init/nginx.conf, con el siguiente código:

```
# /etc/init/nginx.conf - Upstart file
description "nginx http daemon"
Version de API API Version 2016-01-01
```

```
author "email"

start on (filesystem and net-device-up IFACE=lo)
stop on runlevel [!2345]

env DAEMON=/usr/local/nginx/sbin/nginx
env PID=/usr/local/nginx/logs/nginx.pid

expect fork
respawn
respawn limit 10 5

pre-start script
    $DAEMON -t
    if [ $? -ne 0 ]
        then exit $?
    fi
end script

exec $DAEMON
```

- c. Cree un archivo de configuración NGINX en /usr/local/nginx/conf/nginx.conf. En el archivo de configuración, añada lo siguiente:

```
# /usr/local/nginx/conf/nginx.conf - NGINX configuration file

worker_processes 1;

events {
    worker_connections 1024;
}

stream {
    server {
        listen <DB instance port number>;
        proxy_pass <DB instance identifier>:<DB instance port number>;
    }
}
```

- d. Desde la línea de comandos, inicie NGINX con los siguientes comandos:

```
Prompt> sudo initctl reload-configuration
Prompt> sudo initctl list | grep nginx
Prompt> sudo initctl start nginx
```

4. Paso 4: Crear una conexión de ClassicLink entre el servidor proxy y la VPC de destino que contenga la instancia de base de datos de destino y la instancia de replicación

Utilizar ClassicLink para conectar el servidor proxy con la VPC de destino

- a. Abra la consola de EC2 y seleccione la instancia EC2 Classic que se ejecuta el servidor proxy.
- b. Seleccione ClassicLink en Actions y, a continuación, seleccione Link to VPC.
- c. Seleccione el grupo de seguridad que haya utilizado anteriormente en este procedimiento.
- d. Seleccione Link to VPC.

5. Paso 5: Crear los puntos de enlace de AWS DMS con el procedimiento en [Paso 3: Especificar los puntos de enlace de origen y destino \(p. 25\)](#). Debe utilizar el nombre de host EC2 DNS interno del proxy como el nombre del servidor cuando especifique el punto de enlace de origen.
6. Paso 6: Crear una tarea de AWS DMS con el procedimiento en [Paso 4: Crear una tarea \(p. 30\)](#).

## Creación de un grupo de subred de replicación

Dentro de la red que utilizará para migrar bases de datos, deberá especificar qué subredes de su nube Amazon Virtual Private Cloud (Amazon VPC) tiene pensado utilizar. Una subred es un rango de direcciones IP en la VPC dentro de una determinada zona de disponibilidad. Estas subredes pueden distribuirse entre las zonas de disponibilidad de la región de AWS en la que se encuentra la VPC.

Una vez creada una instancia de replicación en la subred que seleccione, podrá usar la consola de AWS DMS para gestionar qué subred utilizará un punto de enlace de origen o de destino.

Puede crear un grupo de subred de replicación para definir qué subredes se deben utilizar. Deberá especificar al menos una subred en dos zonas de disponibilidad diferentes.

Para crear un grupo de subred de replicación

1. Inicie sesión en la Consola de administración de AWS y elija AWS Database Migration Service. Si está registrado como usuario de AWS Identity and Access Management (IAM), debe disponer de los permisos adecuados para acceder a AWS DMS. Para obtener más información sobre los permisos necesarios para migrar bases de datos, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).
2. En el panel de navegación, elija Subnet Groups.
3. Elija Create Subnet Group.
4. En la página Edit Replication Subnet Group que se muestra a continuación especifique su información del grupo de subred de replicación. La tabla siguiente describe la configuración.

### Edit Replication Subnet Group

Identifier	DMS-sngrp																																																						
Description	default VPC subnet group																																																						
VPC	vpc-fd8c1b99 - dms-test-vpc																																																						
Add Subnet(s) to this Subnet Group. You may add subnets one at a time or add all the subnets related to this VPC. You may make additions/edits after this group is created.																																																							
<table border="1"><thead><tr><th colspan="3">Available Subnets</th><th colspan="3">Subnet Group</th></tr><tr><th>AZ</th><th>Subnet</th><th>CIDR</th><th>AZ</th><th>Subnet</th><th>CIDR</th></tr></thead><tbody><tr><td><input type="checkbox"/></td><td>us-east-1a</td><td>subnet-4d802a3b</td><td>172.30.5.0/28</td><td><input type="button" value="Add"/></td><td>No records found.</td></tr><tr><td><input checked="" type="checkbox"/></td><td>us-east-1a</td><td>subnet-6bec2f1d</td><td>172.30.0.0/24</td><td><input type="button" value="Remove"/></td><td></td></tr><tr><td><input type="checkbox"/></td><td>us-east-1b</td><td>subnet-37b3566f</td><td>172.30.1.0/24</td><td><input type="button" value="Reset"/></td><td></td></tr><tr><td><input checked="" type="checkbox"/></td><td>us-east-1b</td><td>subnet-3b945863</td><td>172.30.6.0/28</td><td></td><td></td></tr><tr><td><input checked="" type="checkbox"/></td><td>us-east-1c</td><td>subnet-d6bc43b3</td><td>172.30.2.0/24</td><td></td><td></td></tr><tr><td><input type="checkbox"/></td><td>us-east-1d</td><td>subnet-68ce7755</td><td>172.30.3.0/24</td><td></td><td></td></tr><tr><td><input type="checkbox"/></td><td>us-east-1e</td><td>subnet-b38bf9f98</td><td>172.30.4.0/24</td><td></td><td></td></tr></tbody></table>		Available Subnets			Subnet Group			AZ	Subnet	CIDR	AZ	Subnet	CIDR	<input type="checkbox"/>	us-east-1a	subnet-4d802a3b	172.30.5.0/28	<input type="button" value="Add"/>	No records found.	<input checked="" type="checkbox"/>	us-east-1a	subnet-6bec2f1d	172.30.0.0/24	<input type="button" value="Remove"/>		<input type="checkbox"/>	us-east-1b	subnet-37b3566f	172.30.1.0/24	<input type="button" value="Reset"/>		<input checked="" type="checkbox"/>	us-east-1b	subnet-3b945863	172.30.6.0/28			<input checked="" type="checkbox"/>	us-east-1c	subnet-d6bc43b3	172.30.2.0/24			<input type="checkbox"/>	us-east-1d	subnet-68ce7755	172.30.3.0/24			<input type="checkbox"/>	us-east-1e	subnet-b38bf9f98	172.30.4.0/24		
Available Subnets			Subnet Group																																																				
AZ	Subnet	CIDR	AZ	Subnet	CIDR																																																		
<input type="checkbox"/>	us-east-1a	subnet-4d802a3b	172.30.5.0/28	<input type="button" value="Add"/>	No records found.																																																		
<input checked="" type="checkbox"/>	us-east-1a	subnet-6bec2f1d	172.30.0.0/24	<input type="button" value="Remove"/>																																																			
<input type="checkbox"/>	us-east-1b	subnet-37b3566f	172.30.1.0/24	<input type="button" value="Reset"/>																																																			
<input checked="" type="checkbox"/>	us-east-1b	subnet-3b945863	172.30.6.0/28																																																				
<input checked="" type="checkbox"/>	us-east-1c	subnet-d6bc43b3	172.30.2.0/24																																																				
<input type="checkbox"/>	us-east-1d	subnet-68ce7755	172.30.3.0/24																																																				
<input type="checkbox"/>	us-east-1e	subnet-b38bf9f98	172.30.4.0/24																																																				

Para esta opción	Haga lo siguiente
Identifier	Escriba un nombre para el grupo de subred de replicación que contenga entre 8 y 16 caracteres ASCII imprimibles (excluidos los símbolos /, " y @). El nombre de la cuenta debe ser único en la región de AWS que haya seleccionado. Puede optar por asignar un nombre descriptivo, como incluir la región de AWS y la tarea que esté realizando, por ejemplo <b>DMS-default-VPC</b> .
Descripción	Escriba una breve descripción del grupo de subred de replicación.
VPC	Elija la VPC que desea usar para migrar la base de datos. Tenga en cuenta que la VPC debe tener al menos una subred en dos zonas de disponibilidad como mínimo.
Available Subnets	Elija las subredes que deseé incluir en el grupo de subred de replicación. Debe seleccionar subredes en dos zonas de disponibilidad como mínimo.

5. Elija Add para añadir las subredes al grupo de subred de replicación.
6. Seleccione Create.

## Establecimiento de una clave de cifrado para una instancia de replicación

AWS DMS cifra el almacenamiento utilizado por una instancia de replicación y la información de conexión del punto de enlace. Para cifrar los datos almacenados que utiliza una instancia de replicación, AWS DMS se sirve de una clave maestra exclusiva en su cuenta de AWS. Puede ver y administrar esta clave maestra con AWS Key Management Service (AWS KMS). Puede utilizar la clave maestra predeterminada de su cuenta (aws/dms) o una clave maestra personalizada que usted mismo cree. Si ya posee una clave de cifrado de AWS KMS, también la podrá utilizar para el cifrado.

Puede especificar su propia clave de cifrado suministrando un identificador de clave KMS para cifrar sus recursos de AWS DMS. Cuando especifique su propia clave de cifrado, la cuenta de usuario utilizada para migrar la base de datos deberá tener acceso a ella. Para obtener más información sobre cómo crear sus propias claves de cifrado y proporcionar a los usuarios acceso a una clave de cifrado, consulte la [guía para desarrolladores de AWS KMS](#).

Si no especifica un identificador de clave KMS, AWS DMS utilizará su clave de cifrado predeterminada. KMS crea la clave de cifrado predeterminada para AWS DMS para su cuenta de AWS. Cada cuenta de AWS dispone de una clave de cifrado predeterminada diferente para cada región de AWS.

Para gestionar las claves que se utilizan para cifrar sus recursos de AWS DMS debe utilizar KMS. Encontrará KMS en la Consola de administración de AWS eligiendo Identity & Access Management (Administración de identidad y acceso) en la página de inicio de la consola y luego Encryption Keys (Claves de cifrado) en el panel de navegación.

KMS combina hardware y software seguros y altamente disponibles para ofrecer un sistema de gestión de claves escalado a la nube. Si utiliza KMS, puede crear claves de cifrado y definir las políticas que controlan cómo se pueden utilizar dichas claves. KMS es compatible con AWS CloudTrail, lo que permite auditar el uso de claves para comprobar que las claves se utilizan de forma adecuada. Las claves de KMS se pueden utilizar en combinación con AWS DMS y admitir servicios de AWS tales como Amazon RDS, Amazon S3, Amazon Elastic Block Store (Amazon EBS) y Amazon Redshift.

Cuando haya creado sus recursos de AWS DMS con una clave de cifrado específica, no podrá modificar la clave de cifrado de esos recursos. Asegúrese de determinar los requisitos de clave de cifrado antes de crear sus recursos de AWS DMS.

## Creación una instancia de replicación

La primera tarea a la hora de migrar una base de datos consiste en crear una instancia de replicación que tenga almacenamiento y capacidad de procesamiento suficientes para realizar las tareas que se asignan y migrar datos desde la base de datos de origen a la base de datos de destino. El tamaño necesario para esta instancia varía en función de la cantidad de datos que deba migrar y las tareas que necesita que efectúe la instancia. Para obtener más información sobre las instancias de replicación, consulte [Trabajar con una instancia de replicación de AWS DMS \(p. 62\)](#).

En el procedimiento siguiente, se presupone que va a utilizar el asistente de la consola de AWS DMS. Puede efectuar este paso si selecciona Replication instances (Instancias de replicación) en el panel de navegación de la consola de AWS DMS y, a continuación, selecciona Create replication instance (Crear instancias de replicación).

Para crear una instancia de replicación con la consola de AWS

1. En la página Create replication instance especifique la información de la instancia de replicación. La tabla siguiente describe la configuración.

### Create replication instance

A replication instance initiates the connection between the source and target databases, transfers the data, and caches any changes that occur on the source database during the initial data load. Use the fields below to configure the parameters of your new replication instance including network and security information, encryption details, and performance characteristics. We suggest you shut down the replication instance once your migration is complete to prevent further usage charges.

Name*	e.g. production-replication-server	
Description*	e.g. migrates prod data	
Instance class*	dms.t2.medium	
Replication engine version*	2.3.0	
VPC*	vpc-	
Multi-AZ	No	
Publicly accessible	<input checked="" type="checkbox"/>	

Para esta opción	Haga lo siguiente
Nombre	Escriba un nombre para la instancia de replicación que contenga entre 8 y 16 caracteres ASCII imprimibles

Para esta opción	Haga lo siguiente
	(excluidos /, " y @). El nombre de la cuenta debe ser único en la región de AWS que haya seleccionado. Puede optar por asignar un nombre descriptivo, como incluir la región de AWS y la tarea que esté realizando, por ejemplo <b>west2-mysql2mysql-instance1</b> .
Descripción	Escriba una breve descripción para la instancia de replicación.
Instance class	Elija una clase de instancia con la configuración que necesita para la migración. Tenga en cuenta que la instancia debe tener suficiente capacidad de almacenamiento, red y procesamiento para realizar correctamente la migración. Para obtener más información sobre cómo determinar qué clase de instancia es la mejor opción para la migración, consulte <a href="#">Trabajar con una instancia de replicación de AWS DMS (p. 62)</a> .
Replication engine version	De forma predeterminada, la instancia de replicación ejecuta la última versión del software del motor de replicación de AWS DMS. Le recomendamos que acepte este valor predeterminado; no obstante, puede elegir una versión de motor anterior en caso de que sea necesario.
VPC	Seleccione la Amazon Virtual Private Cloud (Amazon VPC) que desee utilizar. Si la base de datos de origen o de destino se encuentra en una VPC, seleccione esa VPC. Si las bases de datos de origen y de destino se encuentran en diferentes VPC, asegúrese de que ambas se encuentran en subredes públicas y son de acceso público y, a continuación, seleccione la VPC donde se va a ubicar la instancia de replicación. Es preciso que la instancia de replicación tenga acceso a los datos en la VPC de origen. Si las bases de datos de origen o destino no se encuentran en una VPC, seleccione la VPC donde se vaya a ubicar la instancia de replicación.
Varias AZ	Use este parámetro opcional para crear una réplica en espera de la instancia de replicación en otra zona de disponibilidad para el soporte de commutación por error. Si va a utilizar la captura de datos de cambios (CDC) o la replicación continua, debe habilitar esta opción.
Publicly accessible	Seleccione esta opción si desea que la instancia de replicación esté accesible desde Internet.

2. Seleccione la pestaña Advanced que se muestra más abajo para establecer valores para la configuración de red y cifrado de la red en caso de que lo necesite. La tabla siguiente describe la configuración.

▼ Advanced

Allocated storage (GB)\*  

Replication Subnet Group\*  

Availability zone\*  

VPC Security Group(s)   
rds-launch-wizard-1  
rds-launch-wizard  
rds-[REDACTED]

KMS master key  

Description Default master key that protects my  
DMS replication instance volumes when  
no other key is defined

Account [REDACTED]

Key ARN arn:aws:kms:us-  
west-2:[REDACTED]  
[REDACTED]

Para esta opción	Haga lo siguiente
Allocated storage (GB)	<p>La capacidad de almacenamiento la consumen básicamente los archivos de registro y las transacciones que se almacenan en la memoria caché. En el caso de las transacciones en la memoria caché, el almacenamiento se utiliza únicamente cuando las transacciones en memoria caché se deben escribir en el disco. Por lo tanto, AWS DMS no utiliza una cantidad significativa de almacenamiento. Entre las excepciones se incluyen las siguientes:</p> <ul style="list-style-type: none"> <li>• Tablas muy grandes que conllevan una carga de transacciones importante. Cargar una tabla muy grande puede llevar su tiempo, por tanto hay más probabilidades de que las transacciones almacenadas en la memoria caché se escriban en el disco cuando se carga una tabla de gran tamaño.</li> <li>• Las tareas se configuran para detenerse antes de cargar las transacciones en caché. En este caso, todas las transacciones se almacenan en la memoria caché hasta que finaliza la carga completa de todas las tablas. Con esta configuración, es posible que las transacciones en la memoria caché consuman una cantidad considerable de espacio de almacenamiento.</li> <li>• Tareas configuradas con tablas que se cargan en Amazon Redshift. Sin embargo, esta configuración no comporta problema alguno si Amazon Aurora es el destino.</li> </ul> <p>En la mayoría de los casos, la asignación de almacenamiento es suficiente. Sin embargo, es siempre aconsejable prestar atención a las métricas relativas al almacenamiento y ampliar la capacidad de almacenamiento si determina que el consumo es mayor que lo que se ha asignado de forma predeterminada.</p>
Replication Subnet Group	Elija el grupo de subred de replicación en la VPC que haya seleccionado si quiere que se cree la instancia de replicación. Si la base de datos de origen está en una VPC, seleccione el grupo de subred que contiene la base de datos de origen como ubicación para la instancia de replicación. Para obtener más información sobre los grupos de subred para la replicación, consulte <a href="#">Creación de un grupo de subred de replicación (p. 77)</a> .
Availability zone	Seleccione la zona de disponibilidad en la que se encuentra la base de datos de origen.
VPC Security group(s)	La instancia de replicación se crea en una VPC. Si la base de datos de origen está en una VPC, seleccione el grupo de seguridad de la VPC que proporciona acceso a la instancia de base de datos donde reside la base de datos.

Para esta opción	Haga lo siguiente
Clave maestra de KMS	Elija la clave de cifrado que se utilizará para cifrar el almacenamiento de la replicación y la información de la conexión. Si elige (Default) aws/dms (Valor predeterminado) aws/dms, se utilizará la clave de AWS Key Management Service (AWS KMS) asociada con su cuenta y región de AWS. Se muestran una descripción y su número de cuenta junto con el ARN de la clave. Para obtener más información sobre cómo utilizar la clave de cifrado, consulte <a href="#">Configuración de una clave de cifrado y especificación de permisos KMS (p. 49)</a> .

3. Especifique la configuración Maintenance. La tabla siguiente describe la configuración. Para obtener más información sobre la configuración de mantenimiento, consulte [Ventana de mantenimiento de AWS DMS \(p. 66\)](#).

▼ Maintenance

Auto minor version upgrade

Maintenance window\*

Tuesday	03	:	0.5	(UTC-7)
				hrs.

**Create replication instance**

Para esta opción	Haga lo siguiente
Auto minor version upgrade	Seleccione esta opción para que las actualizaciones de motor secundarias se apliquen automáticamente a la instancia de replicación durante el periodo de mantenimiento.
Maintenance window	Seleccione el intervalo de tiempo semanal durante el cual puede llevarse a cabo el mantenimiento del sistema, en tiempo universal coordinado (UTC).  Valor predeterminado: un periodo de 30 minutos seleccionado al azar de un bloque de 8 horas de tiempo por región de AWS, que tiene lugar un día de la semana al azar.

4. Elija Create replication instance.

## Modificación de una instancia de replicación

Puede modificar la configuración de una instancia de replicación para, por ejemplo, cambiar la clase de instancia o para aumentar el almacenamiento.

Al modificar una instancia de replicación, puede aplicar los cambios inmediatamente. Para aplicar los cambios inmediatamente, seleccione la opción *Apply changes immediately* (Aplicar los cambios inmediatamente) en la Consola de administración de AWS, utilice el parámetro `--apply-immediately` al llamar a la CLI de AWS o establezca el parámetro `ApplyImmediately` en `true` al utilizar la API de AWS DMS.

Si decide no aplicar los cambios inmediatamente, estos se colocan en la cola de modificaciones pendientes. Los cambios pendientes en la cola se aplican durante el siguiente periodo de mantenimiento.

**Note**

Si opta por aplicar los cambios inmediatamente, también se aplican los cambios de la cola de modificaciones pendientes. Si alguna de las modificaciones pendientes requiere un tiempo de inactividad, al elegir *Apply changes immediately* (Aplicar cambios inmediatamente) puede causar un tiempo de inactividad imprevisto.

Para modificar una instancia de replicación con la consola de AWS

1. Inicie sesión en la Consola de administración de AWS y elija AWS DMS.
2. En el panel de navegación, elija *Replication instances*.
3. Elija la instancia de replicación que desee modificar. En la siguiente tabla se describen las modificaciones que puede realizar.

Para esta opción	Haga lo siguiente
Nombre	Puede cambiar el nombre de la instancia de replicación. Escriba un nombre para la instancia de replicación que contenga entre 8 y 16 caracteres ASCII imprimibles (excluidos /, " y @). El nombre de la cuenta debe ser único en la región de AWS que haya seleccionado. Puede optar por asignar un nombre descriptivo, como incluir la región de AWS y la tarea que esté realizando, por ejemplo <b>west2-mysql2mysql-instance1</b> .
Instance class	Puede cambiar la clase de instancia. Elija una clase de instancia con la configuración que necesita para la migración. El cambio de la clase de instancia provoca el reinicio de la instancia de replicación. Este reinicio se produce en el siguiente periodo de mantenimiento o puede producirse inmediatamente si selecciona la opción <i>Apply changes immediately</i> .  Para obtener más información sobre cómo determinar qué clase de instancia es la mejor opción para la migración, consulte <a href="#">Trabajar con una instancia de replicación de AWS DMS (p. 62)</a> .
Replication engine version	Puede actualizar la versión del motor que la instancia de replicación utiliza. Si actualiza la versión de motor de replicación, la instancia de replicación se desactivará mientras se esté actualizando.
Varias AZ	Puede cambiar esta opción para crear una réplica en espera de la instancia de replicación en otra zona de disponibilidad como soporte en caso de conmutación por error o quitar esta opción. Si tiene previsto utilizar la captura de datos de cambios (CDC) o replicación continua, debe habilitar esta opción.

Para esta opción	Haga lo siguiente
Allocated storage (GB)	<p>La capacidad de almacenamiento la consumen básicamente los archivos de registro y las transacciones que se almacenan en la memoria caché. En el caso de las transacciones en la memoria caché, el almacenamiento se utiliza únicamente cuando las transacciones en memoria caché se deben escribir en el disco. Por lo tanto, AWS DMS no utiliza una cantidad significativa de almacenamiento. Entre las excepciones se incluyen las siguientes:</p> <ul style="list-style-type: none"> <li>• Tablas muy grandes que llevan una carga de transacciones importante. Cargar una tabla muy grande puede llevar su tiempo, por tanto hay más probabilidades de que las transacciones almacenadas en la memoria caché se escriban en el disco cuando se carga una tabla de gran tamaño.</li> <li>• Las tareas se configuran para detenerse antes de cargar las transacciones en caché. En este caso, todas las transacciones se almacenan en la memoria caché hasta que finaliza la carga completa de todas las tablas. Con esta configuración, es posible que las transacciones en la memoria caché consuman una cantidad considerable de espacio de almacenamiento.</li> <li>• Tareas configuradas con tablas que se cargan en Amazon Redshift. Sin embargo, esta configuración no comporta problema alguno si Amazon Aurora es el destino.</li> </ul> <p>En la mayoría de los casos, la asignación de almacenamiento es suficiente. Sin embargo, es siempre aconsejable prestar atención a las métricas relativas al almacenamiento y ampliar la capacidad de almacenamiento si determina que el consumo es mayor de lo que se ha asignado de forma predeterminada.</p>
VPC Security Group(s)	La instancia de replicación se crea en una VPC. Si la base de datos de origen está en una VPC, seleccione el grupo de seguridad de la VPC que proporciona acceso a la instancia de base de datos donde reside la base de datos.
Auto minor version upgrade	Seleccione esta opción para aplicar automáticamente las actualizaciones del motor secundarias a la instancia de replicación durante el periodo de mantenimiento o inmediatamente si selecciona la opción <i>Apply changes immediately</i> .
Maintenance window	<p>Seleccione el intervalo de tiempo semanal durante el cual puede llevarse a cabo el mantenimiento del sistema, en tiempo universal coordinado (UTC).</p> <p>Valor predeterminado: un periodo de 30 minutos seleccionado al azar de un bloque de 8 horas de tiempo por región de AWS, que tiene lugar un día de la semana al azar.</p>

Para esta opción	Haga lo siguiente
Apply changes immediately	Seleccione esta opción para aplicar inmediatamente cualquier modificación que haya realizado. Según la configuración que elija, si selecciona esta opción se producirá un reinicio inmediato de la instancia de replicación.

## Reinicio de una instancia de replicación

Puede reiniciar una instancia de replicación de AWS DMS para reiniciar el motor de replicación. Cuando se reinicia una instancia de replicación, se produce una interrupción momentánea en esta, durante la cual su estado se establece en Rebooting (Reiniciando). Si la instancia de AWS DMS está configurada para Multi-AZ, el reinicio puede realizarse con una conmutación por error. Cuando finaliza el reinicio, se crea un evento de AWS DMS.

Si la instancia de AWS DMS es un despliegue Multi-AZ, es posible forzar una conmutación por error desde una zona de disponibilidad de AWS a otra cuando se reinicia. Cuando se fuerza una conmutación por error de la instancia de AWS DMS, AWS DMS cambia automáticamente a una instancia de espera de otra zona de disponibilidad. El reinicio con conmutación por error es interesante cuando se desea simular un error en una instancia de AWS DMS para realizar pruebas.

Si hay tareas de migración ejecutándose en la instancia de replicación cuando se produce un reinicio, no se produce pérdida de datos y la tarea se reanuda una vez que haya finalizado el reinicio. Si las tablas de la tarea de migración se encuentran en la mitad de una carga masiva (fase de carga completa), DMS reinicia la migración de estas tablas desde el principio. Si las tablas de la tarea de migración se encuentran en la fase de replicación continua, la tarea se reanuda una vez que se haya completado el reinicio.

No se puede reiniciar una instancia de replicación de AWS DMS que no se encuentra en el estado Available (Disponible). La instancia de AWS DMS puede no estar disponible por varias razones, como una modificación solicitada anteriormente o una acción durante un periodo de mantenimiento. El tiempo necesario para reiniciar una instancia de replicación de AWS DMS suele ser corto (menos de 5 minutos).

## Reiniciar una instancia de replicación con la consola de AWS

Para reiniciar una instancia de replicación, utilice la consola de AWS.

Para reiniciar una instancia de replicación con la consola de AWS

1. Inicie sesión en la Consola de administración de AWS y elija AWS DMS.
2. En el panel de navegación, elija Replication instances.
3. Elija la instancia de replicación que desea reiniciar.
4. Elija Reboot.
5. En el cuadro de diálogo Reboot replication instance (Reiniciar instancia de replicación), elija Reboot With Failover? (¿Desea reiniciar con conmutación por error?) si ha configurado la instancia de replicación para implementación Multi-AZ y desea realizar la conmutación por error a otra zona de disponibilidad de AWS.
6. Elija Reboot.

## Reinicio de una instancia de replicación utilizando la CLI

Para reiniciar una instancia de replicación, utilice el comando `reboot-replication-instance` de la AWS CLI con el parámetro siguiente:

- --replication-instance-arn

#### Example Ejemplo de reinicio normal

En el siguiente ejemplo de la AWS CLI, se reinicia una instancia de replicación.

```
aws dms reboot-replication-instance \  
--replication-instance-arn arnofmyrepinstnace
```

#### Example Ejemplo de reinicio normal con conmutación por error

En el siguiente ejemplo de la AWS CLI, se reinicia una instancia de replicación con conmutación por error.

```
aws dms reboot-replication-instance \  
--replication-instance-arn arnofmyrepinstnace \  
--force-failover
```

## Reinicio de una instancia de replicación utilizando la API

Para reiniciar una instancia de replicación, utilice la acción `RebootReplicationInstance` de la API de AWS DMS con el parámetro siguiente:

- `ReplicationInstanceArn = arnofmyrepinstnace`

#### Example Ejemplo de reinicio normal

En el siguiente ejemplo de código, se reinicia una instancia de replicación.

```
https://dms.us-west-2.amazonaws.com/  
?Action=RebootReplicationInstance  
&DBInstanceArn=arnofmyrepinstnace  
&SignatureMethod=HmacSHA256  
&SignatureVersion=4  
&Version=2014-09-01  
&X-Amz-Algorithm=AWS4-HMAC-SHA256  
&X-Amz-Credential=AKIADQKE4SARGYLE/20140425/us-east-1/dms/aws4_request  
&X-Amz-Date=20140425T192732Z  
&X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date  
&X-Amz-Signature=1dc9dd716f4855e9bdf188c70f1cf9f6251b070b68b81103b59ec70c3e7854b3
```

#### Example Ejemplo de reinicio normal con conmutación por error

En el siguiente ejemplo de código se reinicia una instancia de replicación y se realiza la conmutación por error a otra zona de disponibilidad de AWS.

```
https://dms.us-west-2.amazonaws.com/  
?Action=RebootReplicationInstance  
&DBInstanceArn=arnofmyrepinstnace  
&ForceFailover=true  
&SignatureMethod=HmacSHA256  
&SignatureVersion=4  
&Version=2014-09-01  
&X-Amz-Algorithm=AWS4-HMAC-SHA256  
&X-Amz-Credential=AKIADQKE4SARGYLE/20140425/us-east-1/dms/aws4_request  
&X-Amz-Date=20140425T192732Z  
&X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
```

&X-Amz-Signature=1dc9dd716f4855e9bdf188c70f1cf9f6251b070b68b81103b59ec70c3e7854b3

## Eliminación de una instancia de replicación

Es posible eliminar una instancia de replicación AWS DMS cuando haya terminado de usarla. Si tiene tareas de migración que utilizan la instancia de replicación, debe detener y eliminar las tareas antes de eliminar la instancia de replicación.

Si cierra su cuenta de AWS, todos los recursos de AWS DMS y configuraciones asociadas a su cuenta se eliminan al cabo de dos días. Estos recursos incluyen todas las instancias de replicación, configuración de punto de enlace de origen y de destino, tareas de replicación y certificados SSL. Si después de dos días decide volver a utilizar AWS DMS, vuelva a crear los recursos necesarios.

### Eliminar una instancia de replicación con la consola de AWS

Para eliminar una instancia de replicación, utilice la consola de AWS.

Para eliminar una instancia de replicación utilizando la consola de AWS

1. Inicie sesión en la Consola de administración de AWS y elija AWS DMS.
2. En el panel de navegación, elija Replication instances.
3. Elija la instancia de replicación que desea eliminar.
4. Elija Eliminar.
5. En el cuadro de diálogo (Confirmación), elija Delete (Eliminar).

### Eliminar una instancia de replicación con la CLI

Para eliminar una instancia de replicación, utilice el comando `delete-replication-instance` de la AWS CLI con el parámetro siguiente:

- `--replication-instance-arn`

Example Ejemplo de eliminación

En el siguiente ejemplo de la AWS CLI, se elimina una instancia de replicación.

```
aws dms delete-replication-instance \
--replication-instance-arn <arnofmyrepinstnace>
```

### Eliminar una instancia de replicación con la API

Para eliminar una instancia de replicación, utilice la acción `DeleteReplicationInstance` de la API de AWS DMS con los parámetros siguientes:

- `ReplicationInstanceArn = <arnofmyrepinstnace>`

Example Ejemplo de eliminación

El siguiente ejemplo de código elimina una instancia de replicación.

```
https://dms.us-west-2.amazonaws.com/
```

```
?Action=DeleteReplicationInstance
&DBInstanceArn=arn:aws:dms:us-east-1:123456789012:rep:myrepinst
&SignatureMethod=HmacSHA256
&SignatureVersion=4
&Version=2014-09-01
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIADQKE4SARGYLE/20140425/us-east-1/dms/aws4_request
&X-Amz-Date=20140425T192732Z
&X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
&X-Amz-Signature=1dc9dd716f4855e9bdf188c70f1cf9f6251b070b68b81103b59ec70c3e7854b3
```

## Instrucciones DDL compatibles con AWS DMS

Durante el proceso de migrar datos es posible ejecutar instrucciones en lenguaje de definición de datos (DDL) en la base de datos de origen. El servidor de replicación replica estas instrucciones en la base de datos de destino.

Las instrucciones DDL compatibles permiten realizar las siguientes acciones:

- Crear tablas
- Eliminar tablas
- Cambiar de nombre las tablas
- Agregar columnas
- Eliminar columnas
- Cambiar el nombre de las columnas
- Cambiar el tipo de datos de las columnas

Para obtener información sobre qué instrucciones DDL son compatibles con un origen específico, consulte el tema que describa dicho origen.

# Trabajo con puntos de enlace de AWS DMS

Un punto de enlace proporciona conexión, tipo de almacén de datos, y la ubicación información acerca de su almacén de datos. AWS Database Migration Service utiliza esta información para conectarse a un almacén de datos y migrar datos desde un punto de enlace de origen a un punto de enlace de destino. Puede especificar atributos de conexión adicionales para un punto de enlace mediante la utilización de atributos de conexión adicionales. Estos atributos pueden controlar el inicio de sesión, el tamaño del archivo y otros parámetros. Para obtener más información sobre atributos de conexión adicionales, consulte la sección de la documentación relacionada con su almacén de datos.

A continuación, encontrará más detalles acerca de los puntos de enlace.

## Temas

- [Orígenes para la migración de datos \(p. 90\)](#)
- [Destinos para la migración de datos \(p. 157\)](#)
- [Creación de puntos de enlace de origen y de destino \(p. 246\)](#)

## Orígenes para la migración de datos

AWS Database Migration Service (AWS DMS) puede utilizar la mayoría de los motores de datos más conocidos como origen para la replicación de datos. El origen de la base de datos puede hallarse en un motor autoadministrado en ejecución en una instancia Amazon Elastic Compute Cloud (Amazon EC2) o en una base de datos local. O puede ser un origen de datos en un servicio administrado por Amazon, como Amazon Relational Database Service (Amazon RDS) o Amazon S3.

A continuación, indicamos los orígenes válidos para AWS DMS:

### Bases de datos locales y de instancia Amazon EC2

- Versiones de Oracle 10.2 y posteriores, 11g y hasta 12.2, para las ediciones Enterprise, Standard, Standard One y Standard Two.
- Versiones de Microsoft SQL Server 2005, 2008, 2008R2, 2012, 2014 y 2016 para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.
- Versiones de MySQL 5.5, 5.6 y 5.7.
- MariaDB (se admite como origen de datos compatible con MySQL).
- PostgreSQL 9.4 y posteriores.
- SAP Adaptive Server Enterprise (ASE) versiones 12.5.3 o superior, 15, 15.5, 15.7, 16 y posteriores.
- Versiones 2.6.x, 3.x de MongoDB y posteriores.
- Versiones de Db2 LUW:
  - Versión 9.7, se admiten todos los Fix Packs.
  - Versión 10.1, se admiten todos los Fix Packs.
  - Version 10.5, se admiten todos los Fix Packs excepto Fix Pack 5.

### Microsoft Azure

- AWS DMS es compatible con cargas completas de datos cuando usa una base de datos SQL Azure como origen. No se admite la captura de datos de cambios (CDC).

## Base de datos de instancia Amazon RDS

- Versiones de Oracle 11g (versiones 11.2.0.3.v1 y posteriores) y 12c para las ediciones Enterprise, Standard, Standard One y Standard Two.
- Versiones de Microsoft SQL Server 2008R2, 2012, 2014 y 2016 para las ediciones Enterprise y Standard. CDC es compatible con todas las versiones de Enterprise Edition. CDC solo se admite para la versión Standard Edition 2016 SP1 y posteriores. Las ediciones Web, Workgroup, Developer y Express no son compatibles con AWS DMS.
- Versiones de MySQL 5.5, 5.6 y 5.7. La captura de datos de cambios (CDC) es solo compatible con las versiones 5.6 y posteriores.
- PostgreSQL 9.4 y posteriores. CDC solo es compatible con las versiones 9.4.9 y posteriores y 9.5.4 y posteriores. El parámetro `rds.logical_replication`, necesario para CDC, se admite únicamente en estas versiones y las posteriores.
- MariaDB se admite como origen de datos compatible con MySQL.
- Compatibilidad de Amazon Aurora con MySQL.
- Compatibilidad de Amazon Aurora con PostgreSQL.

## Amazon S3

- AWS DMS admite la carga completa de datos y la captura de datos de cambios (CDC) cuando utiliza Amazon S3 como origen.

## Temas

- [Uso de una base de datos de Oracle como origen para AWS DMS \(p. 91\)](#)
- [Uso de una base de datos de Microsoft SQL Server como origen para AWS DMS \(p. 108\)](#)
- [Uso de Microsoft Azure SQL Database como origen para AWS DMS \(p. 119\)](#)
- [Uso de una base de datos de PostgreSQL como origen para AWS DMS \(p. 119\)](#)
- [Uso de una base de datos compatible con MySQL como origen para AWS DMS \(p. 132\)](#)
- [Uso de una base de datos SAP ASE como origen para AWS DMS \(p. 140\)](#)
- [Uso de MongoDB como origen para AWS DMS \(p. 143\)](#)
- [Uso de Amazon S3 como origen para AWS DMS \(p. 148\)](#)
- [Uso de una base de datos IBM Db2 para Linux, Unix y Windows \(Db2 LUW\) como origen de AWS DMS \(p. 154\)](#)

# Uso de una base de datos de Oracle como origen para AWS DMS

Puede migrar datos desde una o varias bases de datos de Oracle con AWS DMS. Con una base de datos de Oracle como origen, podrá migrar datos a cualquiera de los destinos compatibles con AWS DMS.

Para las bases de datos de Oracle autoadministradas, AWS DMS es compatible con todas las ediciones de base de datos de Oracle para las versiones 10.2 y posteriores, 11g y hasta 12.2 para bases de datos autoadministradas como orígenes. Para las bases de datos de Oracle administrada por Amazon proporcionadas por Amazon RDS, AWS DMS es compatible con todas las ediciones de base de datos de Oracle para las versiones 11g (versiones 11.2.0.3.v1 y posteriores) y hasta 12.2.

Puede utilizar SSL para cifrar las conexiones entre el punto de enlace de Oracle y la instancia de replicación. Para obtener más información acerca de cómo usar SSL con un punto de enlace de Oracle, consulte [Uso de SSL con AWS Database Migration Service \(p. 51\)](#).

Los pasos para configurar una base de datos de Oracle como origen para origen de AWS DMS son los siguientes:

1. Si desea crear una tarea solo de CDC o de carga completa más CDC, entonces debe elegir o bien Oracle LogMiner u Oracle Binary Reader para capturar los cambios de datos. Elegir LogMiner o Binary Reader determina algunos de los permisos posteriores y tareas de configuración. Si desea obtener una comparación de LogMiner y Binary Reader, consulte la siguiente sección.
2. Cree un usuario de Oracle con los permisos adecuados para AWS DMS. Si está creando una tarea de carga completa solamente, no se necesita ninguna configuración adicional.
3. Cree un punto de enlace de DMS que se ajuste a la configuración elegida.

Para obtener información adicional acerca de cómo trabajar con las bases de datos de Oracle y AWS DMS, consulte las secciones siguientes.

#### Temas

- [Usar Oracle LogMiner u Oracle Binary Reader para la captura de datos de cambios \(CDC\) \(p. 92\)](#)
- [Uso de una base de datos de Oracle autoadministrada como origen para AWS DMS \(p. 94\)](#)
- [Uso de una base de datos de Oracle administrada por Amazon como origen para AWS DMS \(p. 97\)](#)
- [Restricciones en el uso de Oracle como origen para AWS DMS \(p. 99\)](#)
- [Atributos de conexión adicionales al usar Oracle como origen para AWS DMS \(p. 100\)](#)
- [Tipos de datos de origen para Oracle \(p. 105\)](#)

## Usar Oracle LogMiner u Oracle Binary Reader para la captura de datos de cambios (CDC)

Oracle ofrece dos métodos para leer los logs REDO al realizar el procesamiento de cambios: Oracle LogMiner y Oracle Binary Reader. Oracle LogMiner proporciona una interfaz de SQL a los logs REDO online y archivados de Oracle. Binary Reader es una característica de AWS DMS que lee y analiza directamente los logs REDO sin procesar.

De forma predeterminada, AWS DMS utiliza Oracle LogMiner para la captura de datos de cambios (CDC). Entre los beneficios de utilizar LogMiner con AWS DMS se incluyen los siguientes:

- LogMiner es compatible con la mayoría de las opciones de Oracle, como, por ejemplo, las opciones de cifrado y de compresión. Binary Reader no admite todas las opciones de Oracle, especialmente las opciones de cifrado y compresión.
- LogMiner es más sencillo de configurar, sobre todo en comparación con configurar el acceso directo de Oracle Binary Reader o cuando los registros REDO están en Automatic Storage Management (ASM).
- LogMiner es totalmente compatible con la mayoría de las opciones de cifrado de Oracle, incluido el cifrado de datos transparente (TDE) de Oracle.
- LogMiner admite los siguientes tipos de compresión HCC para la carga completa y la replicación en curso (CDC):
  - QUERY HIGH
  - ARCHIVE HIGH
  - ARCHIVE LOW
  - QUERY LOW

Binary Reader admite la compresión QUERY LOW solo para replicaciones de carga completa, no para las replicaciones en curso (CDC).

- LogMiner admite clústeres de tabla para su uso por parte de AWS DMS. Binary Reader no.

Entre los beneficios de utilizar Binary Reader con AWS DMS en lugar de LogMiner se incluyen los siguientes:

- En migraciones con gran cantidad de cambios, LogMiner podría tener cierta repercusión en las E/S o la CPU del equipo que hospeda la base de datos de origen de Oracle. Binary Reader presenta menos probabilidades de afectar a las E/S o la CPU porque los registros de archivo se copian en la instancia de replicación y se analizan allí.
- En migraciones con gran cantidad de cambios, el desempeño de la CDC suele ser mucho mejor cuando se utiliza Binary Reader en comparación con Oracle LogMiner.
- Binary Reader admite CDC para LOB en la versión 12c de Oracle. Por el contrario, LogMiner no lo admite.
- Binary Reader admite los siguientes tipos de compresión HCC para la carga completa y la replicación continua (CDC):
  - QUERY HIGH
  - ARCHIVE HIGH
  - ARCHIVE LOW

El tipo de compresión QUERY LOW solo es compatible con las migraciones de carga completa.

En general, utilice Oracle LogMiner para migrar bases de datos de Oracle, a menos que su caso sea uno de los siguientes:

- Necesita ejecutar varias tareas de migración en la base de datos de origen de Oracle.
- El volumen de los cambios o el volumen de los logs REDO de la base de datos de Oracle de origen es grande.
- Va a migrar LOB desde Oracle 12.2 o desde un punto de enlace de origen posterior.
- La carga de trabajo incluye instrucciones UPDATE que solo actualizan las columnas de LOB. En este caso, utilice Binary Reader. Estas instrucciones UPDATE no son compatibles con Oracle LogMiner.
- El origen es la versión 11 de Oracle y realiza instrucciones UPDATE en XMLTYPE y columnas de LOB. En este caso, debe utilizar Binary Reader. Estas instrucciones no son compatibles con Oracle LogMiner.
- Está migrando columnas de LOB desde Oracle 12c. Para Oracle 12c, LogMiner no admite las columnas de LOB, por lo que en este caso debe utilizar Binary Reader.

## Configuración para la captura de datos de cambios (CDC) en una base de datos de origen de Oracle

Cuando se utiliza Oracle como punto de enlace de origen ya sea para carga completa y la captura de datos de cambios (CDC) o simplemente para CDC, debe configurar un atributo de conexión adicional. Este atributo especifica si se debe usar LogMiner o Binary Reader para obtener acceso a los logs de transacciones. Debe especificar un atributo de conexión adicional al crear el punto de enlace de origen. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

LogMiner se utiliza de forma predeterminada, por lo que no tiene que especificar explícitamente su uso. Para habilitar Binary Reader para obtener acceso a los registros de transacciones, añada los siguientes atributos de conexión adicional.

```
useLogMinerReader=N;useBfile=Y;
```

Si la base de datos de origen de Oracle utiliza Oracle Automatic Storage Management (ASM), el atributo de conexión adicional debe incluir el nombre de usuario de ASM y la dirección del servidor ASM. Cuando

crea el punto de enlace de origen, el campo de la contraseña también tiene que tener ambas contraseñas: la contraseña del usuario de origen y la contraseña de ASM.

Por ejemplo, el siguiente formato de atributo de conexión adicional se utiliza para obtener acceso a un servidor que utiliza Oracle ASM.

```
useLogMinerReader=N;asm_user=<asm_username>;asm_server=<first_RAC_server_ip_address>:<port_number>/  
+ASM
```

Si la base de datos de origen de Oracle utiliza Oracle ASM, el campo de la contraseña del punto de enlace de origen tiene que tener tanto la contraseña de usuario de Oracle como la contraseña de ASM, separadas ambas por una coma. Por ejemplo, lo siguiente funciona en el campo de contraseña.

```
<oracle_user_password>,<asm_user_password>
```

## Restricciones para la CDC en una base de datos de origen de Oracle

Las siguientes restricciones se aplican cuando se utiliza una base de datos de Oracle como origen para la captura de datos de cambios de AWS DMS:

- AWS DMS no captura los cambios que realiza el paquete DBMS\_REDEFINITION de Oracle, como los cambios en los metadatos de la tabla y el valor OBJECT\_ID.
- AWS DMS no admite tablas organizadas por índice con un segmento de desbordamiento en el modo CDC cuando se utiliza BFILE. Un ejemplo es cuando se accede a los logs REDO sin usar LogMiner.

## Uso de una base de datos de Oracle autoadministrada como origen para AWS DMS

Una base de datos autoadministrada es una base de datos que configura y controla, ya sea una instancia de base de datos local o una base de datos en Amazon EC2. A continuación, puede obtener información acerca de los privilegios y configuraciones que tiene que configurar cuando utiliza una base de datos de Oracle autoadministrada con AWS DMS.

### Privilegios de cuenta de usuario necesarios en un origen de Oracle autoadministrado para AWS DMS

Para utilizar una base de datos de Oracle como origen en una tarea AWS DMS, se deben conceder al usuario especificado en las definiciones de la base de datos de Oracle para AWS DMS los siguientes privilegios en la base de datos Oracle. Cuando conceda privilegios, utilice el nombre real de los objetos, no el sinónimo del objeto. Por ejemplo, utilice V\_\$OBJECT incluido el guion bajo, no V\$OBJECT sin el guion bajo.

```
GRANT SELECT ANY TRANSACTION to <dms_user>  
GRANT SELECT on V_$ARCHIVED_LOG to <dms_user>  
GRANT SELECT on V_$LOG to <dms_user>  
GRANT SELECT on V_$LOGFILE to <dms_user>  
GRANT SELECT on V_$DATABASE to <dms_user>  
GRANT SELECT on V_$THREAD to <dms_user>  
GRANT SELECT on V_$PARAMETER to <dms_user>  
GRANT SELECT on V_$NLS_PARAMETERS to <dms_user>  
GRANT SELECT on V_$TIMEZONE_NAMES to <dms_user>  
GRANT SELECT on V_$TRANSACTION to <dms_user>  
GRANT SELECT on ALL_INDEXES to <dms_user>  
GRANT SELECT on ALL_OBJECTS to <dms_user>
```

```
GRANT SELECT on DBA_OBJECTS to <dms_user> (required if the Oracle version is earlier than  
11.2.0.3)  
GRANT SELECT on ALL_TABLES to <dms_user>  
GRANT SELECT on ALL_USERS to <dms_user>  
GRANT SELECT on ALL_CATALOG to <dms_user>  
GRANT SELECT on ALL_CONSTRAINTS to <dms_user>  
GRANT SELECT on ALL_CONS_COLUMNS to <dms_user>  
GRANT SELECT on ALL_TAB_COLS to <dms_user>  
GRANT SELECT on ALL_IND_COLUMNS to <dms_user>  
GRANT SELECT on ALL_LOG_GROUPS to <dms_user>  
GRANT SELECT on SYS.DBA_REGISTRY to <dms_user>  
GRANT SELECT on SYS.OBJ$ to <dms_user>  
GRANT SELECT on DBA_TABLESPACES to <dms_user>  
GRANT SELECT on ALL_TAB_PARTITIONS to <dms_user>  
GRANT SELECT on ALL_ENCRYPTED_COLUMNS to <dms_user>  
GRANT SELECT on V_$LOGMNR_LOGS to <dms_user>  
GRANT SELECT on V_$LOGMNR_CONTENTS to <dms_user>
```

Cuando se utiliza la replicación continua (CDC), son necesarios estos permisos adicionales.

- El siguiente permiso es necesario cuando se utiliza CDC para que AWS DMS pueda añadir registros REDO a Oracle LogMiner para las versiones 11g y 12c.

```
Grant EXECUTE ON dbms_logmnr TO <dms_user>;
```

- El siguiente permiso es necesario cuando se utiliza CDC para que AWS DMS pueda añadir registros REDO a Oracle LogMiner solo para la versión 12c.

```
Grant LOGMINING TO <dms_user>;
```

Si utiliza las características adicionales que se indican a continuación, se necesitan los permisos adicionales especificados:

- Si hay vistas, conceda SELECT en ALL\_VIEWS a <usuario\_dms>.
- Si utiliza un patrón para que coincida con los nombres de las tablas en su tarea de replicación, conceda SELECT ANY TABLE.
- Si especifica una lista de tablas en su tarea de replicación, conceda SELECT on cada tabla de la lista.
- Si añade un registro suplementario, conceda ALTER ANY TABLE.
- Si añade un registro suplementario y utiliza una lista de tabla específica, conceda ALTER para cada tabla de la lista.
- Si migra desde Oracle RAC, conceda los permisos SELECT para vistas materializadas con los prefijos g\_\$ y v\_\$.

## Configuración de un origen de Oracle autoadministrado para AWS DMS

Antes de utilizar una base de datos de Oracle autoadministrada como un origen para AWS DMS, debe realizar varias tareas:

- Proporcionar acceso a la cuenta de Oracle: debe proporcionar una cuenta de usuario de Oracle para AWS DMS. La cuenta de usuario debe disponer de privilegios de lectura y escritura en la base de datos de Oracle, según se especifica en la sección anterior.

- Asegúrese de que el modo ARCHIVELOG está activado: Oracle puede ejecutarse en dos modos diferentes, el modo ARCHIVELOG y el modo NOARCHIVELOG. Para utilizar Oracle con AWS DMS, la base de datos de origen debe encontrarse en modo ARCHIVELOG.
- Configure el registro suplementario: si pretende utilizar el origen en una tarea de CDC o en una tarea de carga completa más CDC, tiene que configurar un registro suplementario para capturar los cambios para la replicación.

Hay dos pasos para habilitar el registro suplementario para Oracle. En primer lugar, tiene que habilitar el registro suplementario en el nivel de base de datos. Esto garantiza que LogMiner tenga la información mínima para admitir diferentes estructuras de tabla como tablas organizadas por índice o agrupadas. En segundo lugar, debe habilitar el registro suplementario en el nivel de tabla para cada tabla que se va a migrar.

Para habilitar el registro suplementario en el nivel de base de datos

1. Ejecute la siguiente consulta para determinar si el registro suplementario en el nivel de la base de datos ya está habilitado. El resultado devuelto debe ser desde GE to 9.0.0.

```
SELECT name, value, description FROM v$parameter WHERE name = 'compatible';
```

2. Ejecute la siguiente consulta. El resultado devuelto debe ser YES o IMPLICIT.

```
SELECT supplemental_log_data_min FROM v$database;
```

3. Ejecute la siguiente consulta para habilitar el registro suplementario en el nivel de la base de datos.

```
ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;
```

Hay dos métodos para habilitar el registro suplementario en el nivel de tabla. En el primero, si su cuenta de usuario de base de datos tiene privilegios ALTER TABLE en todas las tablas que se van a migrar, puede utilizar el parámetro de conexión adicional addSupplementalLogging tal y como se describe a continuación. De lo contrario, puede utilizar los pasos siguientes para cada tabla en la migración.

Para habilitar el registro suplementario en el nivel de la tabla

1. Si la tabla tiene una clave principal, añada el registro suplementario PRIMARY KEY para la tabla ejecutando el siguiente comando.

```
ALTER TABLE <table_name> ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;
```

2. Si no existe una clave primaria y la tabla tiene varios índices únicos, entonces AWS DMS utiliza el primer índice único en orden alfabético del nombre del índice.

Cree un grupo de registros suplementario como se muestra en la sección anterior en las columnas del índice.

3. Si no hay una clave principal ni un índice único, el registro suplementario deberá agregarse en todas las columnas. Ejecute la siguiente consulta para añadir registros suplementarios a todas las columnas.

```
ALTER TABLE <table_name> ADD SUPPLEMENTAL LOG DATA (ALL) COLUMNS;
```

En algunos casos, el índice único o la clave primaria de la tabla de destino son diferentes del índice único o la clave primaria de la tabla de origen. En estos casos, añada el registro suplementario en las columnas de la tabla de origen que componen el índice único o la clave primaria de la tabla de destino. Si cambia

la clave primaria de la tabla de destino, añade el registro complementario en las columnas del índice seleccionadas. No lo añade en las columnas de la clave primaria o del índice único originales.

Añada registros suplementarios si fuera necesario, como por ejemplo, si se define un filtro para una tabla. Si una tabla contiene un índice único o una clave principal, añada registros complementarios en cada columna con filtros si esas columnas son diferentes de las columnas de la clave principal o el índice único. Sin embargo, si se ha añadido a la tabla el registro suplementario ALL COLUMNS, no es necesario añadir registros suplementarios.

```
ALTER TABLE <table_name> ADD SUPPLEMENTAL LOG GROUP <group_name> (<column_list>) ALWAYS;
```

## Uso de una base de datos de Oracle administrada por Amazon como origen para AWS DMS

Una base de datos administrada por Amazon es aquella que se encuentra en un servicio de Amazon como Amazon RDS, Amazon Aurora o Amazon S3. A continuación, puede obtener información acerca de los privilegios y configuraciones que tiene que definir cuando utiliza una base de datos de Oracle administrada por Amazon con AWS DMS.

### Privilegios de cuenta de usuario necesarios en un origen de Oracle administrado por Amazon para AWS DMS

Para conceder privilegios de bases de datos de Oracle en Amazon RDS, utilice el procedimiento almacenado `rdsadmin.rdsadmin_util.grant_sys_object`. Para obtener más información, consulte la sección que explica cómo [otorgar privilegios SELECT o EXECUTE para objetos SYS](#).

Conceda los siguientes privilegios a la cuenta de usuario de AWS DMS que se utiliza para obtener acceso al punto de enlace de Oracle origen:

```
GRANT SELECT ANY TABLE TO <dms_user>;
GRANT SELECT ON ALL_VIEWS TO <dms_user>;
GRANT SELECT ANY TRANSACTION TO <dms_user>;
GRANT SELECT ON DBA_TABLESPACES TO <dms_user>;
GRANT SELECT ON ALL_TAB_PARTITIONS TO <dms_user>;
GRANT SELECT ON ALL_INDEXES TO <dms_user>;
GRANT SELECT ON ALL_OBJECTS TO <dms_user>;
GRANT SELECT ON ALL_TABLES TO <dms_user>;
GRANT SELECT ON ALL_USERS TO <dms_user>;
GRANT SELECT ON ALL_CATALOG TO <dms_user>;
GRANT SELECT ON ALL_CONSTRAINTS TO <dms_user>;
GRANT SELECT ON ALL_CONS_COLUMNS TO <dms_user>;
GRANT SELECT ON ALL_TAB_COLS TO <dms_user>;
GRANT SELECT ON ALL_IND_COLUMNS TO <dms_user>;
GRANT SELECT ON ALL_LOG_GROUPS TO <dms_user>;
GRANT LOGMINING TO <dms_user>;
```

Además, ejecute lo siguiente:

```
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$ARCHIVED_LOG','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$LOG','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$LOGFILE','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$DATABASE','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$THREAD','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$PARAMETER','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$NLS_PARAMETERS','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$TIMEZONE_NAMES','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$TRANSACTION','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_REGISTRY','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('OBJ$','<dms_user>','SELECT');
```

```
exec
rdsadmin.rdsadmin_util.grant_sys_object('ALL_ENCRYPTED_COLUMNS','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$LOGMNR_LOGS','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$LOGMNR_CONTENTS','<dms_user>','SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_LOGMNR','<dms_user>','EXECUTE');
```

## Configuración de una fuente de Oracle administrada por Amazon para AWS DMS

Antes de utilizar una base de datos de Oracle administrada por Amazon como un origen para AWS DMS, debe realizar varias tareas:

- Proporcionar acceso a la cuenta de Oracle: debe proporcionar una cuenta de usuario de Oracle para AWS DMS. La cuenta de usuario debe disponer de privilegios de lectura y escritura en la base de datos de Oracle, según se especifica en la sección anterior.
- Establezca el período de retención de copia de seguridad de la base de datos de Amazon RDS en un día o más: la configuración del periodo de retención de copia de seguridad garantiza que la base de datos se ejecuta en el modo ARCHIVELOG. Para obtener más información acerca de la configuración del periodo de retención de copia de seguridad, consulte [Trabajo con copias de seguridad automatizadas](#) en la Guía del usuario de Amazon RDS.
- Configuración de retención de archivos: ejecute lo siguiente para conservar los registros REDO archivados de la instancia de base de datos de Oracle. La ejecución de este comando permite a AWS DMS recuperar la información del registro con LogMiner. Asegúrese de que cuenta con suficiente almacenamiento para los logs REDO archivados durante el periodo de migración.

En el siguiente ejemplo, los registros se conservan durante 24 horas.

```
exec rdsadmin.rdsadmin_util.set_configuration('archivelog retention hours',24);
```

- Configure el registro complementario: si pretende utilizar el origen en una tarea de CDC o en una tarea de carga completa más CDC, configure un registro complementario para capturar los cambios para la replicación.

Hay dos pasos para habilitar el registro suplementario para Oracle. En primer lugar, tiene que habilitar el registro suplementario en el nivel de base de datos. Esto garantiza que LogMiner tenga la información mínima para admitir diferentes estructuras de tabla como tablas organizadas por índice o agrupadas. En segundo lugar, debe habilitar el registro suplementario en el nivel de tabla para cada tabla que se va a migrar.

Para habilitar el registro suplementario en el nivel de base de datos

- Ejecute la siguiente consulta para habilitar el registro suplementario en el nivel de la base de datos.

```
exec rdsadmin.rdsadmin_util.alter_supplemental_logging('ADD');
```

Para habilitar el registro suplementario en el nivel de la tabla

- Ejecute el siguiente comando para habilitar el registro de PRIMARY KEY para tablas que tengan claves primarias.

```
exec rdsadmin.rdsadmin_util.alter_supplemental_logging('ADD','PRIMARY KEY');
```

Para tablas que no tienen claves primarias, utilice el siguiente comando para añadir el registro suplementario.

```
alter table <table_name> add supplemental log data (ALL) columns;
```

Si crea una tabla sin clave primaria, deberá incluir una cláusula de registro suplementario en la instrucción de creación o modificar la tabla para añadir el registro suplementario. El siguiente comando crea una tabla y añade el registro adicional.

```
create table <table_name> (<column_list>, supplemental log data (ALL) columns);
```

Si crea una tabla y después añade una clave principal, debe añadir el registro suplementario a la tabla. Añada el registro suplementario a la tabla mediante el siguiente comando.

```
alter table <table_name> add supplemental log data (PRIMARY KEY) columns;
```

## Configuración de la captura de datos de cambio (CDC) para un origen de Amazon RDS para Oracle para AWS DMS

Puede configurar AWS DMS para utilizar una instancia de Amazon RDS for Oracle como origen de CDC. Puede utilizar Oracle Binary Reader con un origen de Amazon RDS para Oracle (versiones de Oracle 11.2.0.4.v11 y posteriores así como 12.1.0.2.v7 y posteriores).

Incluya los siguientes atributos de conexión adicionales cuando cree el punto de enlace de origen de Amazon RDS for Oracle.

```
useLogminerReader=N;useBfile=Y;accessAlternateDirectly=false;  
useAlternateFolderForOnline=true;oraclePathPrefix=/rdsdbdata/db/ORCL_A/;  
usePathPrefix=/rdsdbdata/log/;replacePathPrefix=true
```

### Note

No debe haber espacios en blanco tras el separador de punto y coma (;) para varios valores de atributo.

## Restricciones en el uso de Oracle como origen para AWS DMS

Se aplican las siguientes restricciones cuando se utiliza una base de datos de Oracle como origen para AWS DMS:

- AWS DMS admite el cifrado de espacios de tablas de cifrado transparente de datos (TDE) de Oracle y el cifrado AWS Key Management Service (AWS KMS) cuando se utiliza con Oracle LogMiner. El resto de métodos de cifrado no son compatibles.
- Las tablas con LOB deben tener una clave principal para utilizar la CDC.
- AWS DMS admite la sintaxis `rename table <table name> to <new table name>` con la versión 11 de Oracle y versiones posteriores.
- AWS DMS no replica los cambios en los datos que se obtienen a partir de operaciones—de partición o subpartición: operaciones de lenguaje de definición de datos (DDL) como ADD, DROP, EXCHANGE o TRUNCATE. Para replicar estos cambios, debe cargar la tabla que se replica. AWS DMS replica cualquier cambio futuro en los datos en particiones añadidas recientemente sin necesidad de volver a cargar la tabla. Sin embargo, las operaciones UPDATE en registros antiguos de datos en particiones producen un error y generan una advertencia `0 rows affected`.
- La instrucción DDL `ALTER TABLE ADD <column> <data_type> DEFAULT <>` no replica el valor predeterminado en el destino. La nueva columna en el destino se establece en NULL. Si la nueva

columna se puede anular, Oracle actualiza todas las filas de la tabla antes de registrar el DDL. En consecuencia, AWS DMS captura los cambios en los contadores, pero no actualiza el destino. Dado que la nueva columna se establece en NULL, si la tabla de destino no tiene una clave principal o índice único, las actualizaciones posteriores generan una advertencia `0 rows affected`.

- AWS DMS admite todos los cambios de datos en el destino que se producen como resultado de ejecutar la instrucción `CREATE TABLE AS` en el origen.

#### Note

Todos los cambios de datos en el destino que se producen como resultado de ejecutar la instrucción `CREATE TABLE AS` en el origen se admiten en las versiones 3.1.3 y posteriores de AWS DMS. En las versiones anteriores, `CREATE TABLE AS` crea una nueva tabla en el destino, pero los datos no se migran a ella.

- Cuando está habilitado el modo de tamaño limitado de LOB, AWS DMS replica los LOB vacíos en el origen de Oracle como valores NULL en el destino.
- Cuando AWS DMS inicia la CDC, asigna una marca de tiempo al número de cambio del sistema (SCN) de Oracle. De forma predeterminada, Oracle conserva tan solo cinco días de la marca de tiempo para las asignaciones de SCN. Oracle genera un error si la marca temporal especificada es demasiado antigua (superior al periodo de conservación de cinco días). Para obtener más información, consulte la [documentación de Oracle](#).
- AWS DMS no es compatible con las conexiones a un origen de Oracle mediante un proxy de ASM.
- AWS DMS no admite columnas virtuales.
- En un entorno con múltiples inquilinos, AWS DMS puede conectarse a una base de datos conectable (PDB) en un contenedor (CDB). En esta configuración, debe utilizar Binary Reader para la captura de datos de cambios (CDC). No se admite LogMiner para PDB.

## Atributos de conexión adicionales al usar Oracle como origen para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar el origen de Oracle. Especifique esta configuración al crear el punto de enlace de origen. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

En la siguiente tabla se muestran los atributos de conexión adicionales que puede utilizar para configurar una base de datos de Oracle como origen para AWS DMS.

Nombre	Descripción
<code>addSupplementalLogging</code>	<p>Establezca este atributo para configurar un registro suplementario para la base de datos de Oracle. Este atributo PRIMARY KEY permite el registro suplementario en todas las tablas seleccionadas para una tarea de migración.</p> <p>Valor predeterminado: N</p> <p>Valores válidos: Y/N</p> <p>Ejemplo: <code>addSupplementalLogging=Y;</code></p> <p><b>Note</b></p> <p>Si utiliza esta opción, tiene que habilitar igualmente el registro suplementario en el nivel de la base de datos tal y como hemos mencionado con anterioridad.</p>

Nombre	Descripción
<code>additionalArchivedLogDestId</code>	Establezca este atributo con <code>archivedLogDestId</code> en una configuración principal o de standby. Este atributo es útil en el caso de una comutación por error. En este caso, AWS DMS necesita saber qué destino para obtener los registros REDO de archivado desde donde leer los cambios. Esta necesidad surge porque la instancia principal anterior es ahora una instancia en espera después de una comutación por error.
<code>useLogminerReader</code>	<p>Establezca este atributo en Y para la captura de datos de cambios mediante la utilidad LogMiner (la predeterminada). Establezca esta opción en N si desea que AWS DMS obtenga acceso a los logs REDO como un archivo binario. Cuando especifique esta opción en N, añada además el valor de configuración <code>useBfile = Y</code>. Para obtener más información, consulte <a href="#">Usar Oracle LogMiner u Oracle Binary Reader para la captura de datos de cambios (CDC) (p. 92)</a>.</p> <p>Valor predeterminado: Y</p> <p>Valores válidos: Y/N</p> <p>Ejemplo: <code>useLogminerReader=N;useBfile=Y;</code></p> <p>Si la base de datos de origen de Oracle utiliza Oracle Automatic Storage Management (ASM), el parámetro de conexión adicional debe incluir el nombre de usuario de ASM y la dirección del servidor ASM. El campo de la contraseña también debe tener ambas contraseñas: la contraseña del usuario de origen y la contraseña de ASM, separadas entre sí por una coma.</p> <p>Ejemplo:  <code>useLogminerReader=N;asm_user=&lt;asm_username&gt;;asm_server=&lt;f+ASM;</code></p>
<code>useBfile</code>	<p>Establezca este atributo en Y para capturar los datos de cambios mediante la utilidad Binary Reader. Establezca <code>useLogminerReader</code> en N para definir este atributo como Y. Para utilizar Binary Reader con Amazon RDS for Oracle como origen, establezca atributos adicionales. Para obtener más información, consulte <a href="#">Usar Oracle LogMiner u Oracle Binary Reader para la captura de datos de cambios (CDC) (p. 92)</a>.</p> <p>Valor predeterminado: N</p> <p>Valores válidos: Y/N</p> <p>Ejemplo: <code>useLogminerReader=N;useBfile=Y;</code></p>

Nombre	Descripción
accessAlternateDirectly	<p>Establezca este atributo en false para utilizar Binary Reader para capturar los datos de cambios de una instancia de Amazon RDS for Oracle como origen. Esto indica a la instancia de DMS que no obtenga acceso a los registros REDO a través de ninguno de los prefijos de ruta sustitutos especificados mediante el acceso directo a los archivos.</p> <p>Para obtener más información, consulte <a href="#">Configuración de la captura de datos de cambio (CDC) para un origen de Amazon RDS para Oracle para AWS DMS</a> (p. 99).</p> <p>Valor predeterminado: verdadero</p> <p>Valores válidos: true/false</p> <p>Ejemplo:</p> <pre>useLogminerReader=N;useBfile=Y;accessAlternateDirectly=false;</pre>
useAlternateFolderForOnline	<p>Establezca este atributo en true para utilizar Binary Reader para capturar los datos de cambios de una instancia de Amazon RDS for Oracle como origen. Esto indica a la instancia de DMS que use cualquier prefijo sustituto especificado para obtener acceso a todos los registros REDO online. Para obtener más información, consulte <a href="#">Configuración de la captura de datos de cambio (CDC) para un origen de Amazon RDS para Oracle para AWS DMS</a> (p. 99).</p> <p>Valor predeterminado: false</p> <p>Valores válidos: true/false</p> <p>Ejemplo:</p> <pre>useLogminerReader=N;useBfile=Y;accessAlternateDirectly=false; useAlternateFolderForOnline=true;</pre>
oraclePathPrefix	<p>Establezca este atributo de cadena en el valor necesario para utilizar Binary Reader para capturar los datos de cambios de una instancia de Amazon RDS for Oracle como origen. Este valor especifica la raíz de Oracle predeterminada usada para obtener acceso a los registros REDO. Para obtener más información, consulte <a href="#">Configuración de la captura de datos de cambio (CDC) para un origen de Amazon RDS para Oracle para AWS DMS</a> (p. 99).</p> <p>Valor predeterminado: ninguno</p> <p>Valor válido: /rdsdbdata/db/ORCL_A/</p> <p>Ejemplo:</p> <pre>useLogminerReader=N;useBfile=Y;accessAlternateDirectly=false; useAlternateFolderForOnline=true;oraclePathPrefix=/rdsdbdata/db/ORCL_A/;</pre>

Nombre	Descripción
<code>usePathPrefix</code>	<p>Establezca este atributo de cadena en el valor necesario para utilizar Binary Reader para capturar los datos de cambios de una instancia de Amazon RDS for Oracle como origen. Este valor especifica el prefijo de ruta utilizado para reemplazar la raíz de Oracle predeterminada empleada para obtener acceso a los registros REDO. Para obtener más información, consulte <a href="#">Configuración de la captura de datos de cambio (CDC) para un origen de Amazon RDS para Oracle para AWS DMS</a> (p. 99).</p> <p>Valor predeterminado: ninguno</p> <p>Valor válido: /rdsdbdata/log/</p> <p>Ejemplo:</p> <pre>useLogminerReader=N;useBfile=Y;accessAlternateDirectly=false; useAlternateFolderForOnline=true;oraclePathPrefix=/rdsdbdata/db/ORCL_A/; usePathPrefix=/rdsdbdata/log/;</pre>
<code>replacePathPrefix</code>	<p>Establezca este atributo en true para utilizar Binary Reader para capturar los datos de cambios de una instancia de Amazon RDS for Oracle como origen. Este valor indica a la instancia de DMS que reemplace la raíz de Oracle predeterminada por el valor de <code>usePathPrefix</code> especificado para obtener acceso a los registros REDO. Para obtener más información, consulte <a href="#">Configuración de la captura de datos de cambio (CDC) para un origen de Amazon RDS para Oracle para AWS DMS</a> (p. 99).</p> <p>Valor predeterminado: false</p> <p>Valores válidos: true/false</p> <p>Ejemplo:</p> <pre>useLogminerReader=N;useBfile=Y;accessAlternateDirectly=false; useAlternateFolderForOnline=true;oraclePathPrefix=/rdsdbdata/db/ORCL_A/; usePathPrefix=/rdsdbdata/log/;replacePathPrefix=true;</pre>
<code>retryInterval</code>	<p>Especifica el número de segundos que espera el sistema antes de reenviar una consulta.</p> <p>Valor predeterminado: 5</p> <p>Valores válidos: números a partir de 1</p> <p>Ejemplo: <code>retryInterval=6</code>;</p>

Nombre	Descripción
archivedLogDestId	<p>Especifica el destino de los logs REDO archivados. El valor debe ser el mismo que el número de DEST_ID en la tabla \$archived_log. Al trabajar con varios destinos de registro (DEST_ID), recomendamos que especifique un identificador de ubicación para los logs REDO archivados. De esta forma se mejora el desempeño garantizando que se obtiene acceso a los registros correctos desde el principio.</p> <p>Valor predeterminado: 0</p> <p>Valores válidos: Number</p> <p>Ejemplo: archivedLogDestId=1;</p>
archivedLogsOnly	<p>Cuando este campo se establece en Y, AWS DMS solo obtiene acceso a los logs REDO archivados. Si los logs REDO archivados se almacenan solo en Oracle ASM, habrá que conceder a la cuenta de usuario de AWS DMS privilegios de ASM.</p> <p>Valor predeterminado: N</p> <p>Valores válidos: Y/N</p> <p>Ejemplo: archivedLogsOnly=Y;</p>
numberDataTypeScale	<p>Especifica la escala de números. Puede seleccionar una escala de hasta 38 o puede seleccionar FLOAT. De forma predeterminada, el tipo de datos NUMBER se convierte a precisión 38, escala 10.</p> <p>Valor predeterminado: 10</p> <p>Valores válidos: -1 a 38 (-1 para FLOAT)</p> <p>Ejemplo: numberDataTypeScale=12</p>
afterConnectScript	<p>Especifica un script que se ejecuta inmediatamente después de que AWS DMS se conecta al punto de enlace.</p> <p>Valores válidos: una instrucción SQL separada por un punto y coma. No se admiten todas las instrucciones SQL.</p> <p>Ejemplo: afterConnectScript=ALTER SESSION SET CURRENT_SCHEMA = system;</p>

Nombre	Descripción
<code>failTasksOnLobTruncation</code>	<p>Cuando se establece en <code>true</code>, este atributo provocar un error en una tarea si el tamaño real de una columna de LOB es superior al <code>LobMaxSize</code> especificado.</p> <p>Si una tarea está establecida en el modo LOB limitado y esta opción está establecida en <code>true</code>, la tarea genera un error en vez de truncar los datos de LOB.</p> <p>Valor predeterminado: <code>false</code></p> <p>Valores válidos: booleano</p> <p>Ejemplo: <code>failTasksOnLobTruncation=true;</code></p>
<code>readTableSpaceName</code>	<p>Cuando se establece en <code>true</code>, este atributo admite la replicación del espacio de tabla.</p> <p>Valor predeterminado: <code>false</code></p> <p>Valores válidos: booleano</p> <p>Ejemplo: <code>readTableSpaceName=true;</code></p>
<code>standbyDelayTime</code>	<p>Utilice este atributo para especificar una hora en minutos que indique el retraso en la sincronización de la base de datos standby.</p> <p>Con la versión 2.3.0 y posteriores de AWS DMS, puede crear una tarea de CDC de Oracle que utilice una instancia en espera de Active Data Guard como origen para replicar los cambios en curso. Esto elimina la necesidad de establecer conexión con una base de datos activa que podría estar en la fase de producción.</p> <p>Valor predeterminado: 0</p> <p>Valores válidos: Number</p> <p>Ejemplo: <code>standbyDelayTime=1;</code></p>

## Tipos de datos de origen para Oracle

El punto de enlace de Oracle para AWS DMS es compatible con la mayoría de los tipos de datos de Oracle. La siguiente tabla muestra los tipos de datos de origen de Oracle que se admiten cuando se utiliza AWS DMS y el mapeo predeterminado a los tipos de datos de AWS DMS.

Para obtener más información sobre cómo ver el tipo de datos que se asigna en el destino, consulte la sección del punto de enlace de destino que esté utilizando.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de Oracle	Tipo de datos de AWS DMS
<code>BINARY_FLOAT</code>	<code>REAL4</code>

Tipos de datos de Oracle	Tipo de datos de AWS DMS
BINARY_DOUBLE	REAL8
BINARY	BYTES
FLOAT (P)	Si la precisión es inferior o igual a 24, utilice REAL4. Si la precisión es superior a 24, utilice REAL8.
NUMBER (P,S)	Cuando la escala sea inferior a 0, utilice REAL8
NUMBER se utiliza con arreglo a la propiedad "Expose number as" en la configuración de la base de datos de origen de Oracle.	Cuando la escala sea 0: <ul style="list-style-type: none"><li>• Y la precisión sea 0, utilice REAL8.</li><li>• Y la precisión sea mayor o igual que 2, utilice INT1.</li><li>• Y la precisión sea mayor que 2 e inferior o igual que 4, utilice INT2.</li><li>• Y la precisión sea mayor que 4 e inferior o igual que 9, utilice INT4.</li><li>• Y la precisión sea superior a 9, utilice NUMERIC.</li><li>• Y la precisión sea mayor o igual que la escala, utilice NUMERIC.</li></ul> En el resto de casos, utilice REAL8.
FECHA	DATETIME
INTERVAL_YEAR TO MONTH	STRING (con indicación year_to_month del intervalo)
INTERVAL_DAY TO SECOND	STRING (con indicación day_to_second del intervalo)
TIME	DATETIME
TIMESTAMP	DATETIME
TIMESTAMP WITH TIME ZONE	STRING (con indicación timestamp_with_timezone)
TIMESTAMP CON ZONA HORARIA LOCAL	STRING (con indicación timestamp_with_local_timezone)
CHAR	STRING
VARCHAR2	STRING
NCHAR	WSTRING
NVARCHAR2	WSTRING
RAW	BYTES
REAL	REAL8
BLOB	BLOB  Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos BLOB para una tarea en particular. AWS DMS solo es compatible con tipos de datos BLOB en tablas que incluyan una clave principal.

Tipos de datos de Oracle	Tipo de datos de AWS DMS
CLOB	CLOB  Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos CLOB para una tarea en particular. En la captura de datos de cambios (CDC), AWS DMS admite tipos de datos CLOB solo en tablas que incluyan una clave principal.
NCLOB	NCLOB  Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos NCLOB para una tarea en particular. En la CDC, AWS DMS admite tipos de datos NCLOB solo en las tablas que incluyan una clave principal.
LONG	CLOB  El tipo de datos LONG no se admite en el modo de aplicación optimizada por lotes (modo TurboStream CDC). Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos LOB para una tarea en particular. En la CDC, AWS DMS admite tipos de datos LOB solo en tablas que tengan una clave principal.
LONG RAW	BLOB  El tipo de datos LONG RAW no se admite en el modo de aplicación optimizada por lotes (modo TurboStream CDC). Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos LOB para una tarea en particular. En la CDC, AWS DMS admite tipos de datos LOB solo en tablas que tengan una clave principal.
XMLTYPE	CLOB  Para admitir el tipo de datos XMLTYPE, es necesario disponer del cliente Oracle completo (en lugar de Oracle Instant Client). Cuando la columna de destino es un CLOB, se admite el modo LOB completo y el modo LOB limitado (en función del destino).

No se admiten las tablas de Oracle que se utilizan como origen con columnas de los siguientes tipos de datos y no se pueden replicar. Si se replican las columnas con estos tipos de datos se obtendrán una columna con el valor NULL.

- BFILE
- ROWID
- REF
- UROWID
- Tabla anidada
- Tipos de datos definidos por el usuario
- ANYDATA

#### Note

No se admiten las columnas virtuales.

## Uso de una base de datos de Microsoft SQL Server como origen para AWS DMS

Puede migrar datos desde una o varias bases de datos de Microsoft SQL Server utilizando AWS DMS (AWS DMS). Con una base de datos de SQL Server como origen, podrá migrar datos a otra base de datos de SQL Server o a una de las bases de datos compatibles.

AWS DMS es compatible, como origen, con las bases de datos de instancias locales y Amazon EC2 para las versiones de Microsoft SQL Server 2005, 2008, 2008R2, 2012, 2014, 2016 y 2017. Se admiten las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.

AWS DMS admite, como origen, las bases de datos de instancia de bases de datos de Amazon RDS para las versiones de SQL Server 2008R2, 2012, 2014, 2016 y 2017. Se admiten las ediciones Enterprise y Standard. CDC es compatible con todas las versiones de Enterprise Edition. CDC solo se admite para la versión Standard Edition 2016 SP1 y posteriores. Las ediciones Web, Workgroup, Developer y Express no son compatibles.

Puede instalar la base de datos de origen de SQL Server en cualquier equipo de la red. También necesita una cuenta de SQL Server con los privilegios de acceso adecuados a la base de datos de origen para el tipo de tarea que eligió, para utilizarla con AWS DMS.

AWS DMS admite la migración de datos de instancias nombradas de SQL Server. Puede utilizar las siguientes notaciones en el nombre del servidor al crear el punto de enlace de origen.

IPAddress\InstanceName

Por ejemplo, el siguiente es un nombre de servidor de punto de enlace de origen correcto. Aquí, la primera parte del nombre es la dirección IP del servidor y la segunda parte es el nombre de la instancia de SQL Server (en este ejemplo, SQLTest).

10.0.0.25\SQLTest

Puede utilizar SSL para cifrar las conexiones entre el punto de enlace de SQL Server y la instancia de replicación. Para obtener más información sobre cómo utilizar SSL con un punto de enlace de SQL Server, consulte [Uso de SSL con AWS Database Migration Service \(p. 51\)](#).

Para capturar cambios de una base de datos de origen de SQL Server, esta se debe configurar para copias de seguridad completas y debe ser de la edición Enterprise, Developer o Standard.

Para obtener más información sobre cómo trabajar con las bases de datos de origen de SQL Server y AWS DMS, consulte lo siguiente.

### Temas

- [Restricciones en el uso de SQL Server como origen para AWS DMS \(p. 109\)](#)
- [Uso de la replicación continua \(CDC\) desde un origen de SQL Server \(p. 109\)](#)
- [Métodos de compresión compatibles \(p. 113\)](#)
- [Trabajar con los grupos de disponibilidad AlwaysOn de SQL Server \(p. 114\)](#)
- [Configuración de una base de datos de SQL Server como origen de replicación para AWS DMS \(p. 114\)](#)
- [Atributos de conexión adicionales al usar SQL Server como origen para AWS DMS \(p. 114\)](#)
- [Tipos de datos de origen para SQL Server \(p. 116\)](#)

## Restricciones en el uso de SQL Server como origen para AWS DMS

Las siguientes restricciones se aplican cuando se utiliza una base de datos de SQL Server como origen para AWS DMS:

- La propiedad de identidad para una columna no se migra a una columna de la base de datos de destino.
- En las versiones de motor de AWS DMS anteriores a la versión 2.4.x, los cambios en las filas con más de 8000 bytes de información, incluido el encabezado y la información de mapeo, no se procesan correctamente. Este problema aparece debido a las restricciones en el tamaño del búfer TLOG de SQL Server. Para evitar este problema, utilice la última versión de AWS DMS.
- El enlace de SQL Server no es compatible con el uso de tablas dispersas.
- No se admite la autenticación de Windows.
- Los cambios en los campos calculados en SQL Server no se replican.
- No se permite usar tablas temporales.
- No se admite el cambio de particiones de SQL Server.
- Un índice en clúster en el origen se crea como índice que no está en clúster en el destino.
- Cuando se utilizan las utilidades WRITETEXT y UPDATETEXT, AWS DMS no captura los eventos aplicados en la base de datos de origen.
- No se admite el siguiente patrón de lenguaje de manipulación de datos (DML):

```
SELECT <*> INTO <new_table> FROM <existing_table>
```

- Cuando se utiliza SQL Server como origen, no se admite el cifrado de nivel de columna.
- A causa de un problema con SQL Server 2008 y 2008 R2, AWS DMS no admite auditorías de servidor en SQL Server 2008 o SQL Server 2008 R2 como orígenes. Por ejemplo, si ejecuta el siguiente comando, AWS DMS generará un error.

```
USE [master]
GO
ALTER SERVER AUDIT [my_audit_test-20140710] WITH (STATE=on)
GO
```

## Uso de la replicación continua (CDC) desde un origen de SQL Server

Puede utilizar la replicación continua (captura de cambio de datos o CDC) para una base de datos de SQL Server autoadministrada local o en Amazon EC2, o para una base de datos administrada por Amazon en Amazon RDS.

AWS DMS admite la replicación continua para estas configuraciones de SQL Server:

- Para instancias de SQL Server de origen locales o en Amazon EC2, AWS DMS admite la replicación continua para SQL Server Enterprise, Standard y Developer Edition.
- Para instancias de SQL Server de origen que se ejecutan en Amazon RDS, AWS DMS admite la replicación continua para SQL Server Enterprise a través de SQL Server 2016 SP1. Más allá de esta versión, AWS DMS admite CDC para las ediciones SQL Server Enterprise y Standard.

Si desea que AWS DMS configure automáticamente la replicación continua, la cuenta de usuario de AWS DMS que utilice para conectarse a la base de datos de origen debe tener el rol de servidor fijo sysadmin. Si no desea asignar el rol sysadmin a la cuenta de usuario que utiliza, puede seguir utilizando la replicación continua. Para ello realice la serie de pasos manuales que se explican a continuación.

Los requisitos siguientes se aplican específicamente cuando se utiliza la replicación continua con una base de datos de SQL Server como origen para AWS DMS:

- Es preciso configurar SQL Server para backups completas y debe realizar una backup antes de empezar a replicar los datos.
- El modelo de recuperación debe establecerse en Bulk logged o Full.
- No se admite el backup de SQL Server en varios discos. Si la copia de seguridad está definida para escribir la copia de seguridad de la base de datos en varios archivos y en diferentes discos, AWS DMS no podrá leer los datos y la tarea de AWS DMS generará un error.
- Para orígenes autoadministrados de SQL Server, tenga en cuenta que las definiciones del publicador de replicación de SQL Server para la base de datos de origen que se utilizan en una tarea de CDC de DMS no se eliminan cuando se elimina una tarea. Estas definiciones de SQL Server para orígenes autoadministrados las debe eliminar un administrador del sistema de SQL Server.
- En la CDC, AWS DMS debe buscar copias de seguridad del registro de transacciones de SQL Server para leer los cambios. No AWS DMS admite el uso de copias de seguridad del registro de transacciones de SQL Server que se hubieran creado mediante el software de copia de seguridad de terceros.
- Para orígenes autoadministrados de SQL Server, tenga en cuenta que SQL Server no captura los cambios en las tablas creadas recientemente hasta que se han publicado. Cuando las tablas se añaden a un origen de SQL Server, AWS DMS administra la creación de la publicación. Sin embargo, este proceso puede prolongarse algunos minutos. Las operaciones efectuadas en tablas de nueva creación durante este intervalo no se capturan ni replican en el destino.
- El cambio de captura de datos de AWS DMS requiere que se active FULLLOGGING en SQL Server. Para activar el FULLLOGGING en SQL Server, habilite MS-REPLICATION o CHANGE DATA CAPTURE (CDC).
- No es posible volver a usar el tlog de SQL Server hasta que se hayan procesado los cambios.
- Las operaciones de CDC no se admiten en las tablas con optimización para memoria. Esta restricción se aplica a SQL Server 2014 (cuando se introdujo por vez primera la característica) y a versiones posteriores.

## Captura de datos de cambios para SQL Server

Para un origen autoadministrado de SQL Server, AWS DMS utiliza lo siguiente:

- MS-Replication para capturar los cambios de las tablas con las claves principales. Puede configurar esta opción automáticamente concediendo privilegios sysadmin al usuario del punto de enlace de AWS DMS en la instancia de origen de SQL Server. También puede seguir los pasos que se proporcionan en esta sección para preparar el origen y utilizar un usuario que no sea sysadmin para el punto de enlace de AWS DMS.
- MS-CDC para capturar los cambios de las tablas sin las claves principales. MS-CDC debe estar habilitado en el nivel de base de datos y para cada una de las tablas por separado.

Para un origen de SQL Server que se ejecuta en Amazon RDS, AWS DMS utiliza MS-CDC para capturar los cambios de las tablas, con o sin las claves principales. Es necesario habilitar MS-CDC en el nivel de base de datos y para cada una de las tablas por separado mediante los procedimientos almacenados específicos de Amazon RDS que se describen en esta sección.

Existen varias formas de utilizar una base de datos de SQL Server para la replicación continua (CDC):

- Configurar la replicación continua con el rol sysadmin. (Esto solo se aplica a los orígenes autoadministrados de SQL Server).

- Configurar la replicación continua para que no se utilice el rol sysadmin. (Esto solo se aplica a los orígenes autoadministrados de SQL Server).
- Configurar la replicación continua para una instancia de base de datos de Amazon RDS para SQL Server.

## Configuración de la replicación continua con el rol sysadmin

Para las tablas con claves principales, AWS DMS puede configurar los artefactos necesarios en el origen. Para las tablas sin claves principales, es necesario configurar MS-CDC.

En primer lugar, habilite MS-CDC para la base de datos ejecutando el siguiente comando. Utilice una cuenta que tenga asignado el rol sysadmin.

```
use [DBname]
EXEC sys.sp_cdc_enable_db
```

A continuación, habilite MS-CDC para cada una de las tablas de origen, lo cual se hace con la ejecución del siguiente comando.

```
EXECUTE sys.sp_cdc_enable_table @source_schema = N'MySchema', @source_name =
N'MyTable', @role_name = NULL;
```

Para obtener más información sobre cómo configurar MS-CDC para tablas específicas, consulte la documentación de [SQL Server](#).

## Configuración de la replicación continua sin asignar el rol sysadmin

Puede configurar la replicación continua para un origen de base de datos de SQL Server que no requiera que la cuenta de usuario tenga privilegios de sysadmin.

### Note

Puede realizar este procedimiento, mientras que la tarea DMS se está ejecutando. Si la tarea DMS se detiene, puede llevar a cabo este procedimiento únicamente si no hay ningún registro de transacciones o copias de seguridad de bases de datos en curso. Esto se debe a que SQL Server requiere el privilegio SYSADMIN para consultar las copias de seguridad para la posición LSN.

Para configurar un origen de base de datos de SQL Server para la replicación continua sin utilizar el rol sysadmin

1. Cree una cuenta de SQL Server con autenticación mediante contraseña utilizando SQL Server Management Studio (SSMS). En este ejemplo, se utiliza una cuenta denominada dmstest.
2. En la sección User Mappings (Mapeos de usuario) de SSMS, elija las bases de datos MSDB y MASTER (que concede permiso público) y asigne el rol DB\_OWNER para la base de datos con la que desea utilizar la replicación continua.
3. Abra el menú contextual (haga clic con el botón derecho) en la cuenta nueva, elija Security (Seguridad) y conceda de forma específica el privilegio Connect SQL.
4. Ejecute los siguientes comandos Grant.

```
GRANT SELECT ON FN_DBLOG TO dmstest;
GRANT VIEW SERVER STATE TO dmstest;
use msdb;
GRANT EXECUTE ON MSDB.DBO.SP_STOP_JOB TO dmstest;
GRANT EXECUTE ON MSDB.DBO.SP_START_JOB TO dmstest;
GRANT SELECT ON MSDB.DBO.BACKUPSET TO dmstest;
GRANT SELECT ON MSDB.DBO.BACKUPMEDIAFAMILY TO dmstest;
GRANT SELECT ON MSDB.DBO.BACKUPFILE TO dmstest;
```

5. En SSMS, abra el menú contextual (haga clic con el botón derecho) para la carpeta Replication y, a continuación, elija Configurar distribución. Siga todos los pasos predeterminados y configure esta instancia de SQL Server para su distribución. Se crea una base de datos de distribución en la categoría de bases de datos.
6. Cree una publicación mediante el procedimiento siguiente.
7. Cree una tarea de AWS DMS con SQL Server como punto de enlace de origen utilizando la cuenta de usuario que ha creado.

#### Note

Los pasos de este procedimiento se aplican solo para tablas con claves principales. Seguirá teniendo que habilitar MS-CDC para las tablas sin claves principales.

### Creación de una publicación de SQL Server para la replicación continua

Para utilizar CDC con SQL Server, debe crear una publicación para cada tabla que participe en la replicación continua.

Para crear una publicación para la replicación continua de SQL Server

1. Inicie sesión en SSMS utilizando la cuenta de usuario SYSADMIN.
2. Amplíe Replication (Replicación).
3. Abra el menú contextual (con el botón derecho del ratón) de Local Publications (Publicaciones locales).
4. En New Publication Wizard (Asistente de nueva publicación), elija Next (Siguiente).
5. Elija la base de datos en la que desea crear la publicación.
6. Elija Transactional publication (Publicación transaccional) y, a continuación, elija Next (Siguiente).
7. Amplíe Tables (Tablas) y seleccione las tablas con PK (también las tablas que desea publicar). Seleccione Next (Siguiente).
8. Elija Next (Siguiente) y, a continuación, Next (Siguiente), porque no necesita crear un filtro o agente de instantánea.
9. Elija Security Settings (Configuración de seguridad) y elija Run under the SQL Server Agent service account (Ejecutar en la cuenta de servicio de Agente SQL Server). Asegúrese de elegir By impersonating the process account (Suplantando la cuenta de proceso) para la conexión del editor. Seleccione OK.
10. Seleccione Next (Siguiente).
11. Seleccione Create the publication (Crear la publicación).
12. Proporcione un nombre de la publicación en el formato AR\_PUBLICATION\_000<DBID>.

Por ejemplo, puede nombrar la publicación AR\_PUBLICATION\_00018. También puede utilizar la función DB\_ID en SQL Server. Para obtener más información sobre la función DB\_ID, consulte [la documentación de SQL Server](#).

### Configuración de la replicación continua en una instancia de base de datos de Amazon RDS para SQL Server

Amazon RDS para SQL Server es compatible con MS-CDC para todas las versiones de Amazon RDS para SQL Server Enterprise hasta SQL Server 2016 SP1. Las ediciones estándar de SQL Server 2016 SP1 y posteriores admiten MS-CDC para Amazon RDS para SQL Server.

A diferencia de los orígenes autoadministrados de SQL Server, Amazon RDS para SQL Server no es compatible con MS-Replication. Por lo tanto, AWS DMS necesita utilizar MS-CDC para las tablas con claves principales o sin ellas.

Amazon RDS no concede privilegios de sysadmin para configurar los artefactos de replicación que utiliza AWS DMS para los cambios en curso en una instancia de SQL Server de origen. Debe habilitar MS-CDC en la instancia Amazon RDS utilizando privilegios de usuario maestro en el procedimiento siguiente.

Para habilitar MS-CDC en una instancia de base de datos de RDS para SQL Server

1. Ejecute la siguiente consulta en el nivel de base de datos.

```
exec msdb.dbo.rds_cdc_enable_db '<DB instance name>'
```

2. Para cada tabla con una clave principal, ejecute la siguiente consulta para la que desee habilitar MS-CDC.

```
exec sys.sp_cdc_enable_table
@source_schema = N'db_name',
@source_name = N'table_name',
@role_name = NULL,
@supports_net_changes = 1
GO
```

Para cada tabla con claves únicas pero sin clave principal, ejecute la siguiente consulta para la que desee habilitar MS-CDC.

```
exec sys.sp_cdc_enable_table
@source_schema = N'db_name',
@source_name = N'table_name',
@index_name = N'unique_index_name'
@role_name = NULL,
@supports_net_changes = 1
GO
```

Para cada tabla sin claves principales ni claves únicas, ejecute la siguiente consulta para la que desee habilitar MS-CDC.

```
exec sys.sp_cdc_enable_table
@source_schema = N'db_name',
@source_name = N'table_name',
@role_name = NULL
GO
```

3. Especifique el periodo de retención para que los cambios estén disponibles en el origen utilizando el comando siguiente.

```
EXEC sys.sp_cdc_change_job @job_type = 'capture' ,@pollinginterval = 86400
```

El parámetro `@pollinginterval` se mide en segundos. El comando anterior conserva los cambios durante un día. AWS recomienda un periodo de retención de un día cuando se utiliza MS-CDC con AWS DMS.

## Métodos de compresión compatibles

La siguiente tabla muestra los métodos de compresión que AWS DMS admite para cada versión de SQL Server.

Versión de SQL Server	Compresión de filas y páginas (en la partición)	Formato de almacenamiento vardecimal
2005	No	No
2008	Sí	No
2012	Sí	No
2014	Sí	No

#### Note

No se admiten las columnas dispersas ni la compresión de la estructura de las columnas.

## Trabajar con los grupos de disponibilidad AlwaysOn de SQL Server

La función de grupos de disponibilidad AlwaysOn de SQL Server es una solución de alta disponibilidad y de recuperación ante desastres que representa una alternativa a nivel empresarial para creación de reflejo de la base de datos.

Para utilizar los grupos de disponibilidad AlwaysOn como origen en AWS DMS, siga el procedimiento siguiente:

- Active la opción de distribución de SQL Server en todas las instancias de las réplicas de disponibilidad.
- En la consola de AWS DMS, abra la configuración de bases de datos de origen de SQL Server. En Server Name especifique el nombre del servicio de nombres de dominio (DNS) o la dirección IP que se configuró para el agente de escucha del grupo de disponibilidad.

Al iniciar una AWS DMS tarea por primera vez, es posible que tarde más de lo habitual en iniciarse. Esta lentitud se debe a que el servidor de grupos de disponibilidad duplica la creación de los artículos de la tabla.

## Configuración de una base de datos de SQL Server como origen de replicación para AWS DMS

Puede configurar una base de datos de SQL Server como un origen de replicación para AWS DMS (AWS DMS). Para realizar una réplica exhaustiva de los cambios, se recomienda utilizar las ediciones Enterprise, Standard o Developer de SQL Server. Se requiere una de estas versiones porque son las únicas que incluyen MS-Replication(EE,SE) y MS-CDC(EE,DEV). La instancia de SQL Server de origen también debe configurarse para copias de seguridad completas. Además, AWS DMS debe establecer conexión con un usuario (un inicio de sesión de instancia de SQL Server) que tenga el rol de servidor fijo sysadmin en la base de datos de SQL Server a la que se conecta.

A continuación, puede encontrar información sobre la configuración de SQL Server como el origen de la replicación para AWS DMS.

## Atributos de conexión adicionales al usar SQL Server como origen para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar su origen de SQL Server. Especifique esta configuración al crear el punto de enlace de origen. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

En la tabla siguiente se muestran los atributos de conexión adicionales que puede utilizar con SQL Server como origen:

Nombre	Descripción
<code>safeguardPolicy</code>	<p>Para obtener un rendimiento óptimo, AWS DMS intenta capturar todos los cambios que no se leyeron del registro de transacciones activas (TLOG). Sin embargo, en ocasiones, por las operaciones de truncado, es posible que el TLOG activo no contenga todos los cambios sin leer. Cuando esto ocurre, AWS DMS obtiene acceso al registro de copia de seguridad para capturar los cambios que faltan. Con el fin de reducir al mínimo la necesidad de obtener acceso al registro de copia de seguridad, AWS DMS evita las operaciones de truncado con uno de los siguientes métodos:</p> <ol style="list-style-type: none"> <li>1. Start transactions in the database (Iniciar transacciones en la base de datos): este es el método predeterminado. Cuando se utiliza este método, AWS DMS evita la operación de truncado de TLOG y, para ello, imita una transacción en la base de datos. Siempre que una transacción esté abierta, los cambios que aparecen después de que la transacción se inicia no se truncan. Si necesita que Microsoft Replication esté habilitado en la base de datos, seleccione este método.</li> <li>2. Exclusively use sp_repldone within a single task (Utilizar sp_repldone exclusivamente en una tarea única): cuando se utiliza este método, AWS DMS lee los cambios y, a continuación, utiliza sp_repldone para marcar las transacciones de TLOG como que ya están listas para las operaciones de truncado. Pese a que este método no incluye actividades transaccionales, solo se puede utilizar cuando Microsoft Replication no se está ejecutando. Además, cuando se utiliza este método, solo una tarea de AWS DMS puede tener acceso a la base de datos en cualquier momento. Por lo tanto, si necesita ejecutar tareas de AWS DMS en paralelo en la misma base de datos, utilice el método predeterminado.</li> </ol> <p>Valor predeterminado:  <code>RELY_ON_SQL_SERVER_REPLICATION_AGENT</code></p> <p>Valores válidos: {<code>EXCLUSIVE_AUTOMATIC_TRUNCATION</code>,  <code>RELY_ON_SQL_SERVER_REPLICATION_AGENT</code>}</p> <p>Ejemplo:  <code>safeguardPolicy=RELY_ON_SQL_SERVER_REPLICATION_AGENT;</code></p>
<code>readBackupOnly</code>	<p>Si este parámetro se establece en <code>Y</code>, AWS DMS solo lee los cambios de las copias de seguridad del registro de transacciones, no desde el archivo del registro de transacciones activo durante la replicación continua. Establecer este parámetro en <code>Y</code> le permite controlar el crecimiento del archivo de registro de transacción activo durante la carga completa y las tareas de replicación en curso. Sin embargo, puede añadir latencia de origen a la replicación continua.</p> <p>Valores válidos: <code>N</code> o <code>Y</code>. El valor predeterminado es <code>N</code>.</p>

Nombre	Descripción
	Ejemplo: <code>readBackupOnly=Y;</code>

## Tipos de datos de origen para SQL Server

La migración de datos que utiliza SQL Server como origen para AWS DMS admite la mayoría de los tipos de datos de SQL Server. La siguiente tabla muestra los tipos de datos de origen de SQL Server que se admiten cuando se utiliza AWS DMS y el mapeo predeterminado desde los tipos de datos de AWS DMS.

Para obtener más información sobre cómo ver el tipo de datos que se asigna en el destino, consulte la sección del punto de enlace de destino que esté utilizando.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de SQL Server	Tipos de datos de AWS DMS
BIGINT	INT8
BIT	BOOLEANO
DECIMAL	NUMERIC
INT	INT4
MONEY	NUMERIC
NUMERIC (p,s)	NUMERIC
SMALLINT	INT2
SMALLMONEY	NUMERIC
TINYINT	UINT1
REAL	REAL4
FLOAT	REAL8
DATETIME	DATETIME
DATETIME2 (SQL Server 2008 y versiones posteriores)	DATETIME
SMALLDATETIME	DATETIME
FECHA	FECHA
TIME	TIME
DATETIMEOFFSET	WSTRING
CHAR	STRING
VARCHAR	STRING
VARCHAR (máx.)	CLOB TEXT

Tipos de datos de SQL Server	Tipos de datos de AWS DMS
	<p>Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos CLOB para una tarea en particular.</p> <p>Para las tablas de SQL Server, AWS DMS actualiza las columnas de LOB en el destino incluso para instrucciones UPDATE que no cambian el valor de la columna de LOB en SQL Server.</p> <p>En la CDC, AWS DMS admite tipos de datos CLOB solo en tablas que incluyan una clave principal.</p>
NCHAR	WSTRING
NVARCHAR (longitud)	WSTRING
NVARCHAR (máx.)	<p>NCLOB</p> <p>NTEXT</p> <p>Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos NCLOB para una tarea en particular.</p> <p>Para las tablas de SQL Server, AWS DMS actualiza las columnas de LOB en el destino incluso para instrucciones UPDATE que no cambian el valor de la columna de LOB en SQL Server.</p> <p>En la CDC, AWS DMS admite tipos de datos CLOB solo en tablas que incluyan una clave principal.</p>
BINARY	BYTES
VARBINARY	BYTES
VARBINARY (máx.)	<p>BLOB</p> <p>IMAGE</p> <p>Para las tablas de SQL Server, AWS DMS actualiza las columnas de LOB en el destino incluso para instrucciones UPDATE que no cambian el valor de la columna de LOB en SQL Server.</p> <p>Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos BLOB para una tarea en particular.</p> <p>AWS DMS solo es compatible con tipos de datos BLOB en tablas que incluyan una clave principal.</p>
TIMESTAMP	BYTES

Tipos de datos de SQL Server	Tipos de datos de AWS DMS
UNIQUEIDENTIFIER	STRING
HIERARCHYID	<p>Utilice HIERARCHYID cuando realice replicaciones en un punto de enlace de destino de SQL Server.</p> <p>Utilice WSTRING (250) para realizar replicaciones en el resto de puntos de enlace.</p>
XML	<p>NCLOB</p> <p>Para las tablas de SQL Server, AWS DMS actualiza las columnas de LOB en el destino incluso para instrucciones UPDATE que no cambian el valor de la columna de LOB en SQL Server.</p> <p>Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de los tipos de datos NCLOB para una tarea en particular.</p> <p>En la CDC, AWS DMS admite tipos de datos NCLOB solo en las tablas que incluyan una clave principal.</p>
GEOMETRY	<p>Utilice GEOMETRY cuando realice replicaciones en puntos de enlace de destino que admitan este tipo de datos.</p> <p>Utilice CLOB cuando realice replicaciones en puntos de enlace de destino que no admitan este tipo de datos.</p>
GEOGRAPHY	<p>Utilice GEOGRAPHY cuando realice replicaciones en puntos de enlace de destino que admitan este tipo de datos.</p> <p>Utilice CLOB cuando realice replicaciones en puntos de enlace de destino que no admitan este tipo de datos.</p>

AWS DMS no admite tablas que incluyan campos con los siguientes tipos de datos:

- CURSOR
- SQL\_VARIANT
- TABLE

#### Note

Se admiten tipos de datos definidos por el usuario en función de su tipo base. Por ejemplo, un tipo de datos definido por el usuario basado en DATETIME se gestiona como un tipo de datos DATETIME.

## Uso de Microsoft Azure SQL Database como origen para AWS DMS

Con AWS DMS, puede utilizar Microsoft Azure SQL Database como origen en prácticamente de la misma forma que en SQL Server. AWS DMS admite, como origen, la misma lista de versiones de base de datos que se admiten para SQL Server que se ejecutan localmente o en una instancia Amazon EC2.

Para obtener más información, consulte [Uso de una base de datos de Microsoft SQL Server como origen para AWS DMS \(p. 108\)](#).

### Note

AWS DMS no admite operaciones de captura de datos de cambios (CDC) con Azure SQL Database.

## Uso de una base de datos de PostgreSQL como origen para AWS DMS

Puede migrar datos desde una o varias bases de datos de PostgreSQL con AWS DMS. Si utiliza una base de datos PostgreSQL como origen, podrá migrar datos a otra base de datos PostgreSQL o a una de las bases de datos compatibles. AWS DMS permite utilizar una base de datos PostgreSQL con la versión 9.4 y posteriores como origen de las bases de datos locales, las bases de datos de una instancia EC2, las bases de datos de una instancia de base de datos de Amazon RDS y las bases de datos de una instancia de base de datos de Amazon Aurora con compatibilidad con PostgreSQL.

### Note

PostgreSQL versión 10.x y posteriores contienen numerosos cambios en los nombres de funciones y los nombres de carpeta con respecto a las versiones anteriores. Si utiliza PostgreSQL versión 10.x o posterior como origen para AWS DMS, consulte el tema [Uso de la versión de PostgreSQL 10.x y posteriores como origen para AWS DMS \(p. 127\)](#) para obtener información sobre la preparación de una base de datos como origen para AWS DMS.

Puede utilizar SSL para cifrar las conexiones entre el punto de enlace de PostgreSQL y la instancia de replicación. Para obtener más información sobre el uso de SSL con un punto de enlace de PostgreSQL, consulte [Uso de SSL con AWS Database Migration Service \(p. 51\)](#).

Para una migración homogénea desde una base de datos de PostgreSQL a una base de datos de PostgreSQL en AWS, se cumple lo siguiente:

- Las columnas JSONB en el origen se migran a columnas JSONB en el destino.
- Las columnas JSON se migran como columnas JSON en el destino.
- Las columnas HSTORE se migran como columnas HSTORE en el destino.

Para una migración heterogénea con PostgreSQL como el origen y otro motor de base de datos como destino, la situación es diferente. En este caso, las columnas JSONB, JSON y HSTORE se convierten al tipo intermedio AWS DMS de NCLOB y, a continuación, se traducen al tipo de columna NCLOB correspondiente en el destino. En este caso, AWS DMS trata los datos JSONB como si fueran una columna de LOB. Durante la fase de carga completa de una migración, la columna de destino debe poder contener valores nulos.

AWS DMS admite la captura de datos de cambios (CDC) con las tablas de PostgreSQL con claves primarias. Si una tabla no dispone de una clave primaria, los logs de escritura previa (WAL) no incluyen una imagen anterior de la fila de la base de datos y AWS DMS no puede actualizar la tabla.

AWS DMS admite CDC en bases de datos PostgreSQL de Amazon RDS cuando la instancia de base de datos está configurada para utilizar la replicación lógica. Amazon RDS admite la replicación lógica

de una instancia de base de datos PostgreSQL con la versión 9.4.9 y posteriores y con la versión 9.5.4 y posteriores. Amazon RDS también admite la replicación lógica de una instancia de base de datos de Amazon Aurora con las versiones 2.2.0 y 2.2.1, con compatibilidad con PostgreSQL 10.6.

Para obtener información adicional acerca de cómo trabajar con las bases de datos PostgreSQL y AWS DMS, consulte las secciones siguientes.

#### Temas

- [Migración de PostgreSQL a PostgreSQL con AWS DMS \(p. 120\)](#)
- [Requisitos previos para utilizar una base de datos de PostgreSQL como origen para AWS DMS \(p. 122\)](#)
- [Requisitos de seguridad cuando se utiliza una base de datos de PostgreSQL como origen para AWS DMS \(p. 123\)](#)
- [Restricciones en el uso bases de datos de PostgreSQL como origen para AWS DMS \(p. 123\)](#)
- [Configuración de una instancia de base de datos de Amazon RDS para PostgreSQL como un origen \(p. 124\)](#)
- [Quitar artefactos de AWS DMS de una base de datos de origen PostgreSQL \(p. 126\)](#)
- [Configuración adicional cuando se utiliza una base de datos de PostgreSQL como origen para AWS DMS \(p. 127\)](#)
- [Uso de la versión de PostgreSQL 10.x y posteriores como origen para AWS DMS \(p. 127\)](#)
- [Atributos de conexión adicionales al usar PostgreSQL como origen para AWS DMS \(p. 128\)](#)
- [Tipos de datos de origen para PostgreSQL \(p. 129\)](#)

## Migración de PostgreSQL a PostgreSQL con AWS DMS

Para una migración heterogénea, en la que se migra desde un motor de base de datos distinto de PostgreSQL a una base de datos de PostgreSQL, AWS DMS es casi siempre la mejor herramienta de migración que se puede utilizar. Pero para una migración homogénea, en la que se migra de una base de datos de PostgreSQL a una base de datos de PostgreSQL, las herramientas nativas pueden ser más eficaces.

Es recomendable usar las herramientas de migración de bases de datos de PostgreSQL nativas como `pg_dump` si se dan las condiciones siguientes:

- Se trata de una migración homogénea, en la que se migra desde una base de datos de PostgreSQL de origen a una base de datos de PostgreSQL de destino.
- Se va a migrar una base de datos completa.
- Las herramientas nativas le permiten migrar sus datos con un tiempo de inactividad mínimo.

La utilidad `pg_dump` usa el comando `COPY` para crear un esquema y un volcado de datos de una base de datos de PostgreSQL. El script de volcado generado por `pg_dump` carga los datos en una base de datos con el mismo nombre y vuelve a crear las tablas, los índices y las claves externas. Puede usar el comando `pg_restore` y el parámetro `-d` para restaurar los datos en una base de datos con un nombre diferente.

Para obtener más información sobre la importación de una base de datos de PostgreSQL en Amazon RDS for PostgreSQL o Amazon Aurora (PostgreSQL), consulte <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide//PostgreSQL.Procedural.Importing.html>.

## Uso de DMS para migrar datos de PostgreSQL a PostgreSQL

AWS DMS puede migrar datos desde, por ejemplo, una base de datos de PostgreSQL de origen que esté instalada localmente a una instancia de Amazon RDS for PostgreSQL o Amazon Aurora (PostgreSQL) de destino. Los tipos de datos de PostgreSQL básicos se migran normalmente sin problemas.

Sin embargo, es posible que los tipos de datos que se admiten en la base de datos de origen pero no en el destino no se migren correctamente. AWS DMS transmite algunos tipos de datos como cadenas si el tipo de datos es desconocido. Algunos tipos de datos, como XML y JSON, se pueden migrar correctamente como archivos pequeños, pero se puede producir un error si los documentos son grandes.

En la siguiente tabla se muestran los tipos de datos de PostgreSQL de origen y se indica si pueden migrarse correctamente:

Tipo de datos	Se migra correctamente	Se migrará parcialmente	No se migrará	Comentarios
INTEGER	X			
SMALLINT	X			
BIGINT	X			
NUMERIC/DECIMAL(p,s)		X		Con $0 < p < 39$ y $0 \leq s \leq 38$
NUMERIC/DECIMAL		X		$p > 38$ o $p=s=0$
REAL	X			
DOUBLE	X			
SMALLSERIAL	X			
SERIAL	X			
BIGSERIAL	X			
MONEY	X			
CHAR		X		Sin precisión especificada
CHAR (n)	X			
VARCHAR		X		Sin precisión especificada
VARCHAR (n)	X			
TEXT	X			
BYTEA	X			
TIMESTAMP	X			
TIMESTAMP(Z)		X		
FECHA	X			
TIME	X			
TIME (z)		X		
INTERVAL		X		
BOOLEANO	X			

Tipo de datos	Se migra correctamente	Se migrará parcialmente	No se migrará	Comentarios
ENUM			X	
CIDR			X	
INET			X	
MACADDR			X	
TSVECTOR			X	
TSQUERY			X	
XML		X		
POINT			X	
LINE			X	
LSEG			X	
BOX			X	
PATH			X	
POLYGON			X	
CIRCLE			X	
JSON		X		
ARRAY			X	
COMPOSITE			X	
RANGE			X	

## Requisitos previos para utilizar una base de datos de PostgreSQL como origen para AWS DMS

Para hacer que una base de datos PostgreSQL sea un origen de AWS DMS, siga estos pasos:

- Utilice una base de datos de PostgreSQL que sea de la versión 9.4.x o posteriores.
- Conceda permisos de superusuario para la cuenta de usuario especificada para la base de datos de origen de PostgreSQL.
- Añada la dirección IP del servidor de replicación AWS DMS al archivo de configuración pg\_hba.conf.
- Establezca los siguientes parámetros y valores en el archivo de configuración postgresql.conf:
  - Establezca wal\_level = logical
  - Defina max\_replication\_slots en un valor mayor de 1.

El valor max\_replication\_slots debe definirse en función del número de tareas que desea ejecutar. Por ejemplo, para ejecutar cinco tareas debe establecer un mínimo de cinco ranuras. Las ranuras se abrirán automáticamente en cuanto se inicie una tarea y permanecerán abiertas incluso cuando la tarea ya no se esté ejecutando. Debe eliminar manualmente las ranuras abiertas.

- Defina max\_wal\_senders en un valor mayor de 1.

El parámetro `max_wal_senders` establece el número de tareas simultáneas que pueden ejecutarse.

- Establezca `wal_sender_timeout =0`

El parámetro `wal_sender_timeout` termina la replicación de conexiones que están inactivas por más tiempo de los milisegundos especificados. Aunque el valor predeterminado sea 60 segundos, se recomienda establecer este parámetro en cero, que deshabilita el mecanismo de tiempo de espera.

- El parámetro `idle_in_transaction_session_timeout` en PostgreSQL versiones 9.6 y posteriores le permite hacer que se agote el tiempo de espera de las transacciones inactivas y devuelvan un error. Algunas transacciones de AWS DMS están inactivas durante un tiempo antes de que el motor de AWS DMS las utilice de nuevo. No finalice las transacciones inactivas cuando utilice AWS DMS.

## Requisitos de seguridad cuando se utiliza una base de datos de PostgreSQL como origen para AWS DMS

El único requisito de seguridad cuando se utiliza PostgreSQL como origen es que la cuenta de usuario especificada debe ser un usuario registrado en la base de datos de PostgreSQL.

## Restricciones en el uso bases de datos de PostgreSQL como origen para AWS DMS

Al utilizar PostgreSQL como origen para AWS DMS se aplican las siguientes restricciones:

- Las tablas de captura deben contar con una clave principal. Si una tabla no dispone de clave principal, AWS DMS omite las operaciones de registro `DELETE` y `UPDATE` para dicha tabla.
- No se admite una marca temporal con una columna de tipo de zona horaria.
- AWS DMS omite un intento de actualizar un segmento de clave principal. En estos casos, el destino identifica la actualización como una que no ha actualizado ninguna fila. Sin embargo, dado que los resultados de actualizar una clave primaria en PostgreSQL son imprevisibles, no se escriben registros en la tabla de excepciones.
- AWS DMS no admite la opción de ejecución `Start Process Changes from Timestamp` (Iniciar procesamiento de cambios a partir de una hora de inicio).
- AWS DMS admite el procesamiento de cargas completas y cambios en Amazon RDS para PostgreSQL. Para obtener más información sobre cómo preparar una instancia de base de datos de PostgreSQL y configurarla para utilizar CDC, consulte [Configuración de una instancia de base de datos de Amazon RDS para PostgreSQL como un origen \(p. 124\)](#).
- La replicación de varias tablas que tengan el mismo nombre donde cada nombre está escrito con un uso de las mayúsculas y minúsculas diferente (por ejemplo, `tabla1`, `TABLA1` y `Tabla1`) puede provocar un comportamiento imprevisible. Debido a este problema, AWS DMS no admite este tipo de replicación.
- En la mayoría de los casos AWS DMS admite cambiar el procesamiento de las instrucciones DDL de `CREATE`, `ALTER` y `DROP` para las tablas. AWS DMS no admite este cambio si las tablas se conserven en un bloque de contenido de una función o procedimiento internos o en otros constructos anidados.

Por ejemplo, el siguiente cambio no se capturó:

```
CREATE OR REPLACE FUNCTION attu.create_distributors1() RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
create table attu.distributors1(did serial PRIMARY KEY, name
varchar(40) NOT NULL);
END;
$$;
```

- AWS DMS no admite cambiar el procesamiento de las operaciones TRUNCATE.
- El tipo de datos de LOB OID no se migra al destino.
- Si el origen es una base de datos de PostgreSQL local o una instancia Amazon EC2, asegúrese de que el complemento de salida `test_decoding` esté instalado en su punto de enlace de origen. Puede encontrar este complemento en el paquete contrib de Postgres. Para obtener más información acerca del complemento `test-decoding` consulte la [documentación de PostgreSQL](#).
- AWS DMS no admite cambiar el procesamiento para establecer valores predeterminados de columnas (usando la cláusula ALTER COLUMN SET DEFAULT en instrucciones ALTER TABLE).
- AWS DMS no admite cambiar el procesamiento para establecer la nulabilidad de una columna (usando la cláusula ALTER COLUMN [SET|DROP] NOT NULL en instrucciones ALTER TABLE).
- AWS DMS no admite la replicación de las tablas con particiones. Cuando se detecta una tabla con particiones, sucede lo siguiente:
  - El punto de enlace notifica una lista de las tablas principales y secundarias.
  - AWS DMS crea la tabla en el destino como una tabla normal con las mismas propiedades que las tablas seleccionadas.
  - Si la tabla principal en la base de datos de origen tiene el mismo valor de clave principal que las tablas secundarias, se genera un error de "clave duplicada".

#### Note

Para replicar tablas con particiones desde un origen PostgreSQL a un destino de PostgreSQL, primero debe crear manualmente las tablas principal y secundaria en el destino. A continuación, puede definir una tarea independiente para replicar en estas tablas. En tal caso, debe establecer la configuración de la tarea en Truncate before loading.

#### Note

El tipo de datos NUMERIC de PostgreSQL no tiene un tamaño fijo. Cuando se transfieren datos que tienen el tipo de datos NUMERIC pero sin precisión ni escala, DMS utiliza NUMERIC(28, 6) (con una precisión de 28 y una escala de 6) de forma predeterminada. Por ejemplo, el valor 0,61111104488373 del origen se convierte a 0,611111 en el destino de PostgreSQL.

## Configuración de una instancia de base de datos de Amazon RDS para PostgreSQL como un origen

Puede utilizar una instancia de base de datos de Amazon RDS para PostgreSQL o una réplica de lectura como origen para AWS DMS. Una instancia de base de datos puede utilizarse tanto para la carga completa como para la CDC (replicación continua); una réplica de lectura solo se puede utilizar para tareas de carga completa y no se puede usar para CDC.

Utilice la cuenta de usuario principal de AWS para la instancia de base de datos de PostgreSQL como la cuenta de usuario para el punto de enlace de origen de PostgreSQL para AWS DMS. La cuenta de usuario maestra tiene los roles necesarios para poder capturar datos de cambio (CDC). Si utiliza una cuenta distinta de la cuenta de usuario principal, la cuenta debe tener los roles `rds_superuser` y `rds_replication`. El rol `rds_replication` concede permisos para gestionar ranuras lógicas y para transmitir datos mediante ranuras lógicas.

Si no utiliza la cuenta de usuario principal para la instancia de base de datos, debe crear varios objetos desde la cuenta de usuario principal para la cuenta que utilice. Para obtener información sobre la creación de los objetos necesarios, consulte [Migrar la base de datos de Amazon RDS para PostgreSQL sin usar la cuenta de usuario principal \(p. 125\)](#).

## Utilizar CDC con RDS para una instancia de base de datos de PostgreSQL

Puede utilizar la función de replicación lógica nativa de PostgreSQL para habilitar CDC durante una migración de bases de datos de una instancia de base de datos de PostgreSQL Amazon RDS. Este

enfoque reduce el tiempo de inactividad y garantiza que la base de datos de destino esté sincronizada con el origen de datos de PostgreSQL. Amazon RDS admite la replicación lógica para una instancia de base de datos de PostgreSQL de la versión 9.4.9 y posteriores y 9.5.4 y posteriores.

#### Note

Las réplicas de lectura de Amazon RDS para PostgreSQL no se pueden utilizar para la CDC (replicación continua).

Para habilitar la replicación lógica para una instancia de base de datos de PostgreSQL RDS, haga lo siguiente:

- En general, utilice la cuenta de usuario principal de AWS para la instancia de base de datos de PostgreSQL como la cuenta de usuario para el punto de enlace de origen de PostgreSQL. La cuenta de usuario principal dispone de los roles necesarios que le permiten configurar la CDC. Si utiliza una cuenta que no sea la cuenta de usuario principal, debe crear varios objetos desde la cuenta principal para la cuenta que utilice. Para obtener más información, consulte [Migrar la base de datos de Amazon RDS para PostgreSQL sin usar la cuenta de usuario principal \(p. 125\)](#).
- Establezca en 1 el parámetro `rds.logical_replication` en el grupo de parámetros de la base de datos. Se trata de un parámetro estático que requiere reiniciar la instancia de base de datos para que el parámetro surta efecto. Como parte de la aplicación de este parámetro, AWS DMS establece los parámetros `wal_level`, `max_wal_senders`, `max_replication_slots` y `max_connections`. Estos cambios de parámetro pueden incrementar la generación de WAL, por lo que el parámetro `rds.logical_replication` se debe establecer cuando se utilicen ranuras lógicas.
- Una práctica recomendada consiste en configurar el parámetro `wal_sender_timeout` en 0. Al configurar este parámetro en 0, se impide que PostgreSQL termine las conexiones de replicación que están inactivas durante más tiempo que el tiempo de espera especificado. Cuando AWS DMS está migrando datos, las conexiones de replicación deben durar más tiempo que el tiempo de espera especificado.

## Migrar la base de datos de Amazon RDS para PostgreSQL sin usar la cuenta de usuario principal

En algunos casos, es posible que no utilice la cuenta de usuario principal para la instancia de base de datos de Amazon RDS PostgreSQL que está utilizando como origen. En estos casos, tiene que crear varios objetos para capturar los eventos del lenguaje de definición de datos (DDL). Puede crear estos objetos en una cuenta que no sea la cuenta principal y, a continuación, crear un activador en la cuenta de usuario principal.

#### Note

Si establece el parámetro `captureDDL` en N en el punto de enlace de origen, no tendrá que crear la tabla y el disparador siguientes en la base de datos de origen.

Utilice el siguiente procedimiento para crear estos objetos. Las cuentas de usuario que no sean cuentas principales se denominan cuentas NoPriv en este procedimiento.

#### Para crear objetos

1. Elija el esquema donde deben crearse los objetos. El esquema predeterminado es `public`. Asegúrese de que el esquema exista y que la cuenta `NoPriv` pueda obtener acceso a él.
2. Inicie sesión en la instancia de base de datos de PostgreSQL con la cuenta `NoPriv`.
3. Cree la tabla `awsdms_ddl_audit`. Para ello, ejecute el siguiente comando, sustituyendo `<objects_schema>` en el código siguiente por el nombre del esquema que se va a utilizar.

```
create table <objects_schema>.awsdms_ddl_audit
(
    c_key      bigserial primary key,
    c_time     timestamp,      -- Informational
    c_user     varchar(64),   -- Informational: current_user
    c_txn      varchar(16),   -- Informational: current transaction
    c_tag      varchar(24),   -- Either 'CREATE TABLE' or 'ALTER TABLE' or 'DROP TABLE'
    c_oid      integer,       -- For future use - TG_OBJECTID
    c_name     varchar(64),   -- For future use - TG_OBJECTNAME
    c_schema   varchar(64),   -- For future use - TG_SCHEMANAME. For now - holds
    current_schema
    c_ddlqry   text          -- The DDL query associated with the current DDL event
)
```

4. Cree la función awsdms\_intercept\_ddl. Para ello, ejecute el siguiente comando y sustituya <objects\_schema> en el código siguiente por el nombre del esquema que se va a utilizar.

```
CREATE OR REPLACE FUNCTION <objects_schema>.awsdms_intercept_ddl()
RETURNS event_trigger
LANGUAGE plpgsql
SECURITY DEFINER
AS $$$
declare _qry text;
BEGIN
    if (tg_tag='CREATE TABLE' or tg_tag='ALTER TABLE' or tg_tag='DROP TABLE') then
        SELECT current_query() into _qry;
        insert into <objects_schema>.awsdms_ddl_audit
        values
        (
            default, current_timestamp, current_user, cast(TXID_CURRENT() as
varchar(16)), tg_tag, 0, '', current_schema, _qry
        );
        delete from <objects_schema>.awsdms_ddl_audit;
    end if;
END;
$$;
```

5. Cierre sesión en la cuenta NoPriv e inicie sesión con una cuenta que tenga el rol rds\_superuser asignado.
6. Cree el activador de eventos awsdms\_intercept\_ddl; para ello, ejecute el siguiente comando.

```
CREATE EVENT TRIGGER awsdms_intercept_ddl ON ddl_command_end
EXECUTE PROCEDURE <objects_schema>.awsdms_intercept_ddl();
```

Cuando haya completado el procedimiento anterior, puede crear el punto de enlace de origen de AWS DMS utilizando la cuenta NoPriv.

## Quitar artefactos de AWS DMS de una base de datos de origen PostgreSQL

Para capturar eventos DDL, AWS DMS crea varios artefactos en la base de datos de PostgreSQL cuando comienza una tarea de migración. Cuando se complete la tarea, es posible que quiera eliminar estos artefactos. Para quitar los artefactos, emita las instrucciones siguientes (en el orden en el que aparecen), donde {AmazonRDSMigration} es el esquema en el que se crearon los artefactos:

```
drop event trigger awsdms_intercept_ddl;
```

El disparador de eventos no pertenece a un esquema específico.

```
drop function {AmazonRDSMigration}.awsdms_intercept_ddl()
drop table {AmazonRDSMigration}.awsdms_ddl_audit
drop schema {AmazonRDSMigration}
```

#### Note

La operación de ingresar un esquema se debe realizar con su sumo cuidado y en caso último caso. No ingrese nunca un esquema operativo, especialmente si es público.

## Configuración adicional cuando se utiliza una base de datos de PostgreSQL como origen para AWS DMS

Puede añadir parámetros de configuración adicionales cuando migre datos desde una base de datos de PostgreSQL de dos maneras:

- Puede añadir valores al atributo extra connection para capturar eventos DDL y especificar el esquema en el que se crean los artefactos de la base de datos DDL. Para obtener más información, consulte [Atributos de conexión adicionales al usar PostgreSQL como origen para AWS DMS \(p. 128\)](#).
- Puede anular parámetros de cadenas de conexión. Seleccione esta opción si necesita realizar alguna de las siguientes acciones:
  - Especifique parámetros de AWS DMS internos. Estos parámetros se necesitan en contadas ocasiones y, por tanto, no están a la vista en la interfaz de usuario.
  - Especificar los valores de paso (passthru) para el cliente de base de datos específico. AWS DMS contiene parámetros de paso a través en la cadena de conexión que se pasa al cliente de base de datos.

## Uso de la versión de PostgreSQL 10.x y posteriores como origen para AWS DMS

Las bases de datos de las versiones de PostgreSQL 10.x y posteriores contienen numerosos cambios en los nombres de funciones y los nombres de carpeta de las versiones PostgreSQL anteriores. Estos cambios hacen que determinadas acciones de migración no sean compatibles con versiones anteriores.

Debido a que la mayoría de los cambios son superficiales, AWS DMS ha creado funciones contenedoras que permiten a AWS DMS trabajar con la versión de PostgreSQL 10.x y posteriores. Las funciones contenedoras tienen una mayor prioridad que las funciones en pg\_catalog. Además, nos aseguramos de que la visibilidad de los esquemas existentes no cambia, de forma que no anulamos ninguna otra función del catálogo del sistema como, por ejemplo, las funciones definidas por el usuario.

Para utilizar estas funciones contenedoras antes de realizar ninguna tarea de migración, ejecute el siguiente código SQL en la base de datos de PostgreSQL de origen. Utilice la misma cuenta de usuario de AWS DMS que está utilizando para la base de datos de destino.

```
BEGIN;
CREATE SCHEMA IF NOT EXISTS fnRenames;
CREATE OR REPLACE FUNCTION fnRenames.pg_switch_xlog() RETURNS pg_lsn AS $$ 
    SELECT pg_switch_wal(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_xlog_replay_pause() RETURNS VOID AS $$ 
    SELECT pg_wal_replay_pause(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_xlog_replay_resume() RETURNS VOID AS $$
```

```

    SELECT pg_wal_replay_resume(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_current_xlog_location() RETURNS pg_lsn AS $$
    SELECT pg_current_wal_lsn(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_is_xlog_replay_paused() RETURNS boolean AS $$
    SELECT pg_is_wal_replay_paused(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_xlogfile_name(lsn pg_lsn) RETURNS TEXT AS $$
    SELECT pg_walfile_name(lsn); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_last_xlog_replay_location() RETURNS pg_lsn AS $$
    SELECT pg_last_wal_replay_lsn(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_last_xlog_receive_location() RETURNS pg_lsn AS $$
    SELECT pg_last_wal_receive_lsn(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_current_xlog_flush_location() RETURNS pg_lsn AS $$
    SELECT pg_current_wal_flush_lsn(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_current_xlog_insert_location() RETURNS pg_lsn AS $$
    SELECT pg_current_wal_insert_lsn(); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_xlog_location_diff(lsn1 pg_lsn, lsn2 pg_lsn)
RETURNS NUMERIC AS $$
    SELECT pg_wal_lsn_diff(lsn1, lsn2); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_xlogfile_name_offset(lsn pg_lsn, OUT TEXT, OUT
INTEGER) AS $$
    SELECT pg_walfile_name_offset(lsn); $$ LANGUAGE SQL;
CREATE OR REPLACE FUNCTION fnRenames.pg_create_logical_replication_slot(slot_name name,
plugin name,
temporary BOOLEAN DEFAULT FALSE, OUT slot_name name, OUT xlog_position pg_lsn) RETURNS
RECORD AS $$
    SELECT slot_name::NAME, lsn::pg_lsn FROM
pg_catalog.pg_create_logical_replication_slot(slot_name, plugin,
temporary); $$ LANGUAGE SQL;
ALTER USER <user name> SET search_path = fnRenames, pg_catalog, "$user", public;

-- DROP SCHEMA fnRenames CASCADE;
-- ALTER USER PG_User SET search_path TO DEFAULT;
COMMIT;

```

#### Note

Si no invoca este código inicial en una base de datos PostgreSQL 10.x de origen, se producirá un error de este tipo.

```

2018-10-29T02:57:50 [SOURCE_CAPTURE ]E: RetCode: SQL_ERROR SqlState: 42703
NativeError: 1 Message:
ERROR: column "xlog_position" does not exist;
No query has been executed with that handle [1022502] (ar_odbc_stmt.c:3647)

```

## Atributos de conexión adicionales al usar PostgreSQL como origen para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar el origen de PostgreSQL. Especifique esta configuración al crear el punto de enlace de origen. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

En la tabla siguiente se muestran los atributos de conexión adicionales que puede utilizar con PostgreSQL como origen para AWS DMS:

Nombre	Descripción
captureDDL	Para capturar eventos DDL, AWS DMS crea varios artefactos en la base de datos de PostgreSQL cuando se inicia la tarea.

Nombre	Descripción
	<p>Más adelante podrá quitar estos artefactos según se describe en <a href="#">Quitar artefactos de AWS DMS de una base de datos de origen PostgreSQL (p. 126)</a>.</p> <p>Si este valor se ha establecido en N, no es necesario crear tablas ni disparadores en la base de datos de origen. Para obtener más información, consulte <a href="#">Migrar la base de datos de Amazon RDS para PostgreSQL sin usar la cuenta de usuario principal (p. 125)</a>.</p> <p>Se capturan los eventos DDL transmitidos.</p> <p>Valor predeterminado: Y</p> <p>Valores válidos: Y/N</p> <p>Ejemplo: <code>captureDDLs=Y;</code></p>
<code>ddlArtifactsSchema</code>	<p>El esquema en el que se crearon los artefactos de la base de datos de DDL.</p> <p>Valor predeterminado: público</p> <p>Valores válidos: Cadena</p> <p>Ejemplo: <code>ddlArtifactsSchema=xyzddlschema;</code></p>
<code>failTasksOnLobTruncation</code>	<p>Cuando se establece en true, este valor provoca un error en una tarea si el tamaño real de una columna de LOB es mayor que el <code>LobMaxSize</code> especificado.</p> <p>Si una tarea está establecida en el modo LOB limitado y esta opción está establecida en true, la tarea genera un error en vez de truncar los datos de LOB.</p> <p>Valor predeterminado: false</p> <p>Valores válidos: booleano</p> <p>Ejemplo: <code>failTasksOnLobTruncation=true;</code></p>
<code>executeTimeout</code>	<p>Establece el tiempo de espera de las instrucciones del cliente para la instancia de PostgreSQL, en segundos. El valor de predeterminado es de 60 segundos.</p> <p>Ejemplo: <code>executeTimeout=100;</code></p>

## Tipos de datos de origen para PostgreSQL

La siguiente tabla muestra los tipos de datos de origen de PostgreSQL que se admiten cuando se utiliza AWS DMS y el mapeo predeterminado a los tipos de datos de AWS DMS.

Para obtener más información sobre cómo ver el tipo de datos que se asigna en el destino, consulte la sección del punto de enlace de destino que esté utilizando.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de PostgreSQL	Tipos de datos de AWS DMS
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
NUMERIC (p,s)	Si la precisión es de 0 a 38, utilice NUMERIC. Si la precisión es 39 o superior, utilice STRING.
DECIMAL(P,S)	Si la precisión es de 0 a 38, utilice NUMERIC. Si la precisión es 39 o superior, utilice STRING.
REAL	REAL4
DOUBLE	REAL8
SMALLSERIAL	INT2
SERIAL	INT4
BIGSERIAL	INT8
MONEY	NUMERIC(38,4)  El tipo de datos MONEY se asigna a FLOAT en SQL Server.
CHAR	WSTRING (1)
CHAR(N)	WSTRING (n)
VARCHAR(N)	WSTRING (n)
TEXT	NCLOB
BYTEA	BLOB
TIMESTAMP	TIMESTAMP
TIMESTAMP (z)	TIMESTAMP
TIMESTAMP con zona horaria	No admitido
FECHA	FECHA
TIME	TIME
TIME (z)	TIME
INTERVAL	STRING (128): 1 YEAR, 2 MONTHS, 3 DAYS, 4 HOURS, 5 MINUTES, 6 SECONDS
BOOLEANO	CHAR (5) false o true
ENUM	STRING (64)
CIDR	STRING (50)

Tipos de datos de PostgreSQL	Tipos de datos de AWS DMS
INET	STRING (50)
MACADDR	STRING (18)
BIT (n)	STRING (n)
BIT VARYING (n)	STRING (n)
UUID	STRING
TSVECTOR	CLOB
TSQUERY	CLOB
XML	CLOB
POINT	STRING (255) "(x,y)"
LINE	STRING (255) "(x,y,z)"
LSEG	STRING (255) "((x1,y1),(x2,y2))"
BOX	STRING (255) "((x1,y1),(x2,y2))"
PATH	CLOB "((x1,y1),(xn,yn))"
POLYGON	CLOB "((x1,y1),(xn,yn))"
CIRCLE	STRING (255) "(x,y),r"
JSON	NCLOB
JSONB	NCLOB
ARRAY	NCLOB
COMPOSITE	NCLOB
HSTORE	NCLOB
INT4RANGE	STRING (255)
INT8RANGE	STRING (255)
NUMRANGE	STRING (255)
STRRANGE	STRING (255)

Los tamaños de columna de PostgreSQL afectan a la conversión de tipos de datos LOB de PostgreSQL a tipos de datos de AWS DMS. Para trabajar con esto, siga los pasos que se indican a continuación para los tipos de AWS DMS datos siguientes:

- BLOB: establezca Limit LOB size to (Limitar tamaño de LOB a) en el valor Maximum LOB size (KB) [Tamaño máximo de LOB (KB)] al crear la tarea.
- CLOB: la replicación gestiona cada carácter como carácter UTF8. Por lo tanto, busque la longitud del texto con más caracteres en la columna, que se muestra aquí como `max_num_chars_text`, y utilícela para especificar el valor de `Limit LOB size to` (Limitar tamaño de LOB a). Si los datos incluyen caracteres de 4 bytes, multiplique por 2 para especificar el valor `Limit LOB size to` (Limitar tamaño

de LOB a), que está en bytes. En este caso, Limit LOB size to (Limitar tamaño de LOB a) es igual a `max_num_chars_text` multiplicado por 2.

- NCLOB: la replicación gestiona cada carácter como carácter de dos bytes. Por lo tanto, busque la longitud del texto con más caracteres en la columna (`max_num_chars_text`) y multiplíquela por 2 para especificar el valor de Limit LOB size to (Limitar tamaño de LOB a). En este caso, Limit LOB size to (Limitar tamaño de LOB a) es igual a `max_num_chars_text` multiplicado por 2. Si los datos incluyen caracteres de 4 bytes, multiplíquelos por 2 de nuevo. En este caso, Limit LOB size to (Limitar tamaño de LOB a) es igual a `max_num_chars_text` multiplicado por 4.

## Uso de una base de datos compatible con MySQL como origen para AWS DMS

Puede migrar datos desde cualquier base de datos compatible con MySQL (MySQL, MariaDB o Amazon Aurora MySQL) utilizando AWS Database Migration Service. Se admiten las versiones 5.5, 5.6 y 5.7 de MySQL, así como MariaDB y Amazon Aurora MySQL para instancias locales. Se admiten todas las bases de datos MySQL administradas por AWS (Amazon RDS para MySQL, Amazon RDS para MariaDB, Amazon Aurora MySQL) como orígenes para AWS DMS.

Puede utilizar SSL para cifrar las conexiones entre su punto de enlace compatible con MySQL y la instancia de replicación. Para obtener más información acerca de cómo utilizar SSL con un punto de enlace compatible con MySQL, consulte [Uso de SSL con AWS Database Migration Service \(p. 51\)](#).

En las secciones siguientes, el término "autoadministrado" se aplica a cualquier base de datos que se instala en una instancia local o en Amazon EC2. El término "administrada por Amazon" se aplica a cualquier base de datos en Amazon RDS, Amazon Aurora o Amazon S3.

Para obtener más información sobre cómo trabajar con bases de datos compatibles con MySQL y AWS DMS, consulte las secciones siguientes.

### Temas

- [Migración de MySQL a MySQL con AWS DMS \(p. 132\)](#)
- [Uso de cualquier base de datos compatible con MySQL como origen para AWS DMS \(p. 134\)](#)
- [Uso de una base de datos compatible con MySQL autoadministrada como origen para AWS DMS \(p. 135\)](#)
- [Uso de una base de datos compatible con MySQL administrada por Amazon como origen para AWS DMS \(p. 136\)](#)
- [Restricciones en el uso bases de datos MySQL como origen para AWS DMS \(p. 136\)](#)
- [Atributos de conexión adicionales al usar MySQL como origen para AWS DMS \(p. 137\)](#)
- [Tipos de datos de origen para MySQL \(p. 138\)](#)

## Migración de MySQL a MySQL con AWS DMS

Para una migración heterogénea, en la que se migra desde un motor de base de datos distinto de MySQL a una base de datos de MySQL, AWS DMS es casi siempre la mejor herramienta de migración que se puede utilizar. Pero para una migración homogénea, en la que se migra de una base de datos de MySQL a una base de datos de MySQL, las herramientas nativas pueden ser más eficaces.

Es recomendable usar las herramientas de migración de bases de datos de MySQL nativas como `mysqldump` si se dan las condiciones siguientes:

- Se trata de una migración homogénea, en la que se migra desde una base de datos de MySQL de origen a una base de datos de MySQL de destino.
- Se va a migrar una base de datos completa.

- Las herramientas nativas le permiten migrar sus datos con un tiempo de inactividad mínimo.

También puede importar datos desde una base de datos de MySQL o MariaDB existente en una instancia de base de datos de MySQL o MariaDB en Amazon RDS. Para ello, copie la base de datos con [mysqldump](#) y canalícela directamente a la instancia de base de datos MySQL o MariaDB de Amazon RDS. La utilidad de línea de comandos [mysqldump](#) suele usarse para crear backups y transferir datos de un servidor MySQL o MariaDB a otro. Se incluye con el software de cliente de MySQL y MariaDB.

Para obtener más información sobre la importación de una base de datos MySQL en Amazon RDS for MySQL o Amazon Aurora (MySQL), consulte [Importación de datos en una instancia de base de datos MySQL](#) e [Importación de datos de una base de datos MySQL o MariaDB a una instancia de base de datos MySQL o MariaDB de Amazon RDS](#).

## Uso de AWS DMS para migrar datos de MySQL a MySQL

AWS DMS puede migrar datos desde, por ejemplo, una base de datos de MySQL de origen que esté instalada localmente a una instancia de Amazon RDS for MySQL o Amazon Aurora (MySQL) de destino. Los tipos de datos de MySQL básicos se migran normalmente sin problemas.

Sin embargo, es posible que los tipos de datos que se admiten en la base de datos de origen pero no en el destino no se migren correctamente. AWS DMS transmite algunos tipos de datos como cadenas si el tipo de datos es desconocido. Algunos tipos de datos, como XML y JSON, se pueden migrar correctamente como archivos pequeños, pero se puede producir un error si los documentos son grandes.

En la siguiente tabla se muestran los tipos de datos de MySQL de origen y se indica si pueden migrarse correctamente:

Tipo de datos	Se migra correctamente	Se migrará parcialmente	No se migrará	Comentarios
INT	X			
BIGINT	X			
MEDIUMINT	X			
TINYINT	X			
DECIMAL (p,s)	X			
BINARY	X			
BIT (M)	X			
BLOB	X			
LONGBLOB	X			
MEDIUMBLOB	X			
TINYBLOB	X			
FECHA	X			
DATETIME	X			
TIME		X		
TIMESTAMP	X			

Tipo de datos	Se migra correctamente	Se migrará parcialmente	No se migrará	Comentarios
YEAR	X			
DOUBLE	X			
FLOAT		X		
VARCHAR(N)	X			
VARBINARY(N)	X			
CHAR(N)	X			
TEXT	X			
LONGTEXT	X			
MEDIUMTEXT	X			
TINYTEXT	X			
GEOMETRY			X	
POINT			X	
LINESTRING			X	
POLYGON			X	
MULTILINESTRING			X	
MULTIPOLYGON			X	
GEOMETRYCOLLECTION			X	
ENUM		X		
SET		X		

## Uso de cualquier base de datos compatible con MySQL como origen para AWS DMS

Antes de comenzar a trabajar con una base de datos MySQL como origen para AWS DMS, asegúrese de cumplir los siguientes requisitos previos. Estos requisitos previos se aplican a fuentes autoadministradas o administradas por Amazon.

Debe tener una cuenta para AWS DMS que tiene el rol de administrador de replicación. El rol necesita los siguientes privilegios:

- REPLICATION CLIENT – Este privilegio es necesario solamente para la tarea de captura de datos de cambios (CDC). Es decir, las tareas de solo carga completa no necesitan este privilegio.
- REPLICATION SAVE – Este privilegio es necesario solamente para la tarea de captura de datos de cambios (CDC). Es decir, las tareas de solo carga completa no necesitan este privilegio.
- SUPER – este privilegio es necesario únicamente en versiones de MySQL anteriores a la 5.6.6.

El usuario de AWS DMS también debe disponer de privilegios SELECT para las tablas de origen designadas para la replicación.

## Uso de una base de datos compatible con MySQL autoadministrada como origen para AWS DMS

Puede utilizar las siguientes bases de datos compatibles con MySQL autoadministradas como orígenes para AWS DMS:

- MySQL Community Edition
- MySQL Standard Edition
- MySQL Enterprise Edition
- MySQL Cluster Carrier Grade Edition
- MariaDB Community Edition
- MariaDB Enterprise Edition
- Column Store de MariaDB

Debe habilitar el log binario si tiene previsto usar la captura de datos de cambio (CDC). Para habilitar el registro binario, debe configurar los siguientes parámetros en el archivo `my.ini` de MySQL (Windows) o `my.cnf` (UNIX).

Parámetro	Valor
<code>server_id</code>	Establezca este parámetro con un valor de 1 o superior.
<code>log-bin</code>	Establezca la ruta del archivo de registro binario, como por ejemplo <code>log-bin=E:\ MySql_Logs\BinLog</code> . No incluya la extensión del archivo.
<code>binlog_format</code>	Establezca este parámetro en <code>ROW</code> .
<code>expire_logs_days</code>	Establezca este parámetro con un valor de 1 o superior. Para evitar la sobrecarga de espacio en disco, se recomienda que no utilice el valor 0, que es el predeterminado.
<code>binlog_checksum</code>	Establezca este parámetro en <code>NONE</code> .
<code>binlog_row_image</code>	Establezca este parámetro en <code>FULL</code> .
<code>log_slave_updates</code>	Establezca este parámetro en <code>TRUE</code> si está utilizando una réplica de lectura de MySQL o MariaDB como origen.

Si su origen utiliza el motor de base de datos NDB (agrupado), deben configurarse los siguientes parámetros para habilitar la CDC en tablas que utilicen ese motor de almacenamiento. Añada estos cambios en el archivo de MySQL `my.ini` (Windows) o `my.cnf` (UNIX).

Parámetro	Valor
<code>ndb_log_bin</code>	Establezca este parámetro en <code>ON</code> . Este valor garantiza que los cambios en las tablas en clúster se anotan en los registros binarios.
<code>ndb_log_update_as_write</code>	Establezca este parámetro en <code>OFF</code> . Este valor impide escribir instrucciones <code>UPDATE</code> como instrucciones <code>INSERT</code> en el registro binario.

Parámetro	Valor
ndb_log_updated_only	Establezca este parámetro en OFF. Este valor garantiza que el registro binario contiene la totalidad de la fila y no solo las columnas que se han modificado.

## Uso de una base de datos compatible con MySQL administrada por Amazon como origen para AWS DMS

Puede utilizar las siguientes bases de datos compatibles con MySQL administradas por Amazon como orígenes para AWS DMS:

- MySQL Community Edition
- MariaDB Community Edition
- Compatibilidad de Amazon Aurora con MySQL

Cuando utilice una base de datos compatible con MySQL administrada por Amazon como origen para AWS DMS, asegúrese de cumplir los siguientes requisitos previos:

- Habilite copias de seguridad automáticas. Para obtener más información acerca de la configuración de copias de seguridad automáticas, consulte [Trabajo con copias de seguridad automatizadas](#) en la Guía del usuario de Amazon RDS.
- Habilite el registro binario si tiene previsto usar la captura de datos de cambio (CDC). Para obtener más información acerca de la configuración del registro binario para una base de datos de Amazon RDS para MySQL, consulte [Trabajo con copias de seguridad automatizadas](#) en la Guía del usuario de Amazon RDS.
- Asegúrese de que los registros binarios están disponibles para AWS DMS. Dado que las bases de datos compatibles con MySQL administradas por Amazon purgan los logs binarios tan pronto como sea posible, debe aumentar el tiempo durante el cual los logs permanecen disponibles. Por ejemplo, para incrementar la retención de logs a 24 horas, ejecute el siguiente comando.

```
call mysql.rds_set_configuration('binlog retention hours', 24);
```

- Defina el parámetro `binlog_format` como "ROW".
- Defina el parámetro `binlog_checksum` como "NONE". Para obtener más información acerca de la configuración de parámetros en Amazon RDS para MySQL, consulte [Trabajo con copias de seguridad automatizadas](#) en la Guía del usuario de Amazon RDS.
- Si está usando una réplica de lectura de Amazon RDS para MySQL o de Amazon RDS para MariaDB como origen, habilite las copias de seguridad en la réplica de lectura.

## Restricciones en el uso bases de datos MySQL como origen para AWS DMS

Cuando se utiliza una base de datos MySQL como origen, AWS DMS no admite lo siguiente:

- No se admite la captura de datos de cambios (CDC) para Amazon RDS para MySQL 5.5 o inferior. Para Amazon RDS para MySQL, debe utilizar la versión 5.6 o superior para habilitar la CDC.
- No se admiten las instrucciones del lenguaje de definición de datos (DDL) `DROP TABLE` y `RENAME TABLE`. Además, no se admiten todas las declaraciones DDL para tablas con particiones.

- Para tablas con particiones en el origen, cuando se establece Target table preparation mode (Modo de preparación de tabla de destino) en Drop tables on target (Borrar tablas en el destino), AWS DMS crea una tabla sencilla sin ninguna partición en el destino de MySQL. Para migrar tablas con particiones a una tabla particionada en el destino, cree previamente las tablas con particiones en la base de datos MySQL de destino.
- No se admite el uso de una instrucción ALTER TABLE <nombre\_tabla> ADD COLUMN <nombre\_columna> para añadir columnas al comienzo (FIRST) o en la mitad de una tabla (AFTER). Las columnas siempre se añaden al final de la tabla.
- No se admite la CDC cuando un nombre de tabla contiene mayúsculas y minúsculas y el motor de origen está alojado en un sistema operativo con nombres de archivo que no distinguen entre mayúsculas y minúsculas. Un ejemplo es Windows u OS X con HFS+.
- No se admite de columna AR\_H\_USER.
- El atributo AUTO\_INCREMENT en una columna no se migra a una columna de la base de datos de destino.
- No se admite la captura de los cambios cuando los registros binarios no se almacenan en almacenamiento de bloques estándar. Por ejemplo, CDC no funciona cuando los registros binarios se almacenan en Amazon S3.
- AWS DMS crea tablas de destino con el motor de almacenamiento InnoDB de forma predeterminada. Si necesita utilizar un motor de almacenamiento distinto de InnoDB, debe crear manualmente la tabla y migrar a ella mediante el [modo "no hacer nada"](#).
- No se pueden utilizar réplicas de lectura Aurora MySQL como origen para AWS DMS.
- Si el origen compatible con MySQL se detiene durante la carga completa, la tarea de AWS DMS no se detiene con un error. La tarea finaliza correctamente, pero es posible que el destino no esté sincronizado con el origen. Si esto ocurre, reinicie la tarea o vuelva a cargar las tablas afectadas.
- Los índices creados en una parte del valor de una columna no se migran. Por ejemplo, el índice CREATE INDEX first\_ten\_chars ON customer (name(10)) no se crea en el destino.
- En algunos casos, la tarea se configura para que no replique LOB ("SupportLobs" es false en la configuración de tareas o se marca "Don't include LOB columns" (No incluir columnas de LOB) en la consola de tareas). En estos casos, AWS DMS no migra ninguna columna MEDIUMBLOB, LONGBLOB, MEDIUMTEXT y LONGTEXT al destino.

Las columnas BLOB, TINYBLOB, TEXT y TINYTEXT no se ven afectadas y se migran al destino.

## Atributos de conexión adicionales al usar MySQL como origen para AWS DMS

Puede utilizar los atributos adicionales de conexión para configurar un origen de MySQL. Especifique esta configuración al crear el punto de enlace de origen. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

En la tabla siguiente se muestran los atributos de conexión adicionales disponibles cuando utiliza Amazon RDS MySQL como origen para AWS DMS:

Nombre	Descripción
eventsPollInterval	Especifica la frecuencia con la que se va a consultar el registro binario para comprobar si hay cambios o eventos nuevos cuando la base de datos está inactiva.  Valor predeterminado: 5  Valores válidos: 1-60  Ejemplo: eventsPollInterval=5;

Nombre	Descripción
	En el ejemplo, AWS DMS comprueba si se han producido cambios en los logs binarios cada cinco segundos.
<code>initstmt=SET time_zone</code>	<p>Especifica la zona horaria para el origen de la base de datos MySQL. Las marcas temporales se traducen a la zona horaria especificada.</p> <p>Valor predeterminado: UTC</p> <p>Valores válidos: cualquier abreviatura de tres caracteres para la zona horaria que desea utilizar, como por ejemplo UTC, EST o GMT. Los valores válidos son las abreviaturas de zona horaria estándar para el sistema operativo que aloja la base de datos MySQL de origen.</p> <p>Ejemplo: <code>initstmt=SET time_zone=UTC;</code></p>
<code>afterConnectScript</code>	<p>Especifica un script que se ejecuta inmediatamente después de que AWS DMS se conecta al punto de enlace. La tarea de migración continúa ejecutándose independientemente de si la instrucción SQL se realiza correcta o incorrectamente.</p> <p>Valores válidos: una o varias instrucciones SQL válidas separadas mediante punto y coma.</p> <p>Ejemplo: <code>afterConnectScript=ALTER SESSION SET CURRENT_SCHEMA = system;</code></p>
<code>CleanSrcMetadataOnMismatch</code>	<p>Limpia y vuelve a crear la información de metadatos de la tabla en la instancia de replicación cuando se produce una discordancia. Por ejemplo, en una situación en la que se ejecuta una DDL modificada en la tabla podría dar como resultado información distinta sobre la tabla en caché en la instancia de replicación. Booleano.</p> <p>Valor predeterminado: <code>false</code></p> <p>Ejemplo: <code>CleanSrcMetadataOnMismatch=false;</code></p>

## Tipos de datos de origen para MySQL

La siguiente tabla muestra los tipos de datos de origen de base de datos MySQL que se admiten cuando se utiliza AWS DMS y el mapeo predeterminado desde los tipos de datos de AWS DMS.

Para obtener más información sobre cómo ver el tipo de datos que se asigna en el destino, consulte la sección del punto de enlace de destino que esté utilizando.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de MySQL	Tipos de datos de AWS DMS
INT	INT4
MEDIUMINT	INT4

Tipos de datos de MySQL	Tipos de datos de AWS DMS
BIGINT	INT8
TINYINT	INT1
DECIMAL (10)	NUMERIC (10,0)
BINARY	BYTES(1)
BIT	BOOLEAN
BIT(64)	BYTES(8)
BLOB	BYTES(66535)
LONGBLOB	BLOB
MEDIUMBLOB	BLOB
TINYBLOB	BYTES(255)
FECHA	FECHA
DATETIME	DATETIME
TIME	STRING
TIMESTAMP	DATETIME
YEAR	INT2
DOUBLE	REAL8
FLOAT	REAL(DOUBLE)  El rango FLOAT admitido es de -1,79E+308 a -2,23E-308, 0 y de 2,23E-308 a 1,79E+308  Si los valores de FLOAT no están en este rango, asigne el tipo de datos FLOAT al tipo de datos STRING.
VARCHAR (45)	WSTRING (45)
VARCHAR (2000)	WSTRING (2000)
VARCHAR (4000)	WSTRING (4000)
VARBINARY (4000)	BYTES (4000)
VARBINARY (2000)	BYTES (2000)
CHAR	WSTRING
TEXT	WSTRING (65535)
LONGTEXT	NCLOB
MEDIUMTEXT	NCLOB
TINYTEXT	WSTRING (255)

Tipos de datos de MySQL	Tipos de datos de AWS DMS
GEOMETRY	BLOB
POINT	BLOB
LINESTRING	BLOB
POLYGON	BLOB
MULTIPOINT	BLOB
MULTILINESTRING	BLOB
MULTIPOLYGON	BLOB
GEOMETRYCOLLECTION	BLOB

#### Note

Si los tipos de datos TIMESTAMP y DATETIME se especifican con un valor "cero" (es decir, 0000-00-00), asegúrese de que la base de datos de destino de la tarea de replicación admita valores "cero" para los tipos de datos TIMESTAMP y DATETIME. De lo contrario, estos valores se registran con un valor NULL en el destino.

Los siguientes tipos de datos de MySQL solo se admiten en carga completa.

Tipos de datos de MySQL	Tipos de datos de AWS DMS
ENUM	STRING
SET	STRING

## Uso de una base de datos SAP ASE como origen para AWS DMS

Puede migrar datos desde una base de datos SAP Adaptive Server Enterprise (ASE), —anteriormente denominada Sybase—, con AWS DMS. Con una base de datos de SAP ASE como origen, podrá migrar datos a ninguna de las demás bases de datos de AWS DMS destino. AWS DMS es compatible con las versiones de SAP ASE 12.5.3 o superior, 15, 15.5, 15.7, 16 y posteriores como fuentes.

Para obtener información adicional acerca de cómo trabajar con las bases de datos SAP ASE y AWS DMS, consulte las secciones siguientes.

#### Temas

- [Requisitos previos para utilizar una base de datos SAP ASE como origen para AWS DMS \(p. 141\)](#)
- [Restricciones en el uso de SAP ASE como origen para AWS DMS \(p. 141\)](#)
- [Permisos necesarios para usar SAP ASE como origen para AWS DMS \(p. 141\)](#)
- [Quitar el punto de truncado \(p. 142\)](#)
- [Tipos de datos de origen para SAP ASE \(p. 142\)](#)

## Requisitos previos para utilizar una base de datos SAP ASE como origen para AWS DMS

Para que una base de datos SAP ASE sea un origen para AWS DMS, haga lo siguiente:

- Habilite la replicación de SAP ASE para las tablas mediante el comando `sp_setreptable`.
- Inhabilite RepAgent en la base de datos de SAP ASE.
- Para replicar a la versión 15.7 de SAP ASE en una instancia Amazon EC2 en Microsoft Windows configurada para caracteres no latinos (por ejemplo, chino), instale SAP ASE 15.7 SP121 en el equipo de destino.

## Restricciones en el uso de SAP ASE como origen para AWS DMS

Al utilizar una base de datos SAP ASE como origen para AWS DMS se aplican las siguientes restricciones:

- Solo se puede ejecutar una tarea AWS DMS para cada base de datos de SAP ASE.
- No se puede cambiar el nombre de una tabla. Por ejemplo, el siguiente comando devuelve un error:

```
sp_rename 'Sales.SalesRegion', 'SalesReg';
```

- No se puede cambiar el nombre de una columna. Por ejemplo, el siguiente comando devuelve un error:

```
sp_rename 'Sales.Sales.Region', 'RegID', 'COLUMN';
```

- Los valores situados al final de las cadenas de tipo de datos binarios se truncan cuando se replican para la base de datos de destino. Por ejemplo, 0x0000000000000000000000000000000100000100000000 en la tabla de origen se convierte en 0x00000000000000000000000000000001000001, en la tabla de destino.
- Si la base de datos predeterminada está configurada para no permitir valores NULL, AWS DMS crea la tabla de destino con columnas que no permiten los valores NULL. Por lo tanto, si una tarea de replicación de carga completa o captura de datos de cambio (CDC) contiene valores vacíos, AWS DMS genera un error. Puede evitar que se produzcan estos errores permitiendo valores NULL en la base de datos de origen ejecutando los siguientes comandos.

```
sp_dboption <database name>, 'allow nulls by default', 'true'  
go  
use <database name>  
CHECKPOINT  
go
```

- No se admite el comando de índice `reorg rebuild`.
- No se admiten los clústeres.

## Permisos necesarios para usar SAP ASE como origen para AWS DMS

Para utilizar una base de datos SAP ASE como origen en una tarea de AWS DMS, conceda los siguientes permisos en la base de datos de AWS DMS a la cuenta de usuario especificada en las definiciones de la base de datos SAP ASE.

- `sa_role`
- `replication_role`

- `sybase_ts_role`
- Si habilita la opción Automatically enable Sybase replication (Habilitar automáticamente la replicación Sybase) en la pestaña Advanced (Opciones avanzadas) al crear el punto de enlace de origen de SAP ASE, también debe conceder permiso a AWS DMS para ejecutar el procedimiento almacenado `sp_setreptable`.

## Quitar el punto de truncado

Al iniciarse una tarea, AWS DMS establece una entrada `$replication_truncation_point` en la vista del sistema `syslogshold`, lo que indica que se está realizando un proceso de replicación. Aunque AWS DMS está en funcionamiento, ofrece el punto de truncado de la replicación periódicamente, según la cantidad de datos que ya se han copiado en el destino.

Una vez que se ha establecido la entrada `$replication_truncation_point`, mantenga la tarea de AWS DMS en ejecución para evitar que el log de la base de datos se haga demasiado grande. Si desea detener la tarea de AWS DMS de forma permanente, quite el punto de truncado de la replicación mediante el siguiente comando:

```
dbcc settrunc('ltm','ignore')
```

Una vez se quite el punto de truncado, no puede reanudar la tarea de AWS DMS. La sesión se seguirá truncando de forma automática en los puntos de control (si se ha establecido el truncado automático).

## Tipos de datos de origen para SAP ASE

Para ver una lista de los tipos de datos de origen de SAP ASE que se admiten cuando se utiliza AWS DMS y el mapeo predeterminado desde los tipos de datos AWS DMS, consulte la tabla siguiente. AWS DMS no admite las tablas de origen de SAP ASE con columnas de tipo de datos definido por el usuario (UDT). Las columnas que se replican con este tipo de datos se crean como NULL.

Para obtener más información sobre cómo ver el tipo de datos que se asigna en el destino, consulte la sección [Destinos para la migración de datos \(p. 157\)](#) de su punto de enlace de destino.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de SAP ASE	Tipos de datos de AWS DMS
BIGINT	INT8
BINARY	BYTES
BIT	BOOLEANO
CHAR	STRING
FECHA	FECHA
DATETIME	DATETIME
DECIMAL	NUMERIC
DOUBLE	REAL8
FLOAT	REAL8

Tipos de datos de SAP ASE	Tipos de datos de AWS DMS
IMAGE	BLOB
INT	INT4
MONEY	NUMERIC
NCHAR	WSTRING
NUMERIC	NUMERIC
NVARCHAR	WSTRING
REAL	REAL4
SMALLDATETIME	DATETIME
SMALLINT	INT2
SMALLMONEY	NUMERIC
TEXT	CLOB
TIME	TIME
TINYINT	UINT1
UNICHAR	CARÁCTER UNICODE
UNIEXT	NCLOB
UNIVARCHAR	UNICODE
VARBINARY	BYTES
VARCHAR	STRING

## Uso de MongoDB como origen para AWS DMS

AWS DMS admite las versiones 2.6.x y 3.x de MongoDB como base de datos de origen.

Si no está familiarizado con MongoDB, tenga en cuenta los siguientes conceptos importantes sobre las bases de datos MongoDB:

- Un registro en MongoDB es un documento formado por una estructura de datos compuesta de pares de campo y valor. El valor de un campo puede incluir otros documentos, matrices y matrices de documentos. Un documento es más o menos equivalente a una fila en una tabla de base de datos relacional.
- Una colección en MongoDB es un grupo de documentos y es aproximadamente equivalente a una tabla de base de datos relacional.
- Internamente, un documento de MongoDB se almacena como archivo JSON binario (BSON) en formato comprimido, que incluye un tipo para cada campo del documento. Cada documento tiene un identificador único.

AWS DMS admite dos modos de migración cuando se utiliza MongoDB como origen. El modo de migración se especifica con el parámetro `Metadata mode` (Modo metadatos) en la consola de administración de AWS o con el atributo de conexión adicional `nestingLevel` al crear el punto de enlace

de MongoDB. La selección del modo de migración afecta al formato resultante de los datos de destino, como se indica a continuación.

#### Modo documento

En el modo documento, el documento de MongoDB se migra tal cual, es decir, sus datos se consolidan en una única columna de una tabla de destino denominada `_doc`. El modo documento es la configuración predeterminada al usar MongoDB como punto de enlace de origen.

Por ejemplo, tenga en cuenta los siguientes documentos en una colección de MongoDB llamada `myCollection`.

```
> db.myCollection.find()
{ "_id" : ObjectId("5a94815f40bd44d1b02bdfe0"), "a" : 1, "b" : 2, "c" : 3 }
{ "_id" : ObjectId("5a94815f40bd44d1b02bdfe1"), "a" : 4, "b" : 5, "c" : 6 }
```

Después de migrar los datos a una tabla de base de datos relacional utilizando el modo documento, los datos se estructuran de la siguiente forma. Los campos de datos del documento de MongoDB se consolidan en la columna `_doc`.

oid_id	_doc
5a94815f40bd44d1b02bdfe0	{ "a" : 1, "b" : 2, "c" : 3 }
5a94815f40bd44d1b02bdfe1	{ "a" : 4, "b" : 5, "c" : 6 }

Si lo desea, puede establecer el `extractDocID` del atributo de conexión adicional en `true` para crear otra columna denominada `_id` que actúe como clave principal. Si va a utilizar la captura de datos de cambio (CDC), establezca este parámetro en `true`.

En modo de documento, AWS DMS gestiona la creación y el cambio de nombre de colecciones como esta:

- Si se añade una nueva colección a la base de datos de origen, AWS DMS crea una tabla de destino para la colección y replica los documentos.
- Si se cambia el nombre de una colección existente en la base de datos de origen, AWS DMS no cambia el nombre de la tabla de destino.

#### Modo de tabla

En el modo de tabla, AWS DMS transforma cada uno de los campos de nivel superior de un documento de MongoDB en una columna de la tabla de destino. Si un campo está anidado, AWS DMS reúne los valores anidados en una sola columna. A continuación, AWS DMS añade un campo clave y tipos de datos al conjunto de columnas de la tabla de destino.

Para cada documento de MongoDB, AWS DMS añade cada clave y tipo al conjunto de columnas de la tabla de destino. Por ejemplo, con el modo de tabla, AWS DMS migra el ejemplo anterior a la siguiente tabla.

oid_id	a	b	c
5a94815f40bd44d1b02bdfe0		2	3
5a94815f40bd44d1b02bdfe1		5	6

Los valores anidados se apllanan en una columna que contiene nombres de clave separados por puntos. El nombre de la columna será la concatenación de los nombres de los campos reunidos,

separados por puntos. Por ejemplo, AWS DMS migra un documento JSON con un campo de valores anidados, como `{"a" : {"b" : {"c": 1}}}`, a una columna denominada `a.b.c`.

Para crear las columnas de destino, AWS DMS analiza un número determinado de documentos de MongoDB y crea un conjunto de todos los campos y sus tipos. AWS DMS utiliza este conjunto a continuación para crear las columnas de la tabla de destino. Si crea o modifica el punto de enlace de origen de MongoDB mediante la consola de , puede especificar el número de documentos que se van a analizar. El valor predeterminado es de 1000 documentos. Si utiliza la CLI de AWS, puede usar el atributo `docsToInvestigate` de conexión adicional.

En modo de tabla, AWS DMS gestiona documentos y colecciones de este modo:

- Cuando añada un documento a una colección existente, el documento se replica. Si hay campos que no existen en el destino, estos campos no se replican.
- Al actualizar un documento, el documento actualizado se replican. Si hay campos que no existen en el destino, estos campos no se replican.
- Se admite en toda su extensión la eliminación de documentos.
- Cuando se añade una colección nueva, no se crea una tabla nueva en el destino si se efectúa mientras se desarrolla una tarea de CDC.
- No se admite cambiar el nombre a una colección.

## Permisos necesarios al usar MongoDB como origen para AWS DMS

Para una migración de AWS DMS con un origen de MongoDB, puede crear una cuenta de usuario con privilegios raíz o bien un usuario que tenga permisos solamente en la base de datos que migrar.

El código siguiente crea un usuario para que sea la cuenta raíz.

```
use admin
db.createUser(
{
  user: "root",
  pwd: "<password>",
  roles: [ { role: "root", db: "admin" } ]
})
```

El código siguiente crea un usuario con privilegios mínimos en la base de datos que se va a migrar.

```
use <database_to_migrate>
db.createUser(
{
  user: "<dms-user>",
  pwd: "<password>",
  roles: [ { role: "read", db: "local" }, "read" ]
})
```

## Configuración de un conjunto de réplicas de MongoDB para la captura de datos de cambios (CDC)

Para utilizar la replicación continua o la captura de datos de cambio (CDC) con MongoDB, AWS DMS requiere acceso al registro de operaciones (oplog) de MongoDB. Para crear dicho log, debe implementar

un conjunto de réplicas si no existe ninguno. Para obtener más información, consulte la [documentación de MongoDB](#).

Puede utilizar CDC con el nodo principal o secundario de un conjunto de réplicas de MongoDB como punto de enlace de origen.

Para convertir una instancia independiente a un conjunto de réplicas

1. Usar la línea de comandos, conectarse a mongo.

```
mongo localhost
```

2. Detenga el servicio mongod.

```
service mongod stop
```

3. Reinicie mongod utilizando el siguiente comando:

```
mongod --replSet "rs0" --auth -port <port_number>
```

4. Pruebe la conexión con el conjunto de réplicas con los siguientes comandos:

```
mongo -u root -p <password> --host rs0/localhost:<port_number>
--authenticationDatabase "admin"
```

Si tiene previsto realizar una migración con el modo documento, seleccione la opción `_id as a separate column` al crear el punto de enlace de MongoDB. Si se selecciona esta opción, se crea otra columna denominada `_id`, que actúa como clave principal. AWS DMS necesita esta segunda columna para poder admitir operaciones de lenguaje de manipulación de datos (DML).

## Requisitos de seguridad cuando se utiliza MongoDB como origen para AWS DMS

AWS DMS admite dos métodos de autenticación para MongoDB. Los dos métodos de autenticación se utilizan para cifrar la contraseña, de forma que solo se pueda utilizar cuando el parámetro `authType` se haya establecido en `PASSWORD` (CONTRASEÑA).

Los métodos de autenticación de MongoDB son los siguientes:

- MONOGODB-CR: valor predeterminado cuando se utiliza la autenticación de MongoDB 2.x.
- SCRAM-SHA-1: valor predeterminado cuando se utiliza la autenticación de MongoDB versión 3.x.

Si no se especifica un método de autenticación, AWS DMS utiliza el método predeterminado para la versión del origen de MongoDB.

## Restricciones cuando se utiliza MongoDB como origen para AWS DMS

A continuación, se describen las restricciones que se aplican cuando se utiliza MongoDB como origen para AWS DMS:

- Cuando la `_id` opción se establece como una columna independiente, la cadena del identificador no puede superar los 200 caracteres.

- Las claves de ID de objeto y de tipo de matriz se convierten en columnas que tienen los prefijos `oid` y `array` en el modo de tabla.  
Internamente, se hace referencia a estas columnas con los nombres con prefijos. Si utiliza reglas de transformación en AWS DMS que hacen referencia a estas columnas, debe especificar la columna con prefijos. Por ejemplo, especifique  `${oid__id}` y no  `${_id}` o  `${array__addresses}` y no  `${_addresses}`.
- Los nombres de colecciones no pueden incluir el símbolo del dólar (\$).
- El modo de tabla y el modo de documento tienen las limitaciones tratadas con anterioridad.

## Atributos de conexión adicionales al usar MongoDB como origen para AWS DMS

Cuando configure el punto de enlace de origen de MongoDB, puede especificar atributos de conexión adicionales. Los atributos de conexión adicionales se especifican mediante pares de clave-valor. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

En la tabla siguiente se describen los atributos de conexión adicionales disponibles cuando se utilizan bases de datos MongoDB como origen de AWS DMS.

Nombre de atributo	Valores válidos	Valor predeterminado y descripción
<code>authType</code>	NO PASSWORD	PASSWORD: si se selecciona NO, los parámetros del nombre de usuario y la contraseña no se utilizan y pueden estar vacíos.
<code>authMechanism</code>	DEFAULT MONGODB_CR SCRAM_SHA_1	DEFAULT: para la versión 2.x de MongoDB, utilice MONGODB_CR. Para la versión 3.x de MongoDB, utilice SCRAM_SHA_1. Este atributo no se utiliza cuando <code>authType=NO</code> .
<code>nestingLevel</code>	NONE ONE	NONE: indique NONE para utilizar el modo de documento. Especifique ONE para utilizar el modo de tabla.
<code>extractDocID</code>	true false	FALSE: utilice este atributo cuando <code>nestingLevel</code> se haya establecido en NONE.
<code>docsToInvestigate</code>	Un número entero positivo mayor que 0.	1000: utilice este atributo cuando <code>nestingLevel</code> se haya establecido en ONE.
<code>authSource</code>	Un nombre de la base de datos MongoDB válido.	admin: este atributo no se utiliza cuando <code>authType=NO</code> .

### Note

Si el punto de enlace de destino es DocumentDB, asegúrese de que los siguientes atributos de conexión adicionales para el origen de MongoDB se establezcan tal y como se indica a continuación:

- `nestingLevel=NONE`

- `extractDocID=false`

Para obtener más información, consulte [Uso de Amazon DocumentDB como destino para AWS Database Migration Service \(p. 232\)](#).

## Tipos de datos de origen para MongoDB

La migración de datos que utiliza MongoDB como origen para AWS DMS admite la mayoría de los tipos de datos de MongoDB. En la tabla siguiente, puede encontrar los tipos de datos origen de MongoDB que se admiten cuando se utiliza AWS DMS y el mapeo predeterminado desde los tipos de datos de AWS DMS. Para obtener más información sobre los tipos de datos de MongoDB, consulte [BSON Types](#) en la documentación de MongoDB.

Para obtener más información sobre cómo ver el tipo de datos que se asigna en el destino, consulte la sección del punto de enlace de destino que esté utilizando.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de MongoDB	Tipos de datos de AWS DMS
Booleano	Bool
Binario	BLOB
Fecha	Fecha
Marca temporal	Fecha
Int	INT4
Long	INT8
Double	REAL8
Cadena (UTF-8)	CLOB
Matriz	CLOB
OID	Cadena
REGEX	CLOB
CODE	CLOB

## Uso de Amazon S3 como origen para AWS DMS

Puede migrar datos desde un bucket de Amazon S3 mediante AWS DMS. Para ello, dé acceso a un bucket de Amazon S3 que contenga uno o varios archivos de datos. En ese bucket de S3 incluya un archivo JSON que describa el mapeo entre los datos y las tablas de la base de datos de los datos de esos archivos.

Los archivos de datos de origen deben estar en el bucket de Amazon S3 antes de que comience la carga completa. El nombre del bucket se especifica mediante el parámetro `bucketName`.

Los archivos de datos de origen deben estar en formato de valores separados por comas (CSV). Denomínelos utilizando la siguiente convención de nomenclatura. En esta convención, `schemaName` es el esquema de origen y `tableName` es el nombre de una tabla dentro de dicho esquema.

```
/schemaName/LOAD001.csv  
/schemaName/LOAD002.csv  
/schemaName/LOAD003.csv  
...
```

Por ejemplo, supongamos que sus archivos de datos están en `mybucket`, en la siguiente ruta de Amazon S3.

```
s3://mybucket/hr/employee
```

En tiempo de carga, AWS DMS presupone que el nombre del esquema de origen es `hr` y que el nombre de la tabla de origen es `employee`.

Además de `bucketName` (que es necesario), tiene la opción de proporcionar un parámetro `bucketFolder` para especificar en qué ubicación del bucket de Amazon S3 debe buscar AWS DMS archivos de datos. Prosiguiendo con el ejemplo anterior, si establece `bucketFolder` en `sourcedata`, AWS DMS leerá los archivos de datos en la ruta siguiente.

```
s3://mybucket/sourcedata/hr/employee
```

Puede especificar el delimitador de columnas, el delimitador de filas, el indicador de valor nulo y otros parámetros mediante los atributos de conexión adicionales. Para obtener más información, consulte [Atributos de conexión adicionales para Amazon S3 como origen para AWS DMS \(p. 152\)](#).

## Definición de tablas externas para Amazon S3 como origen para AWS DMS

Además de los archivos de datos, debe indicar también una definición de tabla externa. Una definición de tabla externa es un documento JSON en el que se describe cómo AWS DMS debe interpretar los datos de Amazon S3. El tamaño máximo de este documento es de 2 MB. Si crea un punto de enlace de origen con la consola de administración de AWS DMS, puede escribir el JSON directamente en el cuadro de mapeo de la tabla. Si utiliza la AWS Command Line Interface (AWS CLI) o la API de AWS DMS para realizar migraciones, puede crear un archivo JSON para especificar la definición de tabla externa.

Supongamos que tiene un archivo de datos que contiene la información siguiente.

```
101,Smith,Bob,4-Jun-14,New York  
102,Smith,Bob,8-Oct-15,Los Angeles  
103,Smith,Bob,13-Mar-17,Dallas  
104,Smith,Bob,13-Mar-17,Dallas
```

A continuación se muestra un ejemplo de definición de tabla externa para estos datos.

```
{  
    "TableCount": "1",  
    "Tables": [  
        {  
            "TableName": "employee",  
            "TablePath": "hr/employee/",  
            "TableOwner": "hr",  
            "TableColumns": [  
                {  
                    "ColumnName": "Id",  
                    "ColumnType": "INT8",  
                    "ColumnNullable": "false",  
                    "ColumnDefault": null,  
                    "ColumnPrecision": null,  
                    "ColumnScale": null  
                }  
            ]  
        }  
    ]  
}
```

```
        "ColumnIsPk": "true"
    },
{
    "ColumnName": "LastName",
    "ColumnType": "STRING",
    "ColumnLength": "20"
},
{
    "ColumnName": "FirstName",
    "ColumnType": "STRING",
    "ColumnLength": "30"
},
{
    "ColumnName": "HireDate",
    "ColumnType": "DATETIME"
},
{
    "ColumnName": "OfficeLocation",
    "ColumnType": "STRING",
    "ColumnLength": "20"
}
],
"TableColumnsTotal": "5"
}
]
```

Los elementos de este documento JSON son los siguientes:

**TableCount:** el número de tablas de origen. En este ejemplo, solo hay una tabla.

**Tables:** una matriz que consta de un mapa JSON por tabla de origen. En este ejemplo, solo hay un mapa. Cada mapa está formado por los siguientes elementos:

- **TableName:** el nombre de la tabla de origen.
- **TablePath:** ruta de su bucket de Amazon S3 donde AWS DMS puede encontrar el archivo de carga de datos completa. Si se especifica un valor **bucketFolder**, este valor se anexa delante de la ruta.
- **TableOwner:** nombre del esquema para esta tabla.
- **TableColumns:** matriz de uno o varios mapas, en la que cada mapa describe una columna de la tabla de origen.
  - **ColumnName:** nombre de una columna de la tabla de origen.
  - **ColumnType:** tipo de datos de la columna. Para consultar los tipos de datos válidos, vea [Tipos de datos de origen para Amazon S3 \(p. 153\)](#).
  - **ColumnLength:** número de bytes de esta columna.
  - **ColumnNullable:** (opcional) valor booleano que es **true** si esta columna puede contener valores **NULL**.
  - **ColumnIsPk:** (opcional) valor booleano que es **true** si esta columna forma parte de la clave principal.
- **TableColumnsTotal:** número total de columnas. Este número debe coincidir con el número de elementos de la matriz **TableColumns**.

En el ejemplo anterior, algunas de las columnas son de tipo STRING. En este caso, utilice el elemento **ColumnLength** para especificar el número máximo de caracteres.

**ColumnLength** se aplica en los siguientes tipos de datos:

- **BYTE**

- STRING

Si no especifica lo contrario, AWS DMS presupone que `ColumnLength` es cero.

Para una columna de tipo NUMERIC, tiene que especificar la precisión y la escala. Precisión es el número total de dígitos de un número y escala es el número de dígitos situados a la derecha de la coma decimal. Utilice los elementos `ColumnPrecision` y `ColumnScale` para esto, tal y como se muestra a continuación.

```
...
{
    "ColumnName": "HourlyRate",
    "ColumnType": "NUMERIC",
    "ColumnPrecision": "5"
    "ColumnScale": "2"
}
...
```

## Uso de la CDC con Amazon S3 como origen para AWS DMS

Una vez que AWS DMS realiza una carga de datos completa, tiene la opción de replicar los cambios de datos en el punto de enlace de destino. Para ello, debe cargar los archivos de captura de datos de cambios (archivos CDC) en su bucket de Amazon S3. AWS DMS lee estos archivos CDC cuando los carga y, a continuación, aplica los cambios en el punto de enlace de destino.

Los archivos CDC se denominan de la forma siguiente:

```
CDC00001.csv
CDC00002.csv
CDC00003.csv
...
```

Para indicar dónde AWS DMS puede encontrar los archivos, debe especificar el parámetro `cdcPath`. Prosiguiendo con el ejemplo anterior, si establece `cdcPath` en `changedata`, AWS DMS leerá los archivos CDC en la ruta siguiente.

```
s3://mybucket/changedata
```

Los registros de un archivo CDC se formatean de la siguiente manera:

- Operación: la operación de cambio que se va a realizar: `INSERT`, `UPDATE` o `DELETE`. Estas palabras clave no distinguen entre mayúsculas y minúsculas.
- Nombre de tabla: nombre de la tabla de origen.
- Nombre de esquema: nombre del esquema de origen.
- Datos: una o varias columnas que representan los datos que se van a cambiar.

A continuación se muestra un ejemplo de un archivo CDC para una tabla con el nombre `employee`.

```
INSERT,employee,hr,101,Smith,Bob,4-Jun-14,New York
UPDATE,employee,hr,101,Smith,Bob,8-Oct-15,Los Angeles
UPDATE,employee,hr,101,Smith,Bob,13-Mar-17,Dallas
DELETE,employee,hr,101,Smith,Bob,13-Mar-17,Dallas
```

## Requisitos previos cuando se usa Amazon S3 como origen para AWS DMS

Para utilizar Amazon S3 como origen para AWS DMS, su bucket de S3 de origen debe estar en la misma región de AWS la instancia de replicación de DMS que migra los datos. Además, la cuenta de AWS que utiliza para la migración debe tener acceso de lectura al bucket de origen.

El rol de AWS Identity and Access Management (IAM) asignado a la cuenta de usuario utilizada para crear la tarea de migración debe tener el siguiente conjunto de permisos.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "s3:GetObject"  
            ],  
            "Resource": [  
                "arn:aws:s3:::mybucket*"  
            ]  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "s3>ListBucket"  
            ],  
            "Resource": [  
                "arn:aws:s3:::mybucket*"  
            ]  
        }  
    ]  
}
```

## Atributos de conexión adicionales para Amazon S3 como origen para AWS DMS

Puede especificar las siguientes opciones como atributos de conexión adicionales.

Opción	Descripción
<code>bucketFolder</code>	(Opcional) Nombre de carpeta en el bucket de S3. Si se proporciona este atributo, los archivos de datos de origen y los archivos CDC se leen en la ruta <code>bucketFolder/schemaName/tableName/</code> . Si no se especifica este atributo, la ruta utilizada es <code>schemaName/tableName/</code> . Ejemplo:  <code>bucketFolder=testFolder;</code>
<code>bucketName</code>	Nombre del bucket de S3. Ejemplo:  <code>bucketName=buckettest;</code>
<code>cdcPath</code>	Ubicación de los archivos de captura de datos de cambios (CDC). Este atributo es obligatorio si una tarea captura datos de cambios; de lo contrario, es opcional. Si <code>cdcPath</code> está incluido, AWS DMS lee los archivos CDC desde esta ruta y replica los cambios de datos en el punto de enlace de destino. Para obtener más información, consulte <a href="#">Uso de la CDC con Amazon S3 como origen para AWS DMS (p. 151)</a> . Ejemplo:

Opción	Descripción
	<code>cdcPath=dataChanges;</code>
<code>csvDelimiter</code>	Delimitador utilizado para separar columnas en los archivos de origen. El valor predeterminado es una coma. Ejemplo: <code>csvDelimiter=,;</code>
<code>csvRowDelimiter</code>	Delimitador utilizado para separar filas en los archivos de origen. El valor predeterminado es una nueva línea (\n). Ejemplo: <code>csvRowDelimiter=\n;</code>
<code>externalTableDefinition</code>	Objeto JSON que describe cómo AWS DMS debe interpretar los datos del bucket de Amazon S3 durante la migración. Para obtener más información, consulte <a href="#">Definición de tablas externas para Amazon S3 como origen para AWS DMS (p. 149)</a> . Ejemplo: <code>externalTableDefinition=&lt;json_object&gt;;</code>
<code>ignoreHeaderRows</code>	Cuando este valor se establece en 1, AWS DMS omite el encabezado de la primera fila en un archivo .csv. Un valor de 1 habilita la característica, un valor de 0 deshabilita la característica. El valor predeterminado es 0.  <code>ignoreHeaderRows=1;</code>
<code>rfc4180</code>	Cuando este valor se establece en true o y, las comillas dobles de inicio tienen que ir seguidas de comillas dobles finales. Este formato cumple con RFC 4180. Cuando el valor se establece en false, los literales de cadena se copian en el destino tal cual. En este caso, un delimitador (fila o columna) señala el final del campo. Por lo tanto, no puede utilizar un delimitador como parte de la cadena, ya que señala el final del valor.  El valor predeterminado es <code>true</code> .  Valores válidos: <code>true</code> , <code>false</code> , <code>y</code> , <code>n</code>  Ejemplo: <code>rfc4180=false;</code>

## Tipos de datos de origen para Amazon S3

La migración de datos que utiliza Amazon S3 como origen para AWS DMS tiene que asignar datos desde Amazon S3 a tipos de datos de AWS DMS. Para obtener más información, consulte [Definición de tablas externas para Amazon S3 como origen para AWS DMS \(p. 149\)](#).

Para obtener más información sobre cómo ver el tipo de datos que se asigna en el destino, consulte la sección del punto de enlace de destino que esté utilizando.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Los siguientes tipos de datos de AWS DMS se utilizan con Amazon S3 como origen:

- **BYTE**: requiere `ColumnLength`. Para obtener más información, consulte [Definición de tablas externas para Amazon S3 como origen para AWS DMS \(p. 149\)](#).
- **FECHA**

- TIME
- DATETIME
- TIMESTAMP
- INT1
- INT2
- INT4
- INT8
- NUMERIC: requiere ColumnPrecision y ColumnScale. Para obtener más información, consulte [Definición de tablas externas para Amazon S3 como origen para AWS DMS \(p. 149\)](#).
- REAL4
- REAL8
- STRING: requiere ColumnLength. Para obtener más información, consulte [Definición de tablas externas para Amazon S3 como origen para AWS DMS \(p. 149\)](#).
- UINT1
- UINT2
- UINT4
- UINT8
- BLOB
- CLOB
- BOOLEAN

## Uso de una base de datos IBM Db2 para Linux, Unix y Windows (Db2 LUW) como origen de AWS DMS

Puede migrar datos desde un IBM Db2 para Linux, Unix y Windows (Db2 LUW) base de datos a cualquier base de datos de destino con compatibles AWS Database Migration Service (AWS DMS). AWS DMS admite como orígenes de migración Db2 LUW las siguientes versiones:

- Version 9.7, se admiten todos los Fix Packs
- Versión 10.1, se admiten todos los Fix Packs
- Version 10.5, se admiten todos los Fix Packs excepto Fix Pack 5
- Versión 11.1, se admiten todos los Fix Packs

Puede utilizar la Capa de conexión segura (SSL) para cifrar las conexiones entre el punto de enlace de Db2 LUW y la instancia de replicación. Para utilizar SSL, utilice la versión del motor de AWS DMS 2.4.2 o superior. Para obtener más información sobre cómo utilizar SSL con un punto de enlace de Db2 LUW, consulte [Uso de SSL con AWS Database Migration Service \(p. 51\)](#).

## Requisitos previos cuando se usa Db2 LUW como origen de AWS DMS

Los siguientes requisitos previos son necesarios para poder utilizar una base de datos Db2 LUW como origen.

Para habilitar la replicación continua, también llamada captura de datos de cambios (CDC), haga lo siguiente:

- Establezca la base de datos para que sea recuperable, que AWS DMS requiere para capturar los cambios. Una base de datos es recuperable si uno o ambos parámetros de configuración de la base de datos, LOGARCHMETH1 y LOGARCHMETH2, se establecen en ON.
- Conceda a la cuenta de usuario los siguientes permisos:
  - SYSADM o DBADM
  - DATAACCESS

## Restricciones cuando se utiliza Db2 LUW como origen de AWS DMS

AWS DMS no admite bases de datos en clúster. Sin embargo, puede definir una base de datos Db2 LUW independiente para cada uno de los puntos de enlace de un clúster. Para obtener más información, consulte la [documentación de Db2 LUW](#).

Al utilizar la replicación continua (CDC), se aplican las siguientes restricciones:

- Al truncarse una tabla con varias particiones, el número de eventos DDL que se muestra en la consola de AWS DMS es igual al número de particiones. Esto se debe a que Db2 LUW registra un DDL individual para cada partición.
- Las siguientes acciones de DDL no se admiten en las tablas con particiones:
  - ALTER TABLE ADD PARTITION
  - ALTER TABLE DETACH PARTITION
  - ALTER TABLE ATTACH PARTITION
- No se admite el tipo de datos DECFLOAT. Por lo tanto, los cambios en las columnas DECFLOAT se omiten durante la replicación continua.
- No se admite la instrucción RENAME COLUMN.
- Al realizar actualizaciones en las tablas de clústeres multidimensionales (MDC), cada actualización se muestra en la consola de AWS DMS como INSERT + DELETE.
- Cuando la opción de tarea Include LOB columns in replication (Incluir columnas LOB en la replicación) no está habilitada, toda tabla que tenga columnas LOB se suspende durante la replicación continua.
- Cuando la opción de auditar tabla está habilitada, el primer registro de marca temporal de la tabla de auditoría es NULL.
- Cuando la opción para cambiar tabla está habilitada, el primer registro de marca temporal de la tabla es cero (es decir, 1970-01-01 00:00:00.000000).
- Para Db2 LUW versión 10.5 y posteriores: las columnas de cadena de longitud variable con datos que se almacenan fuera de fila se omiten. Esta limitación solo se aplica a las tablas creadas con tamaño de fila ampliado.

## Atributos de conexión adicionales al usar Db2 LUW como origen de AWS DMS

Puede utilizar atributos de conexión adicionales para configurar su origen de Db2 LUW. Especifique esta configuración al crear el punto de enlace de origen. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

En la tabla siguiente se muestran los atributos de conexión adicionales que puede utilizar con Db2 LUW como origen.

Nombre	Descripción
CurrentLSN	Para la replicación continua de cambios (CDC), utilice CurrentLSN para especificar un número de secuencia de registro (LSN) donde desea que comience la replicación.
MaxKBytesPerRead	Número máximo de bytes por lectura, como valor NUMBER. El valor predeterminado es 64 KB.
SetDataCaptureChanges	Habilita la replicación continua (CDC) como valor booleano. El valor predeterminado es true.

## Tipos de datos de origen para Db2 LUW de IBM

La migración de datos que utiliza Db2 LUW como origen para AWS DMS admite la mayoría de los tipos de datos de Db2 LUW. La siguiente tabla muestra los tipos de datos de origen de Db2 LUW que se admiten cuando se utiliza AWS DMS y el mapeo predeterminado desde los tipos de datos de AWS DMS. Para obtener más información sobre los tipos de datos de Db2 LUW, consulte la [documentación sobre Db2 LUW](#).

Para obtener más información sobre cómo ver el tipo de datos que se asigna en el destino, consulte la sección del punto de enlace de destino que esté utilizando.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de Db2 LUW	Tipos de datos de AWS DMS
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
DECIMAL (p,s)	NUMERIC (p,s)
FLOAT	REAL8
DOUBLE	REAL8
REAL	REAL4
DECFLOAT (p)	Si la precisión es 16, entonces REAL8; si la precisión es 34, entonces STRING
GRAPHIC (n)	WSTRING, para cadenas de gráficos de longitud fija de caracteres de dos bytes con una longitud mayor que 0 y menor o igual a 127
VARGRAPHIC (n)	WSTRING, para cadenas de gráficos de longitud variable con una longitud mayor que 0 y menor o igual a 16.352 caracteres de dos bytes
LONG VARGRAPHIC (n)	CLOB, para cadenas de gráficos de longitud variable con una longitud mayor que 0 y menor o igual a 16.352 caracteres de dos bytes

Tipos de datos de Db2 LUW	Tipos de datos de AWS DMS
CHARACTER (n)	STRING, para cadenas de longitud fija de caracteres de dos bytes con una longitud mayor que 0 y menor o igual a 255
VARCHAR (n)	STRING, para cadenas de longitud variable de caracteres de dos bytes con una longitud mayor que 0 y menor o igual a 32.704
LONG VARCHAR (n)	CLOB, para cadenas de longitud variable de caracteres de dos bytes con una longitud mayor que 0 y menor o igual a 32.704
CHAR (n) FOR BIT DATA	BYTES
VARCHAR (n) FOR BIT DATA	BYTES
LONG VARCHAR FOR BIT DATA	BYTES
FECHA	FECHA
TIME	TIME
TIMESTAMP	DATETIME
BLOB (n)	BLOB  La longitud máxima es de 2 147 483 647 bytes
CLOB (n)	CLOB  La longitud máxima es de 2 147 483 647 bytes
DBCLOB (n)	CLOB  La longitud máxima es 1 073 741 824 de caracteres de dos bytes
XML	CLOB

## Destinos para la migración de datos

AWS Database Migration Service (AWS DMS) puede utilizar la mayoría de las bases de datos más conocidas como destino para replicar datos. El destino puede hallarse en una instancia Amazon Elastic Compute Cloud (Amazon EC2), en una instancia Amazon Relational Database Service (Amazon RDS) o en una base de datos local.

Las bases de datos incluyen lo siguiente:

### Bases de datos locales y de instancia EC2

- Las versiones 10g, 11g y 12c de Oracle para las ediciones Enterprise, Standard, Standard One y Standard Two
- Versiones de Microsoft SQL Server 2005, 2008, 2008R2, 2012, 2014 y 2016, para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.
- Versiones de MySQL 5.5, 5.6 y 5.7
- MariaDB (se admite como destino de datos compatible con MySQL)

- Versiones 9.4 y posteriores de PostgreSQL
- SAP Adaptive Server Enterprise (ASE) versiones 15, 15.5, 15.7, 16 y posteriores

Bases de datos de instanciasAmazon RDS, Amazon Redshift, Amazon S3, Amazon DynamoDB, Amazon Elasticsearch Service, Amazon Kinesis Data Streams y Amazon DocumentDB

- Amazon RDS para versiones de Oracle 11g (versiones 11.2.0.3.v1 y posteriores) y 12c para las ediciones Enterprise, Standard, Standard One y Standard Two
- Amazon RDS para versiones de Microsoft SQL Server 2008R2, 2012 y 2014, para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.
- Amazon RDS para versiones de MySQL 5.5, 5.6 y 5.7
- Amazon RDS para MariaDB (se admite como destino de datos compatible con MySQL)
- Amazon RDS para versiones 9.4 y posteriores de PostgreSQL
- Compatibilidad de Amazon Aurora con MySQL
- Amazon Aurora con compatibilidad con PostgreSQL
- Amazon Redshift
- Amazon S3
- Amazon DynamoDB
- Amazon Elasticsearch Service
- Amazon Kinesis Data Streams
- Amazon DocumentDB (con compatibilidad con MongoDB)

#### Note

AWS DMS no admite la migración entre regiones de los siguientes tipos de puntos de enlace de destino:

- Amazon DynamoDB
- Amazon Elasticsearch Service
- Amazon Kinesis Data Streams

#### Temas

- [Uso de una base de datos de Oracle como destino para AWS Database Migration Service \(p. 159\)](#)
- [Uso de una base de datos de Microsoft SQL Server como destino para AWS Database Migration Service \(p. 164\)](#)
- [Uso de una base de datos de PostgreSQL como destino para AWS Database Migration Service \(p. 167\)](#)
- [Uso de una base de datos compatible con MySQL como destino para AWS Database Migration Service \(p. 170\)](#)
- [Uso de una base de datos de Amazon Redshift como destino para AWS Database Migration Service \(p. 175\)](#)
- [Uso de una base de datos de SAP ASE como destino para AWS Database Migration Service \(p. 187\)](#)
- [Uso de Amazon S3 como destino para AWS Database Migration Service \(p. 189\)](#)
- [Uso de una base de datos de Amazon DynamoDB como destino para AWS Database Migration Service \(p. 210\)](#)
- [Uso de Amazon Kinesis Data Streams como destino para AWS Database Migration Service \(p. 223\)](#)
- [Uso de un clúster de Amazon Elasticsearch Service como destino de AWS Database Migration Service \(p. 229\)](#)
- [Uso de Amazon DocumentDB como destino para AWS Database Migration Service \(p. 232\)](#)

## Uso de una base de datos de Oracle como destino para AWS Database Migration Service

Puede migrar datos a bases de datos de destino de Oracle mediante AWS DMS, ya sea desde otra base de datos de Oracle o desde una de las otras bases de datos admitidas. Puede utilizar la Capa de conexión segura (SSL) para cifrar las conexiones entre el punto de enlace de Oracle y la instancia de replicación. Para obtener más información acerca de cómo usar SSL con un punto de enlace de Oracle, consulte [Uso de SSL con AWS Database Migration Service \(p. 51\)](#).

AWS DMS admite las versiones de Oracle 10g, 11g y 12c para las bases de datos locales y de instancias EC2 para las ediciones Enterprise, Standard, Standard One y Standard Two. AWS DMS es compatible con las versiones de Oracle 11g (versión 11.2.0.3.v1 y posteriores) y 12c por Amazon RDS ejemplo bases de datos para las ediciones Enterprise, Standard, Standard One y Standard Two.

Al utilizar Oracle como destino, suponemos que los datos deberían migrarse al esquema o usuario que se utiliza para la conexión de destino. Si desea migrar datos a otro esquema, tiene que utilizar la transformación de esquemas. Por ejemplo, suponga que su punto de enlace de destino se conecta con el usuario `RDSMASTER` y desea migrar desde el usuario `PERFDATA` a `PERFDATA`. En este caso, cree una transformación tal y como la siguiente.

```
{  
    "rule-type": "transformation",  
    "rule-id": "2",  
    "rule-name": "2",  
    "rule-action": "rename",  
    "rule-target": "schema",  
    "object-locator": {  
        "schema-name": "PERFDATA"  
    },  
    "value": "PERFDATA"  
}
```

También se da por hecho que cualquier asignación de espacio de tabla de índice o tabla existente se mantiene en el destino si el punto de enlace de origen también es Oracle. Si el punto de enlace de origen es distinto de Oracle, utilizamos los espacios de tabla de índice y de tabla en el destino. Si desea migrar tablas e índices a distintos espacios de nombres de tabla e índice, utilice una transformación de espacio de tabla para hacerlo. Por ejemplo, supongamos que tiene un conjunto de tablas en el esquema `INVENTORY` asignado a algunos espacios de tabla en el origen de Oracle. Para la migración, desea asignar todas estas tablas a un único espacio de tabla `INVENTORYSPACE` en el destino. En este caso, cree una transformación tal y como la siguiente.

```
{  
    "rule-type": "transformation",  
    "rule-id": "3",  
    "rule-name": "3",  
    "rule-action": "rename",  
    "rule-target": "table-tablespace",  
    "object-locator": {  
        "schema-name": "INVENTORY",  
        "table-name": "%",  
        "table-tablespace-name": "%"   
    },  
    "value": "INVENTORYSPACE"  
}
```

Para obtener más información sobre transformaciones, consulte [Especificación de selección de tabla y transformaciones mediante correspondencia de tabla con JSON \(p. 288\)](#).

Para obtener más información sobre cómo trabajar con las bases de datos de Oracle como destino para AWS DMS, consulte las secciones siguientes:

#### Temas

- [Restricciones al uso de Oracle como destino para AWS Database Migration Service \(p. 160\)](#)
- [Privilegios de la cuenta de usuario necesarios para utilizar Oracle como destino \(p. 160\)](#)
- [Configuración de una base de datos de Oracle como destino para AWS Database Migration Service \(p. 161\)](#)
- [Atributos de conexión adicionales al usar Oracle como destino para AWS DMS \(p. 162\)](#)
- [Tipos de datos de destino para Oracle \(p. 162\)](#)

## Restricciones al uso de Oracle como destino para AWS Database Migration Service

Las restricciones al utilizar Oracle como destino para la migración de datos son las siguientes:

- AWS DMS no crea ningún esquema en la base de datos de Oracle de destino. Usted tiene que crear los esquemas que desee en la base de datos de Oracle de destino. El nombre de esquema ya tiene que existir para el destino de Oracle. Las tablas del esquema de origen se importan al usuario o esquema que AWS DMS utiliza para conectarse a la instancia de destino. Para migrar varios esquemas, cree varias tareas de replicación.
- AWS DMS no admite la opción `use direct path full load` para tablas con `INDEXTYPE CONTEXT`. Como alternativa, puede utilizar la carga de matriz.
- Con la opción de aplicación optimizada por lotes, la carga en la tabla de cambios netos utiliza una ruta directa, que no admite el tipo XML. Como alternativa, puede utilizar el modo de aplicación transaccional.

## Privilegios de la cuenta de usuario necesarios para utilizar Oracle como destino

Para utilizar un destino de Oracle en una tarea de AWS Database Migration Service, conceda los siguientes privilegios en la base de datos Oracle. Puede concederlos a la cuenta de usuario especificada en las definiciones de la base de datos Oracle para AWS DMS.

- `SELECT ANY TRANSACTION`
- `SELECT on V$NLS_PARAMETERS`
- `SELECT on V$TIMEZONE_NAMES`
- `SELECT on ALL_INDEXES`
- `SELECT on ALL_OBJECTS`
- `SELECT on DBA_OBJECTS`
- `SELECT on ALL_TABLES`
- `SELECT on ALL_USERS`
- `SELECT on ALL_CATALOG`
- `SELECT on ALL_CONSTRAINTS`
- `SELECT on ALL_CONS_COLUMNS`
- `SELECT on ALL_TAB_COLS`
- `SELECT on ALL_IND_COLUMNS`
- `DROP ANY TABLE`
- `SELECT ANY TABLE`
- `INSERT ANY TABLE`

- UPDATE ANY TABLE
- CREATE ANY VIEW
- DROP ANY VIEW
- CREATE ANY PROCEDURE
- ALTER ANY PROCEDURE
- DROP ANY PROCEDURE
- CREATE ANY SEQUENCE
- ALTER ANY SEQUENCE
- DROP ANY SEQUENCE

Para los requisitos especificados a continuación, conceda los privilegios adicionales con nombre:

- Para utilizar una lista de tablas específica, otorgue SELECT y ALTER en cualquier tabla replicada.
- Para permitir a un usuario crear una tabla en un espacio de tabla predeterminado, conceda el privilegio GRANT UNLIMITED TABLESPACE.
- Para el inicio de sesión, conceda el privilegio CREATE SESSION.
- Si está utilizando una ruta de acceso directa, conceda el privilegio LOCK ANY TABLE.
- Si se ha seleccionado la opción "DROP and CREATE table" o "TRUNCATE before loading" en la configuración de carga completa y el esquema de tabla de destino difiere del esquema para el usuario de AWS DMS, conceda el privilegio DROP ANY TABLE.
- Para almacenar los cambios en las tablas de modificaciones o una tabla de auditoría cuando el esquema de tablas de destino es distinto del esquema para el usuario de AWS DMS, conceda los privilegios CREATE ANY TABLE y CREATE ANY INDEX.

## Privilegios de lectura necesarios para AWS Database Migration Service en la base de datos de destino

La cuenta de usuario de AWS DMS debe tener permisos de lectura para las siguientes tablas de administradores de bases de datos:

- SELECT on DBA\_USERS
- SELECT on DBA\_TAB\_PRIVS
- SELECT on DBA\_OBJECTS
- SELECT on DBA\_SYNONYMS
- SELECT on DBA\_SEQUENCES
- SELECT on DBA\_TYPES
- SELECT on DBA\_INDEXES
- SELECT on DBA\_TABLES
- SELECT en DBA\_TRIGGERS

Si alguno de los privilegios necesarios no se puede conceder a V\$xxx, concédalos a V\_\$.xxx.

## Configuración de una base de datos de Oracle como destino para AWS Database Migration Service

Antes de utilizar una base de datos de Oracle como destino de migración de datos, debe proporcionar una cuenta de usuario de Oracle a AWS DMS. La cuenta de usuario debe disponer de privilegios de lectura y escritura en la base de datos de Oracle, según se especifica en [Privilegios de la cuenta de usuario necesarios para utilizar Oracle como destino \(p. 160\)](#).

## Atributos de conexión adicionales al usar Oracle como destino para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar el destino de Oracle. Especifica esta configuración al crear el punto de enlace de destino. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

La siguiente tabla muestra los atributos de conexión adicionales disponibles cuando se utiliza Oracle como destino.

Nombre	Descripción
<code>useDirectPathFullLoad</code>	Utilice la carga completa de la ruta de acceso directa. Especifíquelo para habilitar o deshabilitar el protocolo de ruta de acceso directa en Oracle Cloud Infrastructure (OCI) para la carga masiva de tablas de Oracle.  Valor predeterminado: Y  Valores válidos: Y/N  Ejemplo: <code>useDirectPathFullLoad=N</code>
<code>charLengthSemantics</code>	La semántica para la longitud de la columna especifica si la longitud de una columna se encuentra en bytes o en caracteres. Ajuste este valor en CHAR.  Ejemplo: <code>charLengthSemantics=CHAR</code>

## Tipos de datos de destino para Oracle

Las bases de datos Oracle que se utilizan con AWS DMS admiten la mayoría de los tipos de datos de Oracle. La siguiente tabla muestra los tipos de datos de destino de Oracle que se admiten cuando se utiliza AWS DMS y la asignación predeterminada desde los tipos de datos de AWS DMS. Para obtener más información sobre cómo ver el tipo de datos que se asigna desde el origen, consulte la sección para el origen que esté utilizando.

Tipo de datos de AWS DMS	Tipos de datos de Oracle
BOOLEANO	NUMBER (1)
BYTES	RAW (longitud)
FECHA	DATETIME
TIME	TIMESTAMP (0)
DATETIME	TIMESTAMP (escala)
INT1	NUMBER (3)
INT2	NUMBER (5)
INT4	NUMBER (10)
INT8	NUMBER (19)

Tipo de datos de AWS DMS	Tipos de datos de Oracle
NUMERIC	NUMBER (p,s)
REAL4	FLOAT
REAL8	FLOAT
STRING	Con indicación de fecha: DATE Con indicación de tiempo: TIMESTAMP Con indicación de marca de tiempo: TIMESTAMP Con indicación de timestamp_with_timezone: TIMESTAMP WITH TIMEZONE Con indicación de timestamp_with_local_timezone: TIMESTAMP WITH LOCAL TIMEZONE Con indicación de interval_year_to_month: INTERVAL YEAR TO MONTH Con indicación de interval_day_to_second: INTERVAL DAY TO SECOND Si longitud > 4000: CLOB En el resto de casos: VARCHAR2 (longitud)
UINT1	NUMBER (3)
UINT2	NUMBER (5)
UINT4	NUMBER (10)
UINT8	NUMBER (19)
WSTRING	Si longitud > 2000: NCLOB En el resto de casos: NVARCHAR2 (longitud)
BLOB	BLOB Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de BLOB para una tarea en particular. Los tipos de datos BLOB se admiten únicamente en las tablas que incluyen una clave principal
CLOB	CLOB Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de CLOB para una tarea en particular. En la CDC, los tipos de datos CLOB se admiten únicamente en las tablas que incluyen una clave principal.
NCLOB	NCLOB Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de NCLOB para una tarea en particular. En la CDC, los tipos de datos NCLOB se admiten únicamente en las tablas que incluyen una clave principal.

Tipo de datos de AWS DMS	Tipos de datos de Oracle
XMLTYPE	<p>El tipo de datos de destino XMLTYPE solo es pertinente en tareas de replicación de Oracle a Oracle.</p> <p>Cuando la base de datos de origen sea Oracle, los tipos de datos de origen se replican tal cual en el destino de Oracle. Por ejemplo, un tipo de datos XMLTYPE en el origen se crea como un tipo de datos XMLTYPE en el destino.</p>

## Uso de una base de datos de Microsoft SQL Server como destino para AWS Database Migration Service

Puede migrar datos a bases de datos de Microsoft SQL Server mediante AWS DMS. Con una base de datos de SQL Server como destino, podrá migrar datos desde otra base de datos de SQL Server o desde una de las bases de datos compatibles.

Para las bases de datos locales y de instancias Amazon EC2, AWS DMS admite como destino las versiones de SQL Server 2005, 2008, 2008R2, 2012, 2014, 2016 y 2017 para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.

Para las bases de datos de instancias de Amazon RDS, AWS DMS admite como destino las versiones de SQL Server 2008R2, 2012, 2014, 2016 y 2017 para las ediciones Enterprise, Standard, Workgroup y Developer. Las ediciones Web y Express no son compatibles.

Para obtener más información sobre cómo trabajar con AWS DMS y las bases de datos de destino de SQL Server, consulte lo siguiente.

### Temas

- [Restricciones al uso de SQL Server como destino para AWS Database Migration Service \(p. 164\)](#)
- [Requisitos de seguridad al utilizar SQL Server como destino para AWS Database Migration Service \(p. 165\)](#)
- [Atributos de conexión adicionales al usar SQL Server como destino para AWS DMS \(p. 165\)](#)
- [Tipos de datos de destino para Microsoft SQL Server \(p. 166\)](#)

## Restricciones al uso de SQL Server como destino para AWS Database Migration Service

Cuando se utiliza una base de datos de SQL Server como destino para AWS DMS, se aplican las siguientes restricciones:

- Al crear manualmente una tabla de destino de SQL Server con una columna calculada, no se admite la replicación de carga completa al utilizar la utilidad de copia en masa de BCP. Para utilizar la replicación de carga completa, deshabilite la opción Use BCP for loading tables en la pestaña Advanced de la consola. Para obtener más información sobre cómo trabajar con BCP, consulte la [documentación de Microsoft SQL Server](#).
- Al replicar tablas con SQL Server y tipos de datos espaciales (GEOMETRY y GEOGRAPHY), AWS DMS reemplaza cualquier identificador de referencia espacial (SRID) que podría haber insertado con el SRID predeterminado. El SRID predeterminado es 0 para GEOMETRY y 4326 para GEOGRAPHY.
- No se permite usar tablas temporales. La migración de tablas temporales podría funcionar con una tarea de solo replicación en modo de aplicación transaccional si dichas tablas se crean manualmente en el destino.

## Requisitos de seguridad al utilizar SQL Server como destino para AWS Database Migration Service

A continuación, se describen los requisitos de seguridad para utilizar AWS DMS con Microsoft SQL Server como destino:

- La cuenta de usuario de AWS DMS debe tener al menos el rol de usuario `db_owner` en la base de datos de SQL Server a la que se está conectando.
- Un administrador del sistema de SQL Server deberá proporcionar este permiso a todas las cuentas de usuario de AWS DMS.

## Atributos de conexión adicionales al usar SQL Server como destino para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar su destino de SQL Server. Especifica esta configuración al crear el punto de enlace de destino. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

La siguiente tabla muestra los atributos de conexión adicionales que puede utilizar cuando SQL Server es el destino.

Nombre	Descripción
<code>useBCPFullLoad</code>	<p>Use este atributo para transferir los datos de las operaciones de carga completa con BCP. Cuando la tabla de destino contiene una columna de identidad que no existe en la tabla de origen, debe desactivar la opción Use BCP for loading table (Usar BCP para cargar la tabla).</p> <p>Valor predeterminado: Y</p> <p>Valores válidos: Y/N</p> <p>Ejemplo: <code>useBCPFullLoad=Y</code></p>
<code>BCPPacketSize</code>	<p>El tamaño máximo de los paquetes (en bytes) que se utiliza para transferir datos con BCP.</p> <p>Valor predeterminado: 16384</p> <p>Valores válidos: 1-100 000</p> <p>Ejemplo: <code>BCPPacketSize=16384</code></p>
<code>controlTablesFileGroup</code>	<p>Especifique un grupo de archivos para las tablas internas de AWS DMS. Cuando la tarea de replicación comienza, todas las tablas de control internas de AWS DMS (<code>awsdms_apply_exception</code>, <code>awsdms_apply</code>, <code>awsdms_changes</code>) se crean en el grupo de archivos especificado.</p> <p>Valor predeterminado: n/a</p> <p>Valores válidos: String</p> <p>Ejemplo: <code>controlTablesFileGroup=filegroup1</code></p>

Nombre	Descripción
	<p>A continuación se ofrece un ejemplo de un comando para la creación de un grupo de archivos.</p> <pre>ALTER DATABASE replicate ADD FILEGROUP Test1FG1; GO ALTER DATABASE replicate ADD FILE (     NAME = test1dat5,     FILENAME = 'C:\temp\DATA\t1dat5.ndf',     SIZE = 5MB,     MAXSIZE = 100MB,     FILEGROWTH = 5MB ) TO FILEGROUP Test1FG1; GO</pre>

## Tipos de datos de destino para Microsoft SQL Server

La siguiente tabla muestra los tipos de datos de destino de Microsoft SQL Server que se admiten cuando se utiliza AWS DMS y la asignación predeterminada desde los tipos de datos de AWS DMS. Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipo de datos de AWS DMS	Tipo de datos de SQL Server
BOOLEANO	TINYINT
BYTES	VARBINARY (longitud)
FECHA	Para SQL Server 2008 y versiones posteriores, utilice DATE.  Para versiones anteriores, si la escala es 3 o inferior, utilice DATETIME. En el resto de casos, utilice VARCHAR (37).
TIME	Para SQL Server 2008 y versiones posteriores, utilice DATETIME2 (%d).  Para versiones anteriores, si la escala es 3 o inferior, utilice DATETIME. En el resto de casos, utilice VARCHAR (37).
DATETIME	Para SQL Server 2008 y versiones posteriores, utilice DATETIME2 (escala).  Para versiones anteriores, si la escala es 3 o inferior, utilice DATETIME. En el resto de casos, utilice VARCHAR (37).
INT1	SMALLINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	NUMBER (p,s)
REAL4	REAL

Tipo de datos de AWS DMS	Tipo de datos de SQL Server
REAL8	FLOAT
STRING	<p>Si la columna es una columna de fecha u hora, haga lo siguiente:</p> <ul style="list-style-type: none"> <li>Para SQL Server 2008 y versiones posteriores, utilice DATETIME2.</li> <li>Para versiones anteriores, si la escala es 3 o inferior, utilice DATETIME. En el resto de casos, utilice VARCHAR (37).</li> </ul> <p>Si la columna no es una columna de fecha o de hora, utilice VARCHAR (longitud).</p>
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INT
UINT8	BIGINT
WSTRING	NVARCHAR (longitud)
BLOB	<p>VARBINARY (máx.)</p> <p>IMAGE</p> <p>Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de BLOB para una tarea en particular. AWS DMS solo es compatible con tipos de datos BLOB en tablas que incluyan una clave principal.</p>
CLOB	<p>VARCHAR (máx.)</p> <p>Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de CLOB para una tarea en particular. En la CDC, AWS DMS admite tipos de datos CLOB solo en tablas que incluyan una clave principal.</p>
NCLOB	<p>NVARCHAR (máx.)</p> <p>Para utilizar este tipo de datos con AWS DMS, será preciso habilitar el uso de NCLOB para una tarea en particular. En la CDC, AWS DMS admite tipos de datos NCLOB solo en las tablas que incluyan una clave principal.</p>

## Uso de una base de datos de PostgreSQL como destino para AWS Database Migration Service

Puede migrar datos a bases de datos de PostgreSQL mediante AWS DMS, ya sea desde otra base de datos de PostgreSQL o desde otra de las bases de datos que admite. Se admiten las versiones 9.4 y posteriores de PostgreSQL para bases de datos locales, de Amazon RDS, Amazon Aurora con compatibilidad con PostgreSQL y de instancias EC2.

AWS DMS adopta un enfoque tabla a tabla al migrar datos del origen al destino en la fase de carga completa. No es posible garantizar el orden de la tabla durante la fase de carga completa. Las tablas no estarán sincronizadas durante la fase de carga completa y mientras se estén aplicando transacciones

almacenadas en la caché para tablas individuales. Como resultado, las limitaciones de integridad referencial activas puede derivar en errores de tareas durante la fase de carga completa.

En PostgreSQL, se implementan claves externas (límites de integridad referencial) mediante disparadores. Durante la fase de carga completa, AWS DMS carga cada tabla de una en una. Recomendamos encarecidamente que deshabilite las restricciones de clave externa durante una carga completa, utilizando uno de los siguientes métodos:

- Deshabilite temporalmente todos los disparadores de la instancia y finalice la carga completa.
- Utilice el parámetro `session_replication_role` en PostgreSQL.

En cualquier momento, un disparador puede estar en uno de los siguientes estados: `origin`, `replica`, `always` o bien `disabled`. Cuando se establece el parámetro `session_replication_role` en `replica`, solo los disparadores con el estado `replica` estarán activos y se disparan cuando se llaman. De lo contrario, los disparadores permanecen inactivos.

PostgreSQL tiene un mecanismo a prueba de errores para evitar que se trunque una tabla, incluso cuando se ha establecido `session_replication_role`. Puede utilizar esto como una alternativa a la inhabilitación de disparadores para ayudar a que la carga completa se ejecute hasta su finalización. Para ello, establezca el modo de preparación de la tabla de destino en `DO NOTHING`. De lo contrario, las operaciones `DROP` y `TRUNCATE` fallan cuando hay limitaciones de clave externa.

En Amazon RDS, puede establecer este parámetro mediante un grupo de parámetros. Para una instancia PostgreSQL que se ejecute en Amazon EC2, puede establecer el parámetro directamente.

Para obtener más información sobre cómo trabajar con las bases de datos de PostgreSQL como destino para AWS DMS, consulte las secciones siguientes:

#### Temas

- [Restricciones al uso de PostgreSQL como destino para AWS Database Migration Service \(p. 168\)](#)
- [Requisitos de seguridad al utilizar una base de datos de PostgreSQL como destino para AWS Database Migration Service \(p. 168\)](#)
- [Atributos de conexión adicionales al usar PostgreSQL como destino para AWS DMS \(p. 169\)](#)
- [Tipos de datos de destino para PostgreSQL \(p. 169\)](#)

## Restricciones al uso de PostgreSQL como destino para AWS Database Migration Service

Cuando se utiliza una base de datos de PostgreSQL como destino para AWS DMS, se aplican las siguientes restricciones:

- El tipo de datos JSON se convierte al tipo de datos Native CLOB.
- En una migración de Oracle a PostgreSQL, si una columna de Oracle contiene un carácter NULL (valor hexadecimal U+0000), AWS DMS convierte dicho carácter en un espacio (valor hexadecimal U+0020). Esto se debe a una restricción de PostgreSQL.

## Requisitos de seguridad al utilizar una base de datos de PostgreSQL como destino para AWS Database Migration Service

Por motivos de seguridad, la cuenta de usuario utilizada para la migración de datos debe ser un usuario registrado en cualquier base de datos de PostgreSQL que utilice como destino.

## Atributos de conexión adicionales al usar PostgreSQL como destino para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar el destino de PostgreSQL. Especifica esta configuración al crear el punto de enlace de destino. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

La siguiente tabla muestra los atributos de conexión adicionales que puede utilizar para configurar PostgreSQL como destino para AWS DMS.

Nombre	Descripción
<code>maxFileSize</code>	Especifica el tamaño máximo (en KB) de cualquier archivo .csv que se utilice para transferir datos a PostgreSQL.  Valor predeterminado: 32768 KB (32 MB)  Valores válidos: 1–1 048 576  Ejemplo: <code>maxFileSize=512</code>
<code>executeTimeout</code>	Establece el tiempo de espera de las instrucciones del cliente para la instancia de PostgreSQL, en segundos. El valor de predeterminado es de 60 segundos.  Ejemplo: <code>executeTimeout=100</code>
<code>afterConnectScript=SET session_replication_role='replica'</code>	Añada este atributo para que AWS DMS omita todas las claves externas y los disparadores de usuario. Esta acción reduce significativamente el tiempo que se tarda en cargar datos masivamente cuando se utiliza el modo de carga completa.  Note  En un destino Amazon Aurora PostgreSQL, esta configuración de <code>afterConnectScript</code> solo surte efecto en el modo de captura de datos modificados (CDC).  Ejemplo: <code>afterConnectScript=SET session_replication_role='replica'</code>

## Típos de datos de destino para PostgreSQL

El punto de enlace de la base de datos de PostgreSQL para AWS DMS admite la mayoría de los tipos de datos de la base de datos de PostgreSQL. La siguiente tabla muestra los tipos de datos de destino de base de datos PostgreSQL que se admiten cuando se utiliza AWS DMS y la asignación predeterminada desde los tipos de datos de AWS DMS. No se admiten los tipos de datos que aparecen en la tabla siguiente.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipo de datos de AWS DMS	Tipo de datos de PostgreSQL
BOOL	BOOL

Tipo de datos de AWS DMS	Tipo de datos de PostgreSQL
BYTES	BYTEA
FECHA	FECHA
TIME	TIME
TIMESTAMP	Si la escala es de 0 a 6, utilice TIMESTAMP. Si la escala es de 7 a 9, utilice VARCHAR (37).
INT1	SMALLINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (P,S)
REAL4	FLOAT4
REAL8	FLOAT8
STRING	Si la longitud es de 1 a 21 845, utilice VARCHAR (longitud en bytes). Si la longitud es de 21 846 a 2 147 483 647, utilice VARCHAR (65535).
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
WSTRING	Si la longitud es de 1 a 21 845, utilice VARCHAR (longitud en bytes). Si la longitud es de 21 846 a 2 147 483 647, utilice VARCHAR (65535).
BCLOB	BYTEA
NCLOB	TEXT
CLOB	TEXT

#### Note

Cuando se replica desde un origen de PostgreSQL, AWS DMS crea la tabla de destino con los mismos tipos de datos para todas las columnas, además de las columnas con los tipos de datos definidos por el usuario. En estos casos, el tipo de datos se crea como de "caracteres variables" en el destino.

## Uso de una base de datos compatible con MySQL como destino para AWS Database Migration Service

Puede migrar datos a cualquier base de datos compatible con MySQL utilizando AWS DMS, desde cualquiera de los motores de datos de origen compatibles con AWS DMS. Si va a migrar a una base de

datos local compatible con MySQL, entonces AWS DMS requiere que su motor de origen resida dentro del ecosistema de AWS. El motor puede residir en un servicio administrado por Amazon como Amazon RDS, Amazon Aurora o Amazon S3. De forma alternativa, el motor puede estar en una base de datos autoadministrada en Amazon EC2.

Puede utilizar SSL para cifrar las conexiones entre su punto de enlace compatible con MySQL y la instancia de replicación. Para obtener más información acerca de cómo utilizar SSL con un punto de enlace compatible con MySQL, consulte [Uso de SSL con AWS Database Migration Service \(p. 51\)](#).

AWS DMS es compatible con las versiones 5.5, 5.6 y 5.7 de MySQL y Aurora MySQL. Además, AWS DMS admite las versiones de MariaDB desde la 10.0.24 a la 10.0.28.

Puede utilizar las siguientes bases de datos compatibles con MySQL como destinos para AWS DMS:

- MySQL Community Edition
- MySQL Standard Edition
- MySQL Enterprise Edition
- MySQL Cluster Carrier Grade Edition
- MariaDB Community Edition
- MariaDB Enterprise Edition
- Column Store de MariaDB
- Amazon Aurora MySQL

#### Note

Independientemente del motor de almacenamiento del origen (MyISAM, MEMORY, etc.), AWS DMS crea una tabla de destino compatible con MySQL como la tabla InnoDB de forma predeterminada. Si necesita disponer de una tabla que utilice un motor de almacenamiento que no sea InnoDB, puede crear manualmente la tabla en el destino compatible con MySQL y migrar la tabla en el modo "no hacer nada". Para obtener más información sobre el modo "no hacer nada", consulte [Configuración de tareas de carga completa \(p. 264\)](#).

Para obtener más información sobre cómo trabajar con las bases de datos compatibles con MySQL como destino para AWS DMS, consulte las secciones siguientes.

#### Temas

- [Uso de cualquier base de datos compatible con MySQL como destino para AWS Database Migration Service \(p. 171\)](#)
- [Restricciones en el uso de una base de datos compatible con MySQL como destino para AWS Database Migration Service \(p. 172\)](#)
- [Atributos de conexión adicionales al usar una base de datos compatible con MySQL como destino para AWS DMS \(p. 172\)](#)
- [Tipos de datos de destino para MySQL \(p. 174\)](#)

## Uso de cualquier base de datos compatible con MySQL como destino para AWS Database Migration Service

Antes de empezar a trabajar con una base de datos compatible con MySQL como destino para AWS DMS, asegúrese de que cumple los siguientes requisitos previos:

- Proporcione una cuenta de usuario para AWS DMS que tiene privilegios de lectura/escritura a la base de datos compatible con MySQL. Para crear los privilegios necesarios, ejecute los siguientes comandos.

```
CREATE USER '<user acct>'@'%' IDENTIFIED BY '<user password>';
GRANT ALTER, CREATE, DROP, INDEX, INSERT, UPDATE, DELETE, SELECT ON <schema>.* TO
'<user acct>'@'%';
GRANT ALL PRIVILEGES ON awsdms_control.* TO '<user acct>'@'%';
```

- Durante la fase de migración de carga completa, debe desactivar las claves externas en las tablas de destino. Para desactivar las comprobaciones de claves externas en una base de datos compatible con MySQL durante una carga completa, puede añadir el siguiente comando a Extra Connection Attributes (Atributos de conexión adicionales) en la sección Advanced (Avanzado) de la información de conexión del punto de enlace de destino.

```
initstmt=SET FOREIGN_KEY_CHECKS=0
```

## Restricciones en el uso de una base de datos compatible con MySQL como destino para AWS Database Migration Service

Al utilizar una base de datos MySQL como destino, AWS DMS no admite lo siguiente:

- Las instrucciones del lenguaje de definición de datos (DDL): TRUNCATE PARTITION, DROP TABLE y RENAME TABLE.
- Utilizar una instrucción ALTER TABLE <table\_name> ADD COLUMN <column\_name> para añadir columnas al inicio o en la mitad de una tabla.
- Cuando solo se actualiza la columna de LOB en una tabla de origen, AWS DMS no actualiza la columna de destino correspondiente. El LOB de destino se actualiza solamente si se actualiza al menos otra columna en la misma transacción.
- Al cargar datos a un destino compatible con MySQL en una tarea de carga completa, AWS DMS no informa errores de claves duplicadas en el registro de tareas.
- Al actualizar el valor de una columna con el valor existente, las bases de datos compatibles con MySQL devuelven una advertencia 0 rows affected. Aunque este comportamiento no es un error desde el punto de vista técnico, es diferente de la forma en que abordan la situación otros motores de base de datos. Por ejemplo, Oracle realiza una actualización de una fila. Para bases de datos compatibles con MySQL, AWS DMS genera una entrada en la tabla de control awsdms\_apply\_exceptions y registra la siguiente advertencia.

```
Some changes from the source database had no impact when applied to
the target database. See awsdms_apply_exceptions table for details.
```

## Atributos de conexión adicionales al usar una base de datos compatible con MySQL como destino para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar el destino compatible con MySQL. Especifica esta configuración al crear el punto de enlace de destino. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

En la siguiente tabla se muestran las opciones de configuración adicionales que puede utilizar al crear un destino compatible con MySQL para AWS DMS.

Nombre	Descripción
<code>targetDbType</code>	<p>Especifica dónde se migran las tablas de origen en el destino, bien en una base de datos o en varias.</p> <p>Valor predeterminado: <code>MULTIPLE_DATABASES</code></p> <p>Valores válidos: <code>{SPECIFIC_DATABASE, MULTIPLE_DATABASES}</code></p> <p>Ejemplo: <code>targetDbType=MULTIPLE_DATABASES</code></p>
<code>parallelLoadThreads</code>	<p>Mejora el desempeño cuando se cargan datos en la base de datos de destino compatible con MySQL. Especifica el número de subprocessos que se va a utilizar para cargar los datos en la base de datos de destino compatible con MySQL. Si se configuran muchos subprocessos esto puede repercutir negativamente en el desempeño de la base de datos porque se requiere una conexión independiente para cada subprocesso.</p> <p>Valor predeterminado: 1</p> <p>Valores válidos: 1-5</p> <p>Ejemplo: <code>parallelLoadThreads=1</code></p>
<code>initstmt=SET FOREIGN_KEY_CHECKS=0</code>	Desactiva las comprobaciones de claves externas.
<code>initstmt=SET time-zone</code>	<p>Especifica la zona horaria para la base de datos compatible con MySQL de destino.</p> <p>Valor predeterminado: UTC</p> <p>Valores válidos: una abreviatura de tres o cuatro caracteres para la zona horaria que desea utilizar. Los valores válidos son las abreviaturas de zona horaria estándar para el sistema operativo que aloja la base de datos compatible con MySQL de destino.</p> <p>Ejemplo: <code>initstmt=SET time_zone=UTC</code></p>
<code>afterConnectScript=SET character_set_connection='latin1'</code>	Especifica que el destino compatible con MySQL debe traducir las instrucciones recibidas al conjunto de caracteres latin1, que es el conjunto de caracteres compilado de manera predeterminada de la base de datos. Este parámetro normalmente mejora el desempeño al convertir de clientes UTF8.
<code>maxFileSize</code>	<p>Especifica el tamaño máximo (en KB) de cualquier archivo .csv que se utilice para transferir datos a una base de datos compatible con MySQL.</p> <p>Valor predeterminado: 32768 KB (32 MB)</p> <p>Valores válidos: 1 - 1048576</p> <p>Ejemplo: <code>maxFileSize=512</code></p>

Nombre	Descripción
CleanSrcMetadataOnMismatch	<p>Limpia y vuelve a crear la información de metadatos de la tabla en la instancia de replicación cuando se produce una discordancia. Por ejemplo, en una situación en la que se ejecuta una DDL modificada en la tabla podría dar como resultado información distinta sobre la tabla en caché en la instancia de replicación. Booleano.</p> <p>Valor predeterminado: <code>false</code></p> <p>Ejemplo: <code>CleanSrcMetadataOnMismatch=false</code></p>

## Tipos de datos de destino para MySQL

La siguiente tabla muestra los tipos de datos de destino de base de datos MySQL que se admiten cuando se utiliza AWS DMS y la asignación predeterminada desde los tipos de datos de AWS DMS.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de AWS DMS	Tipos de datos de MySQL
BOOLEANO	BOOLEANO
BYTES	<p>Si la longitud es de 1 a 65 535, utilice VARBINARY (longitud).</p> <p>Si la longitud es de 65 536 a 2 147 483 647, utilice LONGLOB.</p>
FECHA	FECHA
TIME	TIME
TIMESTAMP	<p>"Si la escala es =&gt; 0 y =&lt; 6, utilice DATETIME (escala)</p> <p>Si la escala es =&gt; 7 y =&lt; 9, utilice: VARCHAR (37)</p>
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE PRECISION
STRING	<p>Si la longitud es de 1 a 21 845, utilice VARCHAR (longitud).</p> <p>Si la longitud es de 21 846 a 2 147 483 647, utilice LONGTEXT.</p>

Tipos de datos de AWS DMS	Tipos de datos de MySQL
UINT1	UNSIGNED TINYINT
UINT2	UNSIGNED SMALLINT
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT
WSTRING	Si la longitud es de 1 a 32 767, utilice VARCHAR (longitud).  Si la longitud es de 32 768 a 2 147 483 647, utilice LONGTEXT.
BLOB	Si la longitud es de 1 a 65 535, utilice BLOB.  Si la longitud es de 65 536 a 2 147 483 647, utilice LONGBLOB.  Si la longitud es 0, utilice LONGBLOB (compatibilidad completa con LOB).
NCLOB	Si la longitud es de 1 a 65 535, utilice TEXT.  Si la longitud es de 65 536 a 2 147 483 647, utilice LONGTEXT con ucs2 para CHARACTER SET.  Si la longitud es 0, utilice LONGTEXT (compatibilidad completa con LOB) con ucs2 para CHARACTER SET.
CLOB	Si la longitud es de 1 a 65 535, utilice TEXT.  Si la longitud es de 65 536 a 2 147 483 647, utilice LONGTEXT.  Si la longitud es 0, utilice LONGTEXT (compatibilidad completa con LOB).

## Uso de una base de datos de Amazon Redshift como destino para AWS Database Migration Service

Puede migrar datos a bases de datos de Amazon Redshift utilizando AWS Database Migration Service. Amazon Redshift es un servicio de almacén de datos administrado de varios petabytes en la nube. Con una base de datos de Amazon Redshift como destino, puede migrar datos desde todas las demás bases de datos de origen compatibles.

El clúster de Amazon Redshift debe estar en la misma cuenta y región de AWS que la instancia de replicación.

Durante una migración de base de datos a Amazon Redshift, AWS DMS primero traslada los datos a un bucket de Amazon S3. Cuando los archivos se encuentran en un bucket de Amazon S3, AWS DMS los transfiere a las tablas adecuadas del almacén de datos de Amazon Redshift. AWS DMS crea el bucket de S3 en la misma región de AWS que la de la base de datos de Amazon Redshift. La instancia de replicación de AWS DMS debe estar en la misma región.

Si utiliza la AWS Command Line Interface (AWS CLI) o la API de DMS para migrar datos a Amazon Redshift, debe configurar un rol de AWS Identity and Access Management (IAM) para otorgarle acceso a S3. Para obtener más información sobre la creación de este rol de IAM, consulte [Creación de los roles de IAM para usar con la AWS CLI y la API de AWS DMS \(p. 38\)](#).

El punto de enlace de Amazon Redshift ofrece la total automatización de lo siguiente:

- Generación de esquemas y mapeo de tipos de datos
- Carga completa de las tablas de la base de datos de origen
- Carga gradual de los cambios realizados en las tablas de origen
- Aplicación de los cambios de esquema en lenguaje de definición de datos (DDL) realizados en la tablas de origen
- Sincronización entre los procesos de carga completa y captura de datos de cambios (CDC)

AWS Database Migration Service es compatible con las operaciones de carga completa y procesamiento de cambios. AWS DMS lee los datos de la base de datos de origen y crea una serie de archivos de valores separados por comas (.csv). Para las operaciones de carga completa, AWS DMS crea archivos para cada tabla. A continuación, AWS DMS copia los archivos de cada tabla en una carpeta separada en Amazon S3. Cuando los archivos se cargan en Amazon S3, AWS DMS envía un comando de copia y los datos de los archivos se copian en Amazon Redshift. Para cambiar las operaciones de procesamiento de cambios, AWS DMS copia los cambios netos en los archivos .csv. A continuación, AWS DMS carga los archivos de cambios netos en Amazon S3 y copia los datos en Amazon Redshift.

Para obtener más información sobre cómo trabajar con Amazon Redshift como destino para AWS DMS, consulte las secciones siguientes:

#### Temas

- [Requisitos previos para el uso de una base de datos de Amazon Redshift como destino para AWS Database Migration Service \(p. 176\)](#)
- [Restricciones al uso de Amazon Redshift como destino para AWS Database Migration Service \(p. 177\)](#)
- [Configuración de una base de datos de Amazon Redshift como destino para AWS Database Migration Service \(p. 177\)](#)
- [Uso del direccionamiento de VPC mejorado con un Amazon Redshift como destino para AWS Database Migration Service \(p. 178\)](#)
- [Creación y uso de claves de AWS KMS para cifrar datos de destino de Amazon Redshift \(p. 178\)](#)
- [Configuración de puntos de enlace cuando se utiliza Amazon Redshift como destino para AWS DMS \(p. 181\)](#)
- [Atributos de conexión adicionales al usar Amazon Redshift como destino para AWS DMS \(p. 183\)](#)
- [Tipos de datos de destino para Amazon Redshift \(p. 186\)](#)

## Requisitos previos para el uso de una base de datos de Amazon Redshift como destino para AWS Database Migration Service

En la siguiente lista se describen los requisitos previos necesarios para trabajar con Amazon Redshift como destino de la migración de datos:

- Utilice la consola de administración de AWS para lanzar un clúster de Amazon Redshift. Debe tener en cuenta la información básica acerca de su cuenta de AWS y su clúster de Amazon Redshift, como la contraseña, el nombre de usuario y el nombre de la base de datos. Necesita estos valores al crear el punto de enlace de destino de Amazon Redshift.
- El clúster de Amazon Redshift debe estar en la misma cuenta y región de AWS que la instancia de replicación.

- La instancia de replicación de AWS DMS necesita conectividad de red al punto de enlace de Amazon Redshift (nombre de host y puerto) que utiliza su clúster.
- AWS DMS utiliza un bucket de Amazon S3 para transferir datos a la base de datos de Amazon Redshift. Para que AWS DMS cree el bucket, la consola de DMS utiliza un rol de IAM, `dms-access-for-endpoint`. Si utiliza la AWS CLI o la API de DMS para crear una migración de base de datos con Amazon Redshift como la base de datos de destino, debe crear esta función de IAM. Para obtener más información sobre la creación de este rol, consulte [Creación de los roles de IAM para usar con la AWS CLI y la API de AWS DMS \(p. 38\)](#).
- AWS DMS convierte BLOB, CLOB y NCLOB a un VARCHAR en la instancia Amazon Redshift de destino. Amazon Redshift no admite los tipos de datos VARCHAR de más de 64 KB, por lo que no se pueden almacenar LOB tradicionales en Amazon Redshift.

## Restricciones al uso de Amazon Redshift como destino para AWS Database Migration Service

Al utilizar una base de datos Amazon Redshift como destino, AWS DMS no admite lo siguiente:

- El siguiente DLL no se admite:

```
ALTER TABLE <table name> MODIFY COLUMN <column name> <data type>;
```

- AWS DMS no puede migrar ni replicar cambios en un esquema con un nombre que comience por un carácter de subrayado (\_). Si tiene esquemas que tienen un nombre que comienza por un carácter de subrayado, utilice transformaciones de asignación para cambiar el nombre del esquema en el destino.
- Amazon Redshift no admite VARCHAR mayores de 64 KB. Los LOB de bases de datos tradicionales no pueden almacenarse en Amazon Redshift.

## Configuración de una base de datos de Amazon Redshift como destino para AWS Database Migration Service

AWS Database Migration Service debe configurarse para que pueda trabajar con la instancia de Amazon Redshift. En la siguiente tabla se describen las propiedades de configuración disponibles para el punto de enlace de Amazon Redshift.

Propiedad	Descripción
servidor	Nombre del clúster de Amazon Redshift que está utilizando.
port	Número de puerto para Amazon Redshift. El valor predeterminado es 5439.
nombre de usuario	Nombre de un usuario de Amazon Redshift registrado.
password	La contraseña del usuario citado en la propiedad del nombre de usuario.
base de datos	Nombre del almacén de datos de Amazon Redshift (servicio) que está utilizando.

Si desea añadir atributos adicionales de la cadena de conexión a su punto de enlace de Amazon Redshift, puede especificar los atributos `maxFileSize` y `fileTransferUploadStreams`. Para obtener más información sobre estos atributos, consulte [Atributos de conexión adicionales al usar Amazon Redshift como destino para AWS DMS \(p. 183\)](#).

## Uso del direccionamiento de VPC mejorado con un Amazon Redshift como destino para AWS Database Migration Service

Cuando utiliza Enhanced VPC Routing con su destino de Amazon Redshift, todo el tráfico COPY entre el clúster de Amazon Redshift y los repositorios de datos a través de Amazon VPC. Puesto que el Enhanced VPC Routing (Direccionamiento VPC mejorado) afecta a la forma en la que Amazon Redshift obtiene acceso a otros recursos, los comandos COPY podrían fallar si no ha configurado su VPC correctamente.

AWS DMS podría verse afectada por este comportamiento porque utiliza el comando COPY para mover datos de S3 a un clúster de Amazon Redshift.

A continuación se presentan los pasos que realiza AWS DMS para cargar datos en Amazon Redshift como destino:

1. AWS DMS copia los datos del origen a los archivos .csv en el servidor de replicación.
2. AWS DMS utiliza AWS SDK para copiar los archivos .csv en un bucket de S3 en su cuenta.
3. Por último, AWS DMS utiliza el comando COPY en Amazon Redshift para copiar los datos de los archivos .csv de S3 en una tabla adecuada en Amazon Redshift.

Si Enhanced VPC Routing no está habilitado, Amazon Redshift dirige el tráfico a través de Internet, incluido el tráfico a otros servicios dentro de la red de AWS. Si la función no está activada, no tendrá que configurar la ruta de acceso a la red. Si la función está activada, deberá crear una ruta de acceso a la red específica entre la VPC de su clúster y sus recursos de datos. Para obtener más información sobre la configuración necesaria, consulte [Enhanced VPC Routing](#) en la documentación de Amazon Redshift.

## Creación y uso de claves de AWS KMS para cifrar datos de destino de Amazon Redshift

Es posible cifrar los datos de destino enviados a Amazon S3 antes de que se copien en Amazon Redshift. Para ello, puede crear y utilizar claves de AWS Key Management Service (AWS KMS) personalizadas. Puede utilizar la clave que ha creado para cifrar los datos de destino mediante uno de los siguientes mecanismos al crear el punto de enlace de destino de Amazon Redshift:

- Utilice la opción siguiente al ejecutar el comando de `create-endpoint` utilizando la AWS CLI.

```
--redshift-settings '{"EncryptionMode": "SSE_KMS", "ServerSideEncryptionKmsKeyId": "your-kms-key-ARN"}'
```

Aquí, `your-kms-key-ARN` está el nombre de recurso de Amazon (ARN) para su clave de KMS. Para obtener más información, consulte [Configuración de puntos de enlace cuando se utiliza Amazon Redshift como destino para AWS DMS \(p. 181\)](#).

- Establezca el atributo de conexión adicional `encryptionMode` al valor `SSE_KMS` y el atributo de conexión adicional `serverSideEncryptionKmsKeyId` al ARN de su clave de KMS. Para obtener más información, consulte [Atributos de conexión adicionales al usar Amazon Redshift como destino para AWS DMS \(p. 183\)](#).

Para cifrar los datos de destino de Amazon Redshift con una clave de KMS, necesita un rol de AWS Identity and Access Management (IAM) que tenga permisos para acceder a los datos de Amazon Redshift. A continuación, se accede a este rol de IAM en una política (una política de claves) asociada a la clave de cifrado que cree. Puede hacerlo en su propia consola de IAM creando lo siguiente:

- Un rol de IAM con una política administrada por Amazon.
- Una clave de cifrado de KMS con una política de claves que hace referencia a este rol.

En los procedimientos siguientes se describe cómo hacerlo.

Para crear un rol de IAM con la política administrada de Amazon requerida

1. Abra la consola de IAM, en <https://console.aws.amazon.com/iam/>.
2. Seleccione Roles en el panel de navegación. Se abre la página Roles.
3. Elija Create role. Se abre la página Create role (Crear rol).
4. Con AWS service (Servicio de AWS) elegido como entidad de confianza, elija DMS como servicio que utiliza el rol.
5. Elija Next: Permissions. Aparece la página Attach permissions policies (Asociar políticas de permisos).
6. Busque y seleccione la política `AmazonDMSRedshiftS3Role`.
7. Elija Next: Tags (Siguiiente: Etiquetas). Aparece la página Add tags (Añadir etiquetas). A continuación, puede añadir las etiquetas que desee.
8. Elija Next: Review (Siguiiente: Revisar) y revise los resultados.
9. Si la configuración es la que necesita, introduzca un nombre para el rol (por ejemplo, `DMS-Redshift-endpoint-access-role`) y cualquier descripción adicional, a continuación, elija Create role (Crear rol). Se abre la página Roles con un mensaje que indica que el rol se ha creado.

Ya ha creado el nuevo rol para acceder a recursos de Amazon Redshift para cifrado con un nombre especificado, por ejemplo, `DMS-Redshift-endpoint-access-role`.

Para crear una clave de cifrado de KMS con una política de claves que hace referencia al rol de IAM.

1. En la consola de IAM, elija Encryption keys (Claves de cifrado) en el panel de navegación. Si es la primera vez que accede a esta opción, aparece una introducción a la AWS Key Management Service y puede elegir Get Started Now (Empezar ahora) para comenzar. De lo contrario, se abre la página Encryption keys (Claves de cifrado).
2. Elija Create key. Se abre la página Create Alias and Description (Crear alias y descripción).
3. Introduzca un alias para la clave (por ejemplo, `DMS-Redshift-endpoint-encryption-key`) y cualquier descripción adicional.
4. Elija Advanced Options. En la sección Advanced Options (Opciones avanzadas), asegúrese de elegir KMS en Key Material Origin (Origen del material de claves), a continuación, elija Next Step (Paso siguiente). Se abre la página Add tags (Añadir etiquetas).
5. Añada las etiquetas que desee para ayudar a identificar la clave y realizar un seguimiento de su uso y, a continuación, seleccione Next Step (Paso siguiente). Se abre la página Define Key Administrative Permissions (Definir permisos administrativos clave) que muestra una lista de usuarios y roles entre los que puede elegir.
6. Añada los usuarios y roles que desee para administrar la clave. Asegúrese de que estos usuarios y roles tengan los permisos necesarios para administrar la clave.
7. Elija si los administradores de claves pueden eliminar la clave, a continuación, seleccione Next Step (Paso siguiente). Se abre la página Define Key Usage Permissions (Definir permisos de uso de claves) que muestra una lista adicional de usuarios y roles entre los que puede elegir.
8. Elija el rol que creó previamente en Roles para habilitar el acceso para cifrar los datos de destino de Amazon Redshift, por ejemplo, `DMS-Redshift-endpoint-access-role`. A continuación, elija Next Step.

Se abre la página Preview Key Policy (Vista previa de política de claves), que muestra el JSON para la política de claves que puede revisar. Aquí, puede ver si la política de claves hace referencia al rol que eligió en el paso anterior. También puede ver las distintas acciones de claves permitidas para las distintas entidades principales (usuarios y roles), tal y como se muestra en el siguiente ejemplo.

```
{  
    "Id": "key-consolepolicy-3",  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "Enable IAM User Permissions",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": [  
                    "arn:aws:iam::944454115380:root"  
                ]  
            },  
            "Action": "kms:*",  
            "Resource": "*"  
        },  
        {  
            "Sid": "Allow access for Key Administrators",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": [  
                    "arn:aws:iam::944454115380:role/Admin"  
                ]  
            },  
            "Action": [  
                "kms>Create*",  
                "kmsDescribe*",  
                "kmsEnable*",  
                "kmsList*",  
                "kmsPut*",  
                "kmsUpdate*",  
                "kmsRevoke*",  
                "kmsDisable*",  
                "kmsGet*",  
                "kmsDelete*",  
                "kmsTagResource",  
                "kmsUntagResource",  
                "kmsScheduleKeyDeletion",  
                "kmsCancelKeyDeletion"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Sid": "Allow use of the key",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": [  
                    "arn:aws:iam::944454115380:role/DMS-Redshift-endpoint-access-role"  
                ]  
            },  
            "Action": [  
                "kmsEncrypt",  
                "kmsDecrypt",  
                "kmsReEncrypt*",  
                "kmsGenerateDataKey*",  
                "kmsDescribeKey"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Sid": "Allow attachment of persistent resources",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": [  
                    "arn:aws:iam::944454115380:role/DMS-Redshift-endpoint-access-role"  
                ]  
            }  
        }  
    ]  
}
```

```
        ],
    },
    "Action": [
        "kms>CreateGrant",
        "kms>ListGrants",
        "kms>RevokeGrant"
    ],
    "Resource": "*",
    "Condition": {
        "Bool": {
            "kms:GrantIsForAWSResource": true
        }
    }
}
]
```

9. Elija Finalizar. Se abre la página Encryption keys (Claves de cifrado) con un mensaje que indica que se ha creado la clave de cifrado maestra.

Ahora ha creado una nueva clave de KMS con un alias especificado (por ejemplo, DMS-Redshift-endpoint-encryption-key). Esta clave permite a AWS DMS cifrar los datos de destino de Amazon Redshift.

#### Note

También puede crear y administrar las claves de KMS utilizando el AWS Key Management Service con los roles y políticas de IAM. Para obtener más información, consulte la [Guía para desarrolladores de Key Management Service \(KMS\)](#).

## Configuración de puntos de enlace cuando se utiliza Amazon Redshift como destino para AWS DMS

Puede utilizar la configuración de punto de enlace para configurar su destino de Amazon Redshift de forma similar al uso de atributos de conexión adicionales. Puede especificar esta configuración al crear el punto de enlace de destino mediante el comando `create-endpoint` en la AWS CLI, con la opción `--redshift-settings "json-settings"`. Aquí, `json-settings` tiene un objeto JSON que contiene parámetros para especificar la configuración. También puede especificar un archivo .json que contiene el mismo objeto `json-settings`, por ejemplo, como en el siguiente: `--redshift-settings file:///your-file-path/my_redshift_settings.json`. Aquí, `my_redshift_settings.json` es el nombre de un archivo .json que contiene el mismo `json-settings` object.

Tenga en cuenta que los nombres de parámetro para configuración de punto de enlace son los mismos que los nombres para atributos de conexión adicionales equivalentes, excepto que los nombres de parámetro para la configuración de punto de enlace tienen mayúsculas iniciales. Tenga en cuenta también que no toda la configuración de puntos de enlace de destino de Amazon Redshift que utilizan atributos de conexión adicionales están disponibles mediante la opción `--redshift-settings` del comando `create-endpoint`. Para obtener más información acerca de la configuración disponible en una llamada de AWS CLI a `create-endpoint`, consulte [create-endpoint](#) en la AWS CLI Command Reference para AWS DMS. Para obtener más información acerca de esta configuración, consulte los atributos de conexión adicionales equivalentes en [Atributos de conexión adicionales al usar Amazon Redshift como destino para AWS DMS \(p. 183\)](#).

Puede utilizar la configuración de puntos de enlace de destino de Amazon Redshift para configurar lo siguiente:

- Una clave de cifrado de datos de AWS KMS personalizada. A continuación puede utilizar esta clave para cifrar los datos de destino enviados a Amazon S3 antes de que se copien en Amazon Redshift.

- Un bucket de S3 personalizado como almacenamiento intermedio para datos migrados a Amazon Redshift.

## Configuración de clave de KMS para cifrado de datos

Los siguientes ejemplos muestran cómo configurar una clave de KMS personalizada para cifrar los datos que se envíen a S3. Para comenzar, podría realizar la siguiente llamada a `create-endpoint` en la AWS CLI.

```
aws dms create-endpoint --endpoint-identifier redshift-target-endpoint --endpoint-type target  
--engine-name redshift --username your-username --password your-password  
--server-name your-server-name --port 5439 --database-name your-db-name  
--redshift-settings '{"EncryptionMode": "SSE_KMS",  
"ServerSideEncryptionKmsKeyId": "arn:aws:kms:us-east-1:944454115380:key/24c3c5a1-f34a-4519-a85b-2debbeff226d1"}'
```

Aquí, el objeto JSON especificado por la opción `--redshift-settings` define dos parámetros. Uno es un parámetro `EncryptionMode` con el valor `SSE_KMS`. El otro es un parámetro `ServerSideEncryptionKmsKeyId` con el valor `arn:aws:kms:us-east-1:944454115380:key/24c3c5a1-f34a-4519-a85b-2debbeff226d1`. Este valor es un nombre de recurso de Amazon (ARN) para su clave de KMS personalizada.

De forma predeterminada, el cifrado de datos de S3 se realiza utilizando el cifrado del lado del servidor de S3. Para el destino de Amazon Redshift del ejemplo anterior, esto es también equivalente a especificar la configuración de su punto de enlace, como se indica en el siguiente ejemplo.

```
aws dms create-endpoint --endpoint-identifier redshift-target-endpoint --endpoint-type target  
--engine-name redshift --username your-username --password your-password  
--server-name your-server-name --port 5439 --database-name your-db-name  
--redshift-settings '{"EncryptionMode": "SSE_S3"}'
```

Para obtener más información sobre el trabajo con el cifrado del lado del servidor de S3, consulte [Protección de datos utilizando el cifrado del lado del servidor](#).

## Configuración de bucket de Amazon S3

Al migrar datos en un punto de enlace de destino de Amazon Redshift, AWS DMS utiliza un bucket de Amazon S3 predeterminado como almacenamiento intermedio de tarea antes de copiar los datos migrados a Amazon Redshift. Por ejemplo, los ejemplos que se muestran para crear un punto de enlace de destino de Amazon Redshift con una clave de cifrado de datos de KMS utilizan este bucket de S3 predeterminado (consulte [Configuración de clave de KMS para cifrado de datos \(p. 182\)](#)).

En su lugar, puede especificar un bucket de S3 personalizado para este almacenamiento intermedio incluyendo los siguientes parámetros en el valor de la opción `--redshift-settings` en el comando de la AWS CLI `create-endpoint`:

- `BucketName`: una cadena que especifica como el nombre del almacenamiento de bucket de S3.
- `BucketFolder`: (opcional) una cadena que puede especificar como nombre de la carpeta de almacenamiento en el bucket de S3 especificado.
- `ServiceAccessRoleArn`: el ARN de un rol de IAM que permite acceso administrativo al bucket de S3. Normalmente, crea este rol en función de la política `AmazonDMSRedshiftS3Role`. Para ver un ejemplo, consulte el procedimiento para crear un rol de IAM con la política administrada de Amazon requerida en [Creación y uso de claves de AWS KMS para cifrar datos de destino de Amazon Redshift \(p. 182\)](#).

#### Note

Si especifica el ARN de un rol de IAM distinto utilizando la opción `--service-access-role-arn` del comando `create-endpoint`, esta opción de rol de IAM tiene prioridad.

El ejemplo siguiente muestra cómo podría utilizar estos parámetros para especificar un bucket de Amazon S3 personalizado en la siguiente llamada `create-endpoint` mediante la AWS CLI.

```
aws dms create-endpoint --endpoint-identifier redshift-target-endpoint --endpoint-type target  
--engine-name redshift --username your-username --password your-password  
--server-name your-server-name --port 5439 --database-name your-db-name  
--redshift-settings '{"ServiceAccessRoleArn": "your-service-access-ARN",  
"BucketName": "your-bucket-name", "BucketFolder": "your-bucket-folder-name"}'
```

## Atributos de conexión adicionales al usar Amazon Redshift como destino para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar el destino de Amazon Redshift. Especifica esta configuración al crear el punto de enlace de destino. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

La siguiente tabla muestra los atributos de conexión adicionales disponibles cuando se utiliza Amazon Redshift como destino.

Nombre	Descripción
<code>maxFileSize</code>	Especifica el tamaño máximo (en KB) de cualquier archivo .csv que se utilice para transferir datos a Amazon Redshift.  Valor predeterminado: 32768 KB (32 MB)  Valores válidos: 1-1048576  Ejemplo: <code>maxFileSize=512</code>
<code>fileTransferUploadStreams</code>	Especifica el número de subprocesos utilizados para cargar un único archivo.  Valor predeterminado: 10  Valores válidos: 1 a 64  Ejemplo: <code>fileTransferUploadStreams=20</code>
<code>acceptanydate</code>	Especifica si se acepta cualquier formato de fecha, incluidos los formatos de fecha no válidos como 0000-00-00. Valor booleano.  Valor predeterminado: false  Valores válidos: true   false  Ejemplo: <code>acceptanydate=true</code>
<code>dateformat</code>	Especifica el formato de fecha. Es una entrada de cadena que está vacía de forma predeterminada. El formato

Nombre	Descripción
	<p>predeterminado es YYYY-MM-DD, pero puede cambiarlo, por ejemplo, a DD-MM-YYYY. Si utiliza otros formatos para los valores de fecha, use el argumento <code>auto</code> con el parámetro <code>dateformat</code>. El argumento <code>auto</code> reconoce varios formatos que no se admiten cuando se utiliza una cadena <code>dateformat</code>. La palabra clave <code>auto</code> distingue entre mayúsculas y minúsculas.</p> <p>Valor predeterminado: vacío</p> <p>Los valores válidos son: '<code>dateformat_string</code>' o <code>auto</code></p> <p>Ejemplo: <code>dateformat=auto</code></p>
<code>timeformat</code>	<p>Especifica el formato de hora. Es una entrada de cadena que está vacía de forma predeterminada. El argumento <code>auto</code> reconoce varios formatos que no se admiten cuando se utiliza una cadena <code>timeformat</code>. Si utiliza valores de fecha y hora con formatos distintos entre sí, use el argumento <code>auto</code> con el parámetro <code>timeformat</code>.</p> <p>Valor predeterminado: 10</p> <p>Valores válidos: '<code>timeformat_string</code>'   '<code>auto</code>'   '<code>epochsecs</code>'   '<code>epochmillisecs</code>'</p> <p>Ejemplo: <code>timeformat=auto</code></p>
<code>emptyasnull</code>	<p>Especifica si AWS DMS debería migrar los campos de tipo CHAR y VARCHAR vacíos como null. El valor <code>true</code> establece en null los campos CHAR y VARCHAR vacíos.</p> <p>Valor predeterminado: <code>false</code></p> <p>Valores válidos: <code>true</code>   <code>false</code></p> <p>Ejemplo: <code>emptyasnull=true</code></p>
<code>truncateColumns</code>	<p>Trunca los datos de las columnas al número adecuado de caracteres de modo que se ajuste a la especificación de la columna. Aplica solo para las columnas con un tipo de datos CHAR o VARCHAR y filas de 4 MB de tamaño, o menos.</p> <p>Valor predeterminado: <code>false</code></p> <p>Valores válidos: <code>true</code>   <code>false</code></p> <p>Ejemplo:</p> <pre><code>truncateColumns=true;</code></pre>

Nombre	Descripción
<code>removeQuotes</code>	<p>Elimina las comillas circundantes de las cadenas en los datos entrantes. Todos los caracteres dentro de las comillas, incluidos los delimitadores, se conservan. Para obtener más información acerca de la eliminación de comillas en un destino de Amazon Redshift, consulte la <a href="#">Amazon Redshift Database Developer Guide</a>.</p> <p>Valor predeterminado: false</p> <p>Valores válidos: true   false</p> <p>Ejemplo:</p> <pre><code>removeQuotes=true;</code></pre>
<code>trimBlanks</code>	<p>Elimina los caracteres de espacio en blanco del final de una cadena VARCHAR. Este parámetro aplica solo para las columnas con un tipo de datos VARCHAR.</p> <p>Valor predeterminado: false</p> <p>Valores válidos: true   false</p> <p>Ejemplo:</p> <pre><code>trimBlanks=false;</code></pre>
<code>encryptionMode</code>	<p>Especifica el modo de cifrado del lado del servidor que desea utilizar para enviar los datos a S3 antes de que se copien en Amazon Redshift. Los valores válidos son SSE_S3 (cifrado del lado del servidor de S3) o SSE_KMS (cifrado de clave de KMS). Si elige SSE_KMS, establezca el parámetro <code>serverSideEncryptionKmsKeyId</code> en el nombre de recurso de Amazon (ARN) para la clave de KMS que se va a utilizar para cifrado.</p> <p>Valor predeterminado: SSE_S3</p> <p>Valores válidos: SSE_S3 o SSE_KMS</p> <p>Ejemplo:<code>encryptionMode=SSE_S3;</code></p>
<code>serverSideEncryptionKmsKeyId</code>	<p>Si establece <code>encryptionMode</code> en SSE_KMS, establezca este parámetro en el ARN para la clave de KMS. Puede encontrar este ARN seleccionando el alias de clave en la lista de claves de KMS creadas para su cuenta. Al crear la clave, debe asociar políticas y roles específicos a la misma. Para obtener más información, consulte <a href="#">Creación y uso de claves de AWS KMS para cifrar datos de destino de Amazon Redshift (p. 178)</a>.</p> <p>Ejemplo:</p> <pre><code>serverSideEncryptionKmsKeyId=arn:aws:kms:us-east-1:944454115380:key/24c3c5a1-f34a-4519-a85b-2debbe226d1;</code></pre>

## Tipos de datos de destino para Amazon Redshift

El punto de enlace de Amazon Redshift para AWS DMS es compatible con la mayoría de los tipos de datos de Amazon Redshift. La siguiente tabla muestra los tipos de datos de destino de Amazon Redshift que se admiten cuando se utiliza AWS DMS y la asignación predeterminada desde los tipos de datos de AWS DMS.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de AWS DMS	Tipos de datos de Amazon Redshift
BOOLEANO	BOOL
BYTES	VARCHAR (longitud)
FECHA	FECHA
TIME	VARCHAR(20)
DATETIME	Si la escala es => 0 y =< 6, utilice: TIMESTAMP (s)  Si la escala es => 7 y =< 9, utilice: VARCHAR (37)
INT1	INT2
INT2	INT2
INT4	INT4
INT8	INT8
NUMERIC	Si la escala es => 0 y =< 37, utilice: NUMERIC (p,s)  Si la escala es => 38 y =< 127, utilice: VARCHAR (longitud)
REAL4	FLOAT4
REAL8	FLOAT8
STRING	Si la longitud es de 1 a 65 535, utilice VARCHAR (longitud en bytes)  Si la longitud es de 65 536 a 2 147 483 647, utilice VARCHAR (65535)
UINT1	INT2
UINT2	INT2
UINT4	INT4
UINT8	NUMERIC (20,0)

Tipos de datos de AWS DMS	Tipos de datos de Amazon Redshift
WSTRING	Si la longitud es de 1 a 65 535, utilice NVARCHAR (longitud en bytes)  Si la longitud es de 65 536 a 2 147 483 647, utilice NVARCHAR (65535)
BLOB	VARCHAR (longitud máxima del LOB *2)  El tamaño de LOB máximo no puede superar 31 KB. Amazon Redshift no admite VARCHAR mayores de 64 KB.
NCLOB	NVARCHAR (longitud máxima del LOB)  El tamaño de LOB máximo no puede superar 63 KB. Amazon Redshift no admite VARCHAR mayores de 64 KB.
CLOB	VARCHAR (longitud máxima del LOB)  El tamaño de LOB máximo no puede superar 63 KB. Amazon Redshift no admite VARCHAR mayores de 64 KB.

## Uso de una base de datos de SAP ASE como destino para AWS Database Migration Service

Puede migrar datos a SAP Adaptive Server Enterprise (ASE) (anteriormente conocida como Sybase) utilizando AWS DMS desde cualquiera de las bases de datos de origen admitidas.

Se admiten las versiones 15, 15.5, 15.7, 16 y posteriores de SAP ASE.

### Requisitos previos para utilizar una base de datos de SAP ASE como destino para AWS Database Migration Service

Antes de empezar a trabajar con una base de datos de SAP ASE como destino para AWS DMS, asegúrese de que cumple los siguientes requisitos previos:

- Debe proporcionar acceso a la cuenta de SAP ASE al usuario de AWS DMS. Este usuario debe tener privilegios de lectura/escritura en la base de datos de SAP ASE.
- Cuando se vaya a hacer la replicación en la versión 15.7 de SAP ASE instalada en una instancia EC2 de Windows que esté configurada en caracteres no latinos (por ejemplo, el chino), AWS DMS requiere que se instale SAP ASE 15.7 SP121 en la máquina de destino de SAP ASE.

### Atributos de conexión adicionales al usar SAP ASE como destino para AWS DMS

Puede utilizar atributos de conexión adicionales para configurar el destino de SAP ASE. Especifica esta configuración al crear el punto de enlace de destino. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

La siguiente tabla muestra los atributos de conexión adicionales disponibles cuando se utiliza SAP ASE como destino:

Nombre	Descripción
<code>enableReplication</code>	Establezca el valor en <code>y</code> para habilitar automáticamente la replicación de SAP ASE. Esto solo es necesario si la replicación de SAP ASE no está ya habilitada.
<code>additionalConnectionProperties</code>	Los parámetros de conexión ODBC adicionales que quiere especificar.

#### Note

Si el nombre de usuario o contraseña especificada en la cadena de conexión contiene caracteres no latinos (por ejemplo, chinos), será necesario utilizar la propiedad siguiente: `charset=gb18030`

## Tipos de datos de destino para SAP ASE

La siguiente tabla muestra los tipos de datos de destino de base de datos SAP ASE que se admiten cuando se utiliza AWS DMS y la asignación predeterminada desde los tipos de datos de AWS DMS.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipos de datos de AWS DMS	Tipos de datos de SAP ASE
BOOLEANO	BIT
BYTES	VARBINARY (longitud)
FECHA	FECHA
TIME	TIME
TIMESTAMP	Si la escala es => 0 y <= 6, utilice BIGDATETIME Si la escala es => 7 y <= 9, utilice: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	NUMERIC (p,s)
REAL4	REAL
REAL8	DOUBLE PRECISION
STRING	VARCHAR (longitud)
UINT1	TINYINT
UINT2	UNSIGNED SMALLINT

Tipos de datos de AWS DMS	Tipos de datos de SAP ASE
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT
WSTRING	VARCHAR (longitud)
BLOB	IMAGE
CLOB	UNIEXT
NCLOB	TEXT

AWS DMS no admite tablas que incluyan campos con los siguientes tipos de datos. Las columnas que se repliquen con estos tipos de datos aparecen con valor NULL.

- Tipos definidos por el usuario (UDT)

## Uso de Amazon S3 como destino para AWS Database Migration Service

Puede migrar datos a Amazon S3 utilizando AWS DMS desde cualquiera de las bases de datos de origen admitidas. Al utilizar Amazon S3 como destino en una tarea de AWS DMS, tanto los datos de la carga completa como los de la captura de datos de cambios (CDC) se escriben con el formato de valores separados por comas (.csv) de forma predeterminada. Para opciones de almacenamiento más compacto y consulta más rápida, también tiene la opción de escribir los datos en formato Apache Parquet (.parquet).

Los nombres de archivo de AWS DMS creados durante una carga completa utilizando un contador hexadecimal incremental: por ejemplo: LOAD00001.csv, LOAD00002..., LOAD00009, LOAD0000A, etc. para archivos .csv. AWS DMS nombra los archivos CDC utilizando marcas temporales, por ejemplo, 20141029-1134010000.csv. Para cada tabla de origen, AWS DMS crea una carpeta bajo la carpeta de destino especificada. AWS DMS escribe todos los archivos de carga completa y CDC en el bucket de Amazon S3 especificado.

El parámetro `bucketFolder` contiene la ubicación en la que se almacenan los archivos .csv o .parquet antes de que se carguen en el bucket de S3. Con archivos .csv, los datos de la tabla se almacenan en el siguiente formato en el bucket de S3.

```
<schema_name>/<table_name>/LOAD0000001.csv
<schema_name>/<table_name>/LOAD0000002.csv
...
<schema_name>/<table_name>/LOAD0000009.csv
<schema_name>/<table_name>/LOAD000000A.csv
<schema_name>/<table_name>/LOAD000000B.csv
...
<schema_name>/<table_name>/LOAD000000F.csv
<schema_name>/<table_name>/LOAD0000010.csv
...
```

Puede especificar el delimitador de columnas, el delimitador de filas y otros parámetros mediante los atributos de conexión adicionales. Para obtener más información acerca de los atributos de conexión adicionales, consulte [Atributos de conexión adicionales al usar Amazon S3 como destino para AWS DMS \(p. 202\)](#) al final de esta sección.

Al utilizar AWS DMS para replicar cambios de datos, la primera columna del archivo de salida .csv o .parquet cómo se cambiaron los datos como se muestra para el siguiente archivo .csv.

```
I,101,Smith,Bob,4-Jun-14,New York
U,101,Smith,Bob,8-Oct-15,Los Angeles
U,101,Smith,Bob,13-Mar-17,Dallas
D,101,Smith,Bob,13-Mar-17,Dallas
```

En este ejemplo, suponga que hay una tabla **EMPLOYEE** en la base de datos de origen. AWS DMS escribe datos en el archivo .csv o .parquet, en respuesta a los siguientes eventos:

- Un nuevo empleado (Bob Smith, ID de 101) es contratado el 4 de junio de 2014 en la oficina de Nueva York. En el archivo .csv o .parquet, la **I** de la primera columna indica que se ha insertado (**INSERT**) en la tabla del **EMPLEADO** en la base de datos de origen.
- El 8 de octubre de 15, se transfiere a Bob a la oficina de Los Ángeles. En el archivo .csv o .parquet, la **U** indica que la fila correspondiente de la tabla del **EMPLEADO** se ha actualizado (**UPDATE**) para reflejar la nueva ubicación de la oficina de Bob. El resto de la línea refleja la fila en la tabla del **EMPLEADO** tal y como aparece después de la actualización (**UPDATE**).
- El 13 de marzo de 2017, vuelven a transferir a Bob a la oficina de Dallas. En el archivo .csv o .parquet, la **U** indica que esta fila se ha actualizado de nuevo (con **UPDATE**). El resto de la línea refleja la fila en la tabla del **EMPLEADO** tal y como aparece después de la actualización (**UPDATE**).
- Después de trabajar en Dallas durante un tiempo, Bob se marcha de la empresa. En el archivo .csv o .parquet, la **D** indica que la fila se eliminó (con **DELETE**) en la tabla de origen. El resto de la línea refleja cómo aparecía la fila en la tabla del **EMPLEADO** antes de eliminarla.

## Requisitos previos para utilizar Amazon S3 como destino

Antes de utilizar Amazon S3 como destino, compruebe que se cumplen las siguientes condiciones:

- El bucket de S3 que está utilizando como destino está en la misma región que la instancia de replicación DMS que utiliza para la migración de los datos.
- La cuenta de AWS que utiliza para la migración tiene un rol de IAM con acceso de escritura y eliminación en el bucket de S3 que utiliza como destino.
- Este rol tiene acceso de etiquetado por lo que puede etiquetar cualquier objeto de S3 escrito en el bucket de destino.

Para configurar el acceso de esta cuenta, asegúrese de que el rol asignado a la cuenta de usuario utilizada para crear la tarea de migración tenga el siguiente conjunto de permisos.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:DeleteObject",
                "s3:PutObjectTagging"
            ],
            "Resource": [
                "arn:aws:s3:::buckettest2*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3>ListBucket"
```

```
        ],
        "Resource": [
            "arn:aws:s3:::buckettest2*"
        ]
    }
}
```

## Restricciones al uso de Amazon S3 como destino

Al archivo de Amazon S3 que utiliza como destino se le aplican las siguientes restricciones:

- Solo se admiten los siguientes comandos de lenguaje de definición de datos (DDL): TRUNCATE TABLE, DROP TABLE y CREATE TABLE.
- No se admite el modo LOB completo.
- No se admiten cambios en la estructura de la tabla de origen durante la carga completa. Si se admiten cambios en los datos durante la carga completa.
- Varias tareas que replican los datos de la misma tabla de origen al mismo bucket de punto de enlace de S3 de destino tiene como consecuencia que esas tareas escriban en el mismo archivo. Le recomendamos que especifique diferentes puntos de enlace de destino (buckets) si el origen de datos es de la misma tabla.

## Seguridad

Para utilizar Amazon S3 como destino, la cuenta utilizada para la migración debe tener acceso de escritura y de eliminación al bucket de Amazon S3 que se utiliza como destino. Especifique el nombre de recurso de Amazon (ARN) de un rol de IAM que tenga los permisos necesarios para acceder a Amazon S3.

AWS DMS admite un conjunto de concesiones predefinidas para Amazon S3, conocidas como listas de control de acceso (ACL) predefinidas. Cada ACL predefinida tiene un conjunto de beneficiarios y permisos que puede utilizar para configurar permisos para el bucket de Amazon S3. Puede especificar una ACL predefinida utilizando `cannedAclForObjects` en el atributo de la cadena de conexión para el punto de enlace de destino de S3. Para obtener más información acerca del atributo de conexión adicional `cannedAclForObjects`, consulte [Atributos de conexión adicionales al usar Amazon S3 como destino para AWS DMS \(p. 202\)](#). Para obtener más información sobre las ACL predefinidas de Amazon S3, consulte [ACL predefinidas](#).

El rol de IAM que utilice para la migración tiene que ser capaz de realizar la operación de la API `s3:PutObjectAcl`.

## Uso de Apache Parquet para almacenar objetos de Amazon S3

El formato de valores separados por comas (.csv) es el formato de almacenamiento predeterminado para los objetos de destino de Amazon S3. Para un almacenamiento más compacto y consultas más rápidas, puede utilizar en su lugar Apache Parquet (.parquet) como formato de almacenamiento.

Apache Parquet es un formato de almacenamiento de archivos de código abierto diseñado originalmente para Hadoop. Para obtener más información en Apache Parquet, consulte <https://parquet.apache.org/>.

Para definir .parquet como formato de almacenamiento para los objetos de destino de S3, puede utilizar los siguientes mecanismos:

- La configuración de punto de enlace que proporcione como parámetros de un objeto JSON al crear el punto de enlace mediante la AWS CLI o la API para AWS DMS. Para obtener más información, consulte [Configuración de puntos de enlace cuando se utiliza Amazon S3 como destino para AWS DMS \(p. 200\)](#).

- Atributos de conexión adicionales que proporciona como una lista separada por puntos y coma al crear el punto de enlace. Para obtener más información, consulte [Atributos de conexión adicionales al usar Amazon S3 como destino para AWS DMS \(p. 202\)](#).

## Etiquetado de objetos de Amazon S3

Puede etiquetar objetos de Amazon S3 que una instancia de replicación crea especificando objetos JSON adecuados como parte de las reglas de mapeo de tabla de tareas. Para obtener más información sobre requisitos y opciones para etiquetado de objetos de S3, incluidos nombre de etiqueta válidos, consulte [Etiquetado de objetos](#) en la Guía para desarrolladores de Amazon Simple Storage Service. Para obtener más información sobre el mapeo de tablas utilizando JSON, consulte [Especificación de selección de tabla y transformaciones mediante correspondencia de tabla con JSON \(p. 288\)](#).

Etiquete los objetos de S3 creados para tablas y esquemas especificados mediante uno o varios objetos JSON del tipo de regla `selection`. A continuación, siga este objeto (u objetos) `selection` mediante uno o varios objetos JSON del tipo de regla `post-processing` con acción `add-tag`. Estas reglas de post-procesamiento identifican los objetos de S3 que desea etiquetar y especifican los nombres y los valores de las etiquetas que desea añadir a estos objetos de S3.

Puede encontrar los parámetros para especificar en objetos JSON del tipo de regla `post-processing` en la siguiente tabla.

Parámetro	Valores posibles	Descripción
<code>rule-type</code>	<code>post-processing</code>	Un valor que aplica acciones de posprocesamiento a los objetos de destino generados. Puede especificar una o varias reglas de posprocesamiento para etiquetar objetos de S3 seleccionados.
<code>rule-id</code>	Un valor numérico.	Un único valor numérico para identificar la regla.
<code>rule-name</code>	Un valor alfanumérico.	Un nombre exclusivo para identificar la regla.
<code>rule-action</code>	<code>add-tag</code>	La acción de posprocesamiento que desea aplicar al objeto de S3. Puede añadir una o varias etiquetas mediante un único objeto de posprocesamiento JSON para la acción <code>add-tag</code> .
<code>object-locator</code>	<code>schema-name</code> : el nombre del esquema de tabla.  <code>table-name</code> : el nombre de la tabla.	El nombre de cada esquema y tabla al que se aplica la regla. Puede utilizar el símbolo de porcentaje "%" como carácter comodín para la totalidad o parte del valor de cada parámetro <code>object-locator</code> . Así, puede hacer coincidir estos elementos: <ul style="list-style-type: none"> <li>• Una sola tabla en un esquema único</li> <li>• Una sola tabla en varios o todos los esquemas</li> <li>• Algunas o todas las tablas en un esquema único</li> </ul>

Parámetro	Valores posibles	Descripción
tag-set	<p>key: cualquier nombre válido para una etiqueta individual.</p> <p>value: cualquier valor JSON válido para esta etiqueta.</p>	<ul style="list-style-type: none"> <li>Algunas o todas las tablas en algunos o todos los esquemas</li> </ul> <p>Los nombres y los valores de una o varias etiquetas que desea establecer en cada objeto de S3 creado que coincida con el <code>object-locator</code> especificado. Puede especificar hasta 10 pares clave-valor en un único objeto de parámetro <code>tag-set</code>. Para obtener más información sobre el etiquetado de objetos de S3, consulte <a href="#">Etiquetado de objetos</a> en la Guía para desarrolladores de Amazon Simple Storage Service.</p> <p>También puede especificar un valor dinámico para todo o parte del valor para los parámetros <code>key</code> y <code>value</code> de una etiqueta utilizando <code>#{dyn-value}</code>. Aquí, <code>#{dyn-value}</code> puede ser <code>#{schema-name}</code> o <code>#{table-name}</code>. De este modo, puede insertar el nombre del esquema o tabla seleccionado actualmente como valor del parámetro completo o parcial.</p> <p><b>Note</b></p> <p><b>Important</b></p> <p>Si inserta un valor dinámico para el parámetro <code>key</code>, puede generar etiquetas con nombres duplicados para un objeto de S3, en función de cómo lo utilice. En este caso, solo uno de los ajustes de etiqueta duplicados se añade al objeto.</p>

Al especificar varios tipos de reglas `post-processing` para etiquetar una selección de objetos de S3, cada objeto de S3 se etiqueta utilizando solo un objeto `tag-set` de una regla de posprocesamiento. El conjunto particular de etiquetas usado para etiquetar un determinado objeto de S3 es el de la regla de posprocesamiento cuyo localizador de objeto asociado coincide mejor con dicho objeto de S3.

Por ejemplo, supongamos que dos reglas de posprocesamiento identifican el mismo objeto de S3. Supongamos también que el localizador de objeto de una regla utiliza comodines y el localizador de objeto de la otra regla utiliza una coincidencia exacta para identificar el objeto de S3 (sin comodines). En este caso, se utiliza el conjunto de etiquetas asociado a la regla de posprocesamiento con la coincidencia exacta para etiquetar el objeto de S3. Si varias reglas de posprocesamiento coinciden con un objeto de S3 dado igual de bien, se utiliza para etiquetar el conjunto de etiquetas asociado con la primera regla de posprocesamiento.

### Example Añadir etiquetas estáticas a un objeto de S3 creado para una única tabla y esquema

La siguiente selección y reglas de posprocesamiento añaden tres etiquetas (`tag_1`, `tag_2` y `tag_3` con los valores estáticos correspondientes `value_1`, `value_2` y `value_3`) a un objeto de S3 creado. Este objeto de S3 corresponde a única tabla en el origen denominada `STOCK` con un esquema denominado `aat2`.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "5",  
            "rule-name": "5",  
            "object-locator": {  
                "schema-name": "aat2",  
                "table-name": "STOCK"  
            },  
            "rule-action": "include"  
        },  
        {  
            "rule-type": "post-processing",  
            "rule-id": "41",  
            "rule-name": "41",  
            "rule-action": "add-tag",  
            "object-locator": {  
                "schema-name": "aat2",  
                "table-name": "STOCK",  
            },  
            "tag-set": [  
                {  
                    "key": "tag_1",  
                    "value": "value_1"  
                },  
                {  
                    "key": "tag_2",  
                    "value": "value_2"  
                },  
                {  
                    "key": "tag_3",  
                    "value": "value_3"  
                }  
            ]  
        }  
    ]  
}
```

### Example Añadir etiquetas estáticas y dinámicas a objetos de S3 creados para varias tablas y esquemas

El siguiente ejemplo tiene una selección y dos reglas de posprocesamiento, donde la entrada del origen incluye todas las tablas y todos sus esquemas.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "1",  
            "rule-name": "1",  
            "object-locator": {  
                "schema-name": "%",  
                "table-name": "%"  
            },  
            "rule-action": "include"  
        },  
        {  
            "rule-type": "post-processing",  
            "rule-id": "42",  
            "rule-name": "42",  
            "rule-action": "add-tag",  
            "object-locator": {  
                "schema-name": "%",  
                "table-name": "%"  
            },  
            "tag-set": [  
                {  
                    "key": "tag_1",  
                    "value": "value_1"  
                },  
                {  
                    "key": "tag_2",  
                    "value": "value_2"  
                }  
            ]  
        }  
    ]  
}
```

```

},
{
    "rule-type": "post-processing",
    "rule-id": "21",
    "rule-name": "21",
    "rule-action": "add-tag",
    "object-locator": {
        "schema-name": "%",
        "table-name": "%",
    },
    "tag-set": [
        {
            "key": "dw-schema-name",
            "value": "${schema-name}"
        },
        {
            "key": "dw-schema-table",
            "value": "my_prefix_${table-name}"
        }
    ]
},
{
    "rule-type": "post-processing",
    "rule-id": "41",
    "rule-name": "41",
    "rule-action": "add-tag",
    "object-locator": {
        "schema-name": "aat",
        "table-name": "ITEM",
    },
    "tag-set": [
        {
            "key": "tag_1",
            "value": "value_1"
        },
        {
            "key": "tag_2",
            "value": "value_2"
        }
    ]
}
]
}

```

La primera regla de posprocesamiento añade dos etiquetas (`dw-schema-name` y `dw-schema-table`) con valores dinámicos correspondientes ( `${schema-name}`  y  `my_prefix_${table-name}` ) para casi todos los objetos de S3 creados en el destino. La excepción es el objeto de S3 identificado y etiquetado con la segunda regla de posprocesamiento. De este modo, cada objeto de S3 de destino identificado por el localizador de objeto comodín se crea con etiquetas que identifican el esquema y la tabla a la que corresponde en el origen.

La segunda regla de posprocesamiento añade `tag_1` y `tag_2` con los valores estáticos correspondientes `value_1` y `value_2` a un objeto de S3 creado que se identifica mediante un localizador de objeto de coincidencia exacta. Este objeto de S3 creado corresponde por tanto a la única tabla en el origen denominada `ITEM` con un esquema denominado `aat`. Debido a la coincidencia exacta, estas etiquetas reemplazan a cualquier etiquetas de este objeto añadida a partir de la primera regla de posprocesamiento, que coincide con objetos de S3 solo por el comodín.

#### Example Añadir nombres y valores de etiqueta dinámicos a objetos de S3

El siguiente ejemplo tiene dos reglas de selección y una regla de posprocesamiento. Aquí, la entrada del origen incluye solo la tabla `ITEM` en el esquema `retail` o `wholesale`.

```
{
}
```

```

"rules": [
    {
        "rule-type": "selection",
        "rule-id": "1",
        "rule-name": "1",
        "object-locator": {
            "schema-name": "retail",
            "table-name": "ITEM"
        },
        "rule-action": "include"
    },
    {
        "rule-type": "selection",
        "rule-id": "1",
        "rule-name": "1",
        "object-locator": {
            "schema-name": "wholesale",
            "table-name": "ITEM"
        },
        "rule-action": "include"
    },
    {
        "rule-type": "post-processing",
        "rule-id": "21",
        "rule-name": "21",
        "rule-action": "add-tag",
        "object-locator": {
            "schema-name": "%",
            "table-name": "ITEM",
        },
        "tag-set": [
            {
                "key": "dw-schema-name",
                "value": "${schema-name}"
            },
            {
                "key": "dw-schema-table",
                "value": "my_prefix_ITEM"
            },
            {
                "key": "${schema-name}_ITEM_tag_1",
                "value": "value_1"
            },
            {
                "key": "${schema-name}_ITEM_tag_2",
                "value": "value_2"
            }
        ]
    }
]
}

```

La etiqueta definida para la regla de posprocesamiento añade dos etiquetas (`dw-schema-name` y `dw-schema-table`) a todos los objetos de S3 creados para la tabla `ITEM` en el destino. La primera etiqueta tiene el valor dinámico `"${schema-name}"` y la segunda etiqueta tiene un valor estático `"my_prefix_ITEM"`. De este modo, cada objeto de S3 de destino se crea con etiquetas que identifican el esquema y la tabla a la que corresponde en el origen.

Además, el conjunto de etiquetas añade dos etiquetas adicionales con nombres dinámicos `(${schema-name}_ITEM_tag_1` y `(${schema-name}_ITEM_tag_2")`. Estos tienen los valores estáticos correspondientes `value_1` y `value_2`. Por lo tanto, cada una de estas etiquetas se denomina según el esquema actual, `retail` o `wholesale`. No se puede crear un nombre de etiqueta dinámico duplicado en este objeto, ya que cada objeto se crea para un solo nombre de esquema único. El nombre de esquema se utiliza para crear un nombre de etiqueta único por lo demás.

## Creación de claves de AWS KMS para cifrar objetos de destino de Amazon S3

Puede crear y utilizar claves de AWS KMS personalizadas para cifrar sus objetos de destino de Amazon S3. Despu s de crear una clave de KMS adecuada, puede utilizar esta clave para cifrar los objetos mediante uno de los siguientes mecanismos al crear el punto de enlace de S3:

- Utilice las siguientes opciones para objetos de destino de S3 (con el formato de almacenamiento de archivos .csv predeterminado) al ejecutar el comando `create-endpoint` mediante la AWS CLI.

```
--s3-settings '{"ServiceAccessRoleArn": "your-service-access-ARN",  
"CsvRowDelimiter": "\n", "CsvDelimiter": ",", "BucketFolder": "your-bucket-folder",  
"BucketName": "your-bucket-name", "EncryptionMode": "SSE_KMS",  
"ServerSideEncryptionKmsKeyId": "your-KMS-key-ARN"}'
```

Aqu , `your-KMS-key-ARN` est  el nombre de recurso de Amazon (ARN) para su clave de KMS. Para obtener m s informaci n, consulte [Configuraci n de puntos de enlace cuando se utiliza Amazon S3 como destino para AWS DMS \(p. 200\)](#).

- Establezca el atributo de conexi n adicional `encryptionMode` al valor `SSE_KMS` y el atributo de conexi n adicional `serverSideEncryptionKmsKeyId` al ARN de su clave de KMS. Para obtener m s informaci n, consulte [Atributos de conexi n adicionales al usar Amazon S3 como destino para AWS DMS \(p. 202\)](#).

Para cifrar objetos de destino de S3 utilizando una clave de KMS, necesita un rol de IAM que tenga permisos para acceder al bucket de Amazon S3. A continuaci n, se accede a este rol de IAM en una pol tica (una pol tica de claves) asociada a la clave de cifrado que cree. Puede hacerlo en su propia consola de IAM creando lo siguiente:

- Una pol tica con permisos para acceder al bucket de Amazon S3.
- Un rol de IAM con esta pol tica.
- Una clave de cifrado de KMS con una pol tica de claves que hace referencia a este rol.

En los procedimientos siguientes se describe c mo hacerlo.

Para crear una pol tica de IAM con permisos para acceder al bucket de Amazon S3

1. Abra la consola de IAM, en <https://console.aws.amazon.com/iam/>.
2. En el panel de navegaci n, elija Policies (Pol ticas). Se abre la p gina Policies (Pol ticas).
3. Elija Create Policy. Se abre la p gina Crear pol tica.
4. Elija Service (Servicio) y, a continuaci n, S3. Aparece una lista de permisos deacci n.
5. Elija Expand all (Ampliar todo) para ampliar la lista y elegir los siguientes permisos como m nimo:
  - ListBucket
  - PutObject
  - DeleteObject

Elija cualquier otro permiso que necesite y, a continuaci n, elija Collapse all (Contraer todo) para contraer la lista.

6. Elija Resources (Recursos) para especificar los recursos a los que desea acceder. Como m nimo, elija All resources (Todos los recursos) para proporcionar acceso general a recursos de S3.
7. A ada cualquier otra condici n o permiso que necesite, a continuaci n, elija Review policy (Revisar pol tica). Compruebe los resultados en la p gina Review policy (Revisar pol tica).

8. Si la configuración es la que necesita, introduzca un nombre para la política (por ejemplo, DMS-S3-endpoint-access) y cualquier descripción, a continuación, elija Create policy (Crear política). Se abre la página Policies (Políticas) con un mensaje que indica que se ha creado la política.
9. Busque y seleccione el nombre de la política en la lista Policies (Políticas). Aparece la página Summary (Resumen) que muestra JSON para la política similar al siguiente.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "VisualEditor0",  
            "Effect": "Allow",  
            "Action": [  
                "s3:PutObject",  
                "s3>ListBucket",  
                "s3>DeleteObject"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

Ya ha creado la nueva política para acceder a recursos de S3 para cifrado con un nombre especificado, por ejemplo, DMS-S3-endpoint-access.

#### Para crear un rol de IAM con esta política

1. En la consola de IAM, elija Roles en el panel de navegación. Se abre la página de detalle Roles.
2. Elija Create role. Se abre la página Create role (Crear rol).
3. Con el servicio de AWS seleccionado como entidad de confianza, elija DMS como servicio que utilizará el rol.
4. Elija Next: Permissions. La vista Attach permissions policies (Asociar políticas de permisos) aparece en la página Create role (Crear rol).
5. Busque y seleccione la política de roles que ha creado en el procedimiento anterior (DMS-S3-endpoint-access).
6. Elija Next: Tags (Siguiente: Etiquetas). Aparece la vista Add tags (Añadir etiquetas) en la página Create role (Crear rol). A continuación, puede añadir las etiquetas que desee.
7. Elija Next: Review. Aparece la vista Review (Revisar) en la página Create role (Crear rol). Aquí, puede verificar los resultados.
8. Si la configuración es la que necesita, introduzca un nombre para el rol (requerido, por ejemplo, DMS-S3-endpoint-access-role) y cualquier descripción adicional, a continuación, elija Create role (Crear rol). Se abre la página de detalle Roles con un mensaje que indica que el rol se ha creado.

Ya ha creado el nuevo rol para acceder a recursos de S3 para cifrado con un nombre especificado, por ejemplo, DMS-S3-endpoint-access-role.

#### Para crear una clave de cifrado de KMS con una política de claves que hace referencia al rol de IAM.

1. En la consola de IAM, elija Encryption keys (Claves de cifrado) en el panel de navegación. Si es la primera vez que accede a esta opción, aparece una introducción a la AWS Key Management Service y puede elegir Get Started Now (Empezar ahora) para comenzar. De lo contrario, se abre la página Encryption keys (Claves de cifrado).
2. Elija Create key. Se abre la página Create Alias and Description (Crear alias y descripción).

3. Introduzca un alias para la clave (por ejemplo, DMS-S3-endpoint-encryption-key) y cualquier descripción adicional.
4. Elija Advanced Options. En la sección Advanced Options (Opciones avanzadas), asegúrese de elegir KMS en Key Material Origin (Origen del material de claves), a continuación, elija Next Step (Paso siguiente). Se abre la página Add tags (Añadir etiquetas).
5. Añada las etiquetas que desee para ayudar a identificar la clave y realizar un seguimiento de su uso y, a continuación, seleccione Next Step (Paso siguiente). Se abre la página Define Key Administrative Permissions (Definir permisos administrativos clave) que muestra una lista de usuarios y roles entre los que puede elegir.
6. Añada los usuarios y roles que desee para administrar la clave. Asegúrese de que estos usuarios y roles tengan los permisos necesarios para administrar la clave.
7. Elija si los administradores de claves pueden eliminar la clave, a continuación, seleccione Next Step (Paso siguiente). Se abre la página Define Key Usage Permissions (Definir permisos de uso de claves) que muestra una lista adicional de usuarios y roles entre los que puede elegir.
8. Elija el rol que creó previamente en Roles para habilitar el acceso para cifrar los objetos de destino de S3, por ejemplo DMS-S3-endpoint-access-role. A continuación, elija Next Step.

Se abre la página Preview Key Policy (Vista previa de política de claves), que muestra el JSON para la política de claves que puede revisar. Aquí, puede ver si la política de claves hace referencia al rol que eligió en el paso anterior. También puede ver las distintas acciones de claves permitidas para las distintas entidades principales (usuarios y roles), tal y como se muestra en el siguiente ejemplo.

```
{  
  "Id": "key-consolepolicy-3",  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "Enable IAM User Permissions",  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": [  
          "arn:aws:iam::944454115380:root"  
        ]  
      },  
      "Action": "kms:*",  
      "Resource": "*"  
    },  
    {  
      "Sid": "Allow access for Key Administrators",  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": [  
          "arn:aws:iam::944454115380:role/Admin"  
        ]  
      },  
      "Action": [  
        "kms>Create*",  
        "kmsDescribe*",  
        "kmsEnable*",  
        "kmsList*",  
        "kmsPut*",  
        "kmsUpdate*",  
        "kmsRevoke*",  
        "kmsDisable*",  
        "kmsGet*",  
        "kmsDelete*",  
        "kmsTagResource",  
        "kmsUntagResource",  
        "kmsScheduleKeyDeletion",  
        "kmsCancelKeyDeletion"  
      ],  
    }  
  ]  
}
```

```
        "Resource": "*"
},
{
    "Sid": "Allow use of the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::944454115380:role/DMS-S3-endpoint-access-role"
        ]
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
},
{
    "Sid": "Allow attachment of persistent resources",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::944454115380:role/DMS-S3-endpoint-access-role"
        ]
    },
    "Action": [
        "kms>CreateGrant",
        "kms>ListGrants",
        "kms:RevokeGrant"
    ],
    "Resource": "*",
    "Condition": {
        "Bool": {
            "kms:GrantIsForAWSResource": true
        }
    }
}
]
```

- Elija Finalizar. Se abre la página Encryption keys (Claves de cifrado) con un mensaje que indica que se ha creado la clave de cifrado maestra.

Ahora ha creado una nueva clave de KMS con un alias especificado (por ejemplo, DMS-S3-endpoint-encryption-key). Esta clave permite a AWS DMS cifrar los objetos de destino de S3.

#### Note

También puede crear y administrar las claves de KMS utilizando el AWS Key Management Service con los roles y políticas de IAM. Para obtener más información, consulte la [Guía para desarrolladores de Key Management Service \(KMS\)](#).

## Configuración de puntos de enlace cuando se utiliza Amazon S3 como destino para AWS DMS

Puede utilizar la configuración de punto de enlace para configurar su destino de Amazon S3 de forma similar al uso de atributos de conexión adicionales. Puede especificar esta configuración al crear el punto de enlace de destino mediante el comando `create-endpoint` en la AWS CLI, con la opción `--s3-settings 'json-settings'`. Aquí, `json-settings` tiene un objeto JSON que contiene parámetros para especificar la configuración. También puede especificar un archivo .json que contiene el mismo objeto `json-settings`, por ejemplo, como en el siguiente: `--s3-settings file:///your-file-path/`

`my_s3_settings.json`. Aquí, `my_s3_settings.json` es el nombre de un archivo .json que contiene el mismo `json-settings` object.

Tenga en cuenta que los nombres de parámetro para configuración de punto de enlace son los mismos que los nombres para atributos de conexión adicionales equivalentes, excepto que los nombres de parámetro para la configuración de punto de enlace tiene mayúsculas iniciales. Tenga en cuenta también que no toda la configuración de puntos de enlace de destino de S3 que utilizan atributos de conexión adicionales están disponibles mediante la opción `--s3-settings` del comando `create-endpoint`. Para obtener más información acerca de la configuración disponible en una llamada de AWS CLI a `create-endpoint`, consulte [create-endpoint](#) en la AWS CLI Command Reference para AWS DMS. Para obtener más información acerca de esta configuración, consulte los atributos de conexión adicionales equivalentes en [Atributos de conexión adicionales al usar Amazon S3 como destino para AWS DMS \(p. 202\)](#).

Puede utilizar la configuración de puntos de enlace de destino de S3 para configurar lo siguiente:

- Una clave de KMS personalizada para cifrar los objetos de destino de S3
- Archivos Parquet como formato de almacenamiento para objetos de destino de S3

## Configuración de clave de KMS para cifrado de datos

Los siguientes ejemplos muestran cómo configurar una clave de KMS personalizada para cifrar los objetos de destino de S3. Para comenzar, podría realizar la siguiente llamada a `create-endpoint` en la AWS CLI.

```
aws dms create-endpoint --endpoint-identifier s3-target-endpoint --engine-name s3 --  
endpoint-type target  
--s3-settings '{"ServiceAccessRoleArn": "your-service-access-ARN", "CsvRowDelimiter":  
"\n",  
"CsvDelimiter": ",", "BucketFolder": "your-bucket-folder",  
"BucketName": "your-bucket-name",  
"EncryptionMode": "SSE_KMS",  
"ServerSideEncryptionKmsKeyId": "arn:aws:kms:us-  
east-1:944454115380:key/72abb6fb-1e49-4ac1-9aed-c803dfcc0480"}'
```

Aquí, el objeto JSON especificado por la opción `--s3-settings` define dos parámetros. Uno es un parámetro `EncryptionMode` con el valor `SSE_KMS`. El otro es un parámetro `ServerSideEncryptionKmsKeyId` con el valor `arn:aws:kms:us-east-1:944454115380:key/72abb6fb-1e49-4ac1-9aed-c803dfcc0480`. Este valor es un nombre de recurso de Amazon (ARN) para su clave de KMS personalizada. En el caso de un destino de S3, también se especifica la configuración adicional. Identifican el rol de acceso del servidor, proporcionan delimitadores para el formato de almacenamiento de objetos CSV predeterminado y proporcionan la ubicación y el nombre del bucket para almacenar objetos de destino de S3.

De forma predeterminada, el cifrado de datos de S3 se realiza utilizando el cifrado del lado del servidor de S3. Para el destino de S3 del ejemplo anterior, esto es también equivalente a especificar la configuración de su punto de enlace, como se indica en el siguiente ejemplo.

```
aws dms create-endpoint --endpoint-identifier s3-target-endpoint --engine-name s3 --  
endpoint-type target  
--s3-settings '{"ServiceAccessRoleArn": "your-service-access-ARN", "CsvRowDelimiter":  
"\n",  
"CsvDelimiter": ",", "BucketFolder": "your-bucket-folder",  
"BucketName": "your-bucket-name",  
"EncryptionMode": "SSE_S3"}'
```

Para obtener más información sobre el trabajo con el cifrado del lado del servidor de S3, consulte [Protección de datos utilizando el cifrado del lado del servidor](#).

## Configuración para utilizar archivos .parquet para almacenar objetos de destino de S3

El formato predeterminado para la creación de objetos de destino de S3 son archivos .csv. Los ejemplos siguientes muestran algunos ajustes de puntos de enlace para especificar archivos .parquet como formato para crear objetos de destino de S3. Puede especificar el formato de los archivos .parquet con todos los valores predeterminados, como en el ejemplo siguiente.

```
aws dms create-endpoint --endpoint-identifier s3-target-endpoint --engine-name s3 --  
endpoint-type target  
--s3-settings '{"ServiceAccessRoleArn": "your-service-access-ARN", "DataFormat":  
"parquet"}'
```

Aquí, el parámetro DataFormat se ha definido en parquet para habilitar el formato con todos los valores predeterminados de S3. Estos valores predeterminados incluyen una codificación de diccionario ("EncodingType: "rle-dictionary") que utiliza una combinación de RLE y paquetes de bits para almacenar los valores repetidos con mayor eficiencia.

Puede añadir ajustes adicionales para otras opciones distintas a los predeterminados como en el ejemplo siguiente.

```
aws dms create-endpoint --endpoint-identifier s3-target-endpoint --engine-name s3 --  
endpoint-type target  
--s3-settings '{"ServiceAccessRoleArn": "your-service-access-ARN", "BucketFolder": "your-  
bucket-folder",  
"BucketName": "your-bucket-name", "compressionType": "GZIP", "DataFormat": "parquet",  
"EncodingType: "plain-dictionary", "dictPageSizeLimit": "3,072,000",  
"EnableStatistics": "false"}'
```

Aquí, además de parámetros para varias opciones estándar de bucket de S3 y el parámetro DataFormat, se definen los siguientes parámetros adicionales del archivo .parquet:

- EncodingType: definido en una codificación de diccionario (plain-dictionary) que almacena los valores encontrados en cada columna en un fragmento por columna de la página del diccionario.
- dictPageSizeLimit: definido en un tamaño máximo de página de diccionario de 3 MB.
- EnableStatistics: deshabilita el valor predeterminado que permite la recopilación de estadísticas sobre páginas de archivo Parquet y grupos de filas.

## Atributos de conexión adicionales al usar Amazon S3 como destino para AWS DMS

Puede especificar las siguientes opciones como atributos de conexión adicionales. Si tiene varios valores de atributo de conexión, sepárelos entre sí mediante punto y coma sin espacios en blanco adicionales.

Opción	Descripción
addColumnName	Un parámetro opcional al establecer en true o y que puede usar para añadir información del nombre de la columna al archivo de salida .csv.  Valor predeterminado: false  Valores válidos: true, false, y, n  Ejemplo:  addColumnName=true;

Opción	Descripción
bucketFolder	<p>Parámetro opcional para definir un nombre de carpeta en el bucket de S3. Si se facilitan, los objetos de destino se crean como archivos .parquet o .csv en la ruta <code>bucketFolder/schema_name/table_name/</code>. Si no se especifica este parámetro, la ruta utilizada es <code>schema_name/table_name/</code>.</p> <p>Ejemplo:</p> <pre>bucketFolder=testFolder;</pre>
bucketName	<p>El nombre del bucket de S3 donde los objetos de destino S3 se crean como archivos .csv o.parquet.</p> <p>Ejemplo:</p> <pre>bucketName=buckettest;</pre>
cannedAclForObjects	<p>Valor que permite que AWS DMS especifique una lista de control de acceso predefinida para objetos creados en el bucket de S3 como archivos .csv o .parquet. Para obtener más información sobre las ACL predefinidas de Amazon S3, consulte <a href="#">ACL predefinidas</a> en la Guía para desarrolladores de Amazon S3.</p> <p>Valor predeterminado: NINGUNO</p> <p>Los valores válidos para este atributo son: NONE; PRIVATE; PUBLIC_READ; PUBLIC_READ_WRITE; AUTHENTICATED_READ; AWS_EXEC_READ; BUCKET_OWNER_READ; BUCKET_OWNER_FULL_CONTROL.</p> <p>Ejemplo:</p> <pre>cannedAclForObjects=PUBLIC_READ;</pre>

Opción	Descripción
cdcInsertsOnly	<p>Un parámetro opcional durante una carga de captura de datos de cambios (CDC) para escribir solo operaciones INSERT en archivos de salida de valores separados por comas (.csv) o almacenamiento en columnas (.parquet). De forma predeterminada (el valor <code>false</code>), el primer campo de un registro .csv o .parquet contiene la letra I (INSERT), U (UPDATE) o D (DELETE) para indicar si la fila se insertó, actualizó o eliminó en la base de datos de origen para una carga CDC en el destino. Si <code>cdcInsertsOnly</code> se establece en <code>true</code> o <code>y</code>, solo las operaciones INSERT de la base de datos de origen se migran al archivo .csv o.parquet.</p> <p>Solo para el formato .csv, la forma en que se registran estas operaciones INSERT depende del valor de <code>includeOpForFullLoad</code>. Si <code>includeOpForFullLoad</code> está establecido en <code>true</code>, el primer campo de cada registro CDC se establece en I para indicar la operación INSERT en el origen. Si <code>includeOpForFullLoad</code> se establece en <code>false</code>, cada registro CDC se escribe sin un primer campo para indicar la operación INSERT en el origen. Para obtener más información acerca de cómo estos parámetros funcionan juntos, consulte <a href="#">Indicar operaciones de base de datos de origen en datos de S3 migrados (p. 209)</a>.</p> <p><b>Note</b></p> <p>AWS DMS admite esta interacción entre <code>cdcInsertsOnly</code> e <code>includeOpForFullLoad</code> en las versiones 3.1.4 y posteriores.</p> <p>Valor predeterminado: <code>false</code></p> <p>Valores válidos: <code>true</code>, <code>false</code>, <code>y</code>, <code>n</code></p> <p>Ejemplo:</p> <pre>cdcInsertsOnly=true;</pre>

Opción	Descripción
<code>includeOpForFullLoad</code>	<p>Un parámetro opcional durante una carga completa para escribir las operaciones INSERT solo en archivos de salida de valores separados por comas (.csv).</p> <p><b>Note</b></p> <p>AWS DMS admite <code>includeOpForFullLoad</code> en las versiones 3.1.4 y posteriores.</p> <p>Para la carga completa, los registros solo se pueden insertar. De forma predeterminada (el valor <code>false</code>), no se registra información en estos archivos de salida para una carga completa para indicar que las filas se insertaron en la base de datos de origen. Si <code>includeOpForFullLoad</code> se establece en <code>true</code> o <code>y</code>, la operación INSERT se registra como una anotación I en el primer campo del archivo.csv.</p> <p><b>Note</b></p> <p>Este parámetro funciona junto con <code>cdcInsertsOnly</code> para la salida solo en archivos .csv. Para obtener más información acerca de cómo estos parámetros funcionan juntos, consulte <a href="#">Indicar operaciones de base de datos de origen en datos de S3 migrados (p. 209)</a>.</p> <p>Valor predeterminado: <code>false</code></p> <p>Valores válidos: <code>true</code>, <code>false</code>, <code>y</code>, <code>n</code></p> <p>Ejemplo:</p> <pre>includeOpForFullLoad=true;</pre>
<code>compressionType</code>	<p>Un parámetro opcional al establecer en <code>GZIP</code> utiliza GZIP para comprimir los archivos .csv o .parquet de destino. Cuando este parámetro se establece en el valor predeterminado, deja los archivos sin comprimir.</p> <p>Valor predeterminado: <code>NONE</code></p> <p>Valores válidos: <code>GZIP</code> o <code>NONE</code></p> <p>Ejemplo:</p> <pre>compressionType=GZIP;</pre>
<code>csvDelimiter</code>	<p>Delimitador utilizado para separar columnas en los archivos .csv de origen. El valor predeterminado es una coma (,).</p> <p>Ejemplo:</p> <pre>csvDelimiter=,;</pre>
<code>csvRowDelimiter</code>	<p>Delimitador utilizado para separar filas en los archivos de origen .csv. El valor predeterminado es una nueva línea (\n).</p> <p>Ejemplo:</p> <pre>csvRowDelimiter=\n;</pre>

Opción	Descripción
<code>maxFileSize</code>	<p>Un valor que especifica el tamaño máximo (en KB) de los archivos .csv que se crean al migrar a un destino de S3 durante la carga completa.</p> <p>Valor predeterminado: 1048576 KB (1 GB)</p> <p>Valores válidos: 1-1048576</p> <p>Ejemplo:</p> <pre>maxFileSize=512</pre>
<code>rfc4180</code>	<p>Un parámetro opcional que se utiliza para establecer un comportamiento de conformidad con RFC de los datos migrados a Amazon S3 utilizando solo el formato de archivo .csv. Cuando este valor se establece en <code>y</code> o <code>true</code> utilizando Amazon S3 como destino, si los datos tienen comillas o caracteres de nueva línea, AWS DMS incluye toda la columna con un par de comillas dobles ("") adicional. Cada comilla dentro de los datos se repite dos veces. Este formato cumple con RFC 4180.</p> <p>Valor predeterminado: <code>true</code></p> <p>Valores válidos: <code>true</code>, <code>false</code>, <code>y</code>, <code>n</code></p> <p>Ejemplo:</p> <pre>rfc4180=false;</pre>
<code>encryptionMode</code>	<p>El modo de cifrado del lado del servidor que desea que cifre sus archivos de objeto .csv o .parquet copiados en S3. Los valores válidos son <code>SSE_S3</code> (cifrado del lado del servidor de S3) o <code>SSE_KMS</code> (cifrado de clave de KMS). Si elige <code>SSE_KMS</code>, establezca el parámetro <code>serverSideEncryptionKeyId</code> en el nombre de recurso de Amazon (ARN) para la clave de KMS que se va a utilizar para cifrado.</p> <p>Valor predeterminado: <code>SSE_S3</code></p> <p>Valores válidos: <code>SSE_S3</code> o <code>SSE_KMS</code></p> <p>Ejemplo:</p> <pre>encryptionMode=SSE_S3;</pre>
<code>serverSideEncryptionKeyId</code>	<p>Si establece <code>encryptionMode</code> en <code>SSE_KMS</code>, establezca este parámetro en el nombre de recurso de Amazon (ARN) para la clave de KMS. Puede encontrar este ARN seleccionando el alias de clave en la lista de claves de KMS creadas para su cuenta. Al crear la clave, debe asociar políticas y roles específicos asociados a esta clave de KMS. Para obtener más información, consulte <a href="#">Creación de claves de AWS KMS para cifrar objetos de destino de Amazon S3 (p. 197)</a>.</p> <p>Ejemplo:</p> <pre>serverSideEncryptionKeyId=arn:aws:kms:us-east-1:944454115380:key/72abb6fb-1e49-4ac1-9aed-c803dfcc0480;</pre>

Opción	Descripción
<code>dataFormat</code>	<p>El formato de salida para los archivos que AWS DMS utiliza para crear objetos de S3. Para destinos de Amazon S3, AWS DMS admite archivos .csv o .parquet. Los archivos .parquet tienen un formato de almacenamiento binario en columnas con opciones de compresión eficientes y un rendimiento de consultas más rápido. Para obtener más información sobre los archivos .parquet, consulte <a href="https://parquet.apache.org/">https://parquet.apache.org/</a>.</p> <p>Valor predeterminado: <code>csv</code></p> <p>Valores válidos: <code>csv</code> o <code>parquet</code></p> <p>Ejemplo:</p> <pre><code>dataFormat=parquet;</code></pre>
<code>encodingType</code>	<p>El tipo de codificación Parquet. Las opciones del tipo de codificación incluyen lo siguiente:</p> <ul style="list-style-type: none"> <li>• <code>rle-dictionary</code>: esta codificación de diccionario que utiliza una combinación de RLE y paquetes de bits para almacenar los valores repetidos con mayor eficiencia.</li> <li>• <code>plain</code>: sin codificación.</li> <li>• <code>plain-dictionary</code>: esta codificación de diccionario crea un diccionario de valores que se encuentran en una columna dada. El diccionario se almacena en una página de diccionario para cada fragmento de columna.</li> </ul> <p>Valor predeterminado: <code>rle-dictionary</code></p> <p>Valores válidos: <code>rle-dictionary</code>, <code>plain</code> o <code>plain-dictionary</code></p> <p>Ejemplo:</p> <pre><code>encodingType=plain-dictionary;</code></pre>
<code>dictPageSizeLimit</code>	<p>El tamaño máximo permitido, en bytes, para una página de diccionario en un archivo .parquet. Si una página de diccionario supera este valor, la página utiliza la codificación sin formato.</p> <p>Valor predeterminado: 1 024 000 (1 MB)</p> <p>Valores válidos: cualquier valor entero válido</p> <p>Ejemplo:</p> <pre><code>dictPageSizeLimit=2,048,000;</code></pre>
<code>rowGroupLength</code>	<p>La cantidad de filas en un grupo de filas de un archivo .parquet.</p> <p>Valor predeterminado: 10 024 (10 KB)</p> <p>Valores válidos: cualquier valor entero válido</p> <p>Ejemplo:</p> <pre><code>rowGroupLength=20,048;</code></pre>

Opción	Descripción
<code>dataPageSize</code>	<p>El tamaño máximo permitido, en bytes, para una página de datos en un archivo .parquet.</p> <p>Valor predeterminado: 1 024 000 (1 MB)</p> <p>Valores válidos: cualquier valor entero válido</p> <p>Ejemplo:</p> <pre>dataPageSize=2,048,000;</pre>
<code>parquetVersion</code>	<p>La versión del formato de archivo .parquet.</p> <p>Valor predeterminado: PARQUET_1_0</p> <p>Valores válidos: PARQUET_1_0 o PARQUET_2_0</p> <p>Ejemplo:</p> <pre>parquetVersion=PARQUET_2_0;</pre>
<code>enableStatistics</code>	<p>Establezca en <code>true</code> o <code>y</code> para habilitar las estadísticas acerca de las páginas de archivo .parquet y grupos de filas.</p> <p>Valor predeterminado: <code>true</code></p> <p>Valores válidos: <code>true</code>, <code>false</code>, <code>y</code>, <code>n</code></p> <p>Ejemplo:</p> <pre>enableStatistics=false;</pre>
<code>timestampColumnName</code>	<p>Un parámetro opcional para incluir una columna de marca temporal en los datos de punto de enlace de destino de S3.</p> <p><b>Note</b></p> <p>AWS DMS admite <code>timestampColumnName</code> en las versiones 3.1.4 y posteriores.</p> <p>AWS DMS incluye una columna adicional en los datos migrados cuando se establece <code>timestampColumnName</code> en un valor no vacío. Para una carga completa, cada fila de la columna de marca temporal contiene una marca temporal que indica cuándo DMS transfirió los datos del origen al destino. Para una carga CDC, cada fila de la columna de marca temporal contiene la marca temporal de confirmación de esa fila en la base de datos de origen. El formato del valor de la columna de marca temporal es <code>yyyy-MM-dd HH:mm:ss . SSSSSS</code>. Para CDC, la precisión de microsegundos depende de la marca temporal de confirmación admitida por DMS para la base de datos de origen. Cuando el parámetro <code>addColumnName</code> está establecido en <code>true</code>, DMS incluye también el nombre de la columna de marca temporal definido como el valor no vacío de <code>timestampColumnName</code>.</p> <p>Ejemplo:</p> <pre>timestampColumnName=TIMESTAMP;</pre>

## Indicar operaciones de base de datos de origen en datos de S3 migrados

Cuando AWS DMS migra registros a un destino de S3, puede crear un campo adicional en cada registro migrado. Este campo adicional indica la operación que se aplica al registro en la base de datos de origen.

Para una carga completa, cuando `includeOpForFullLoad` es `true` y el formato de salida es `.csv`, DMS siempre crea un primer campo adicional en cada registro `.csv`. Este campo contiene la letra `I` (`INSERT`) para indicar que la fila se insertó en la base de datos de origen. Para una carga CDC, cuando `cdcInsertsOnly` es `false` (el valor predeterminado), DMS siempre crea un primer campo adicional en cada registro `.csv` o `.parquet`. Este campo contiene la letra `I` (`INSERT`), `U` (`UPDATE`) o `D` (`DELETE`) para indicar si la fila se insertó, actualizó o eliminó en la base de datos de origen.

Cuando el formato de salida es solo `.csv`, si DMS crea y configura este primer campo y cómo lo hace depende también del valor de `includeOpForFullLoad` y del valor de `cdcInsertsOnly`. En la siguiente tabla, puede ver cómo el valor de estos atributos funcionan juntos y afectan al valor de los registros migrados en este formato.

### Note

AWS DMS admite `includeOpForFullLoad` y la interacción se describe aquí entre `includeOpForFullLoad` y `cdcInsertsOnly` en las versiones 3.1.4 y posteriores.

Con estos valores de los parámetros		DMS establece los registros de destino de la forma siguiente para la salida <code>.csv</code>	
<code>true</code>	<code>true</code>	Valor del primer campo añadido establecido en <code>I</code>	Valor del primer campo añadido establecido en <code>I</code>
<code>false</code>	<code>false</code>	No se ha añadido el campo	Valor del primer campo añadido establecido en <code>I, U o D</code>
<code>false</code>	<code>true</code>	No se ha añadido el campo	No se ha añadido el campo
<code>true</code>	<code>false</code>	Valor del primer campo añadido establecido en <code>I</code>	Valor del primer campo añadido establecido en <code>I, U o D</code>

Cuando `includeOpForFullLoad` y `cdcInsertsOnly` se establecen en el mismo valor, los registros de destino se establecen de acuerdo con el atributo que controla el valor del registro para el tipo de migración actual. Este atributo es `includeOpForFullLoad` para la carga completa y `cdcInsertsOnly` para la carga CDC.

Cuando `includeOpForFullLoad` y `cdcInsertsOnly` se establecen en valores diferentes, AWS DMS hace que el valor del registro de destino sea coherente para la carga CDC y la carga completa. Para ello, hace que el valor del registro para una carga CDC se ajuste al valor del registro de una carga completa anterior especificada por `includeOpForFullLoad`.

Supongamos, por ejemplo, que una carga completa se establece para añadir un primer campo que indique un registro insertado. En este caso, una carga CDC posterior se establece para añadir un primer campo que indique un registro insertado, actualizado o eliminado, según corresponda en el origen. Supongamos ahora que una carga completa se establece en no añadir un primer campo para indicar un registro insertado. En este caso, una carga CDC se establece también para no añadir un primer campo a cada registro independientemente de su operación de registro correspondiente en el origen.

## Uso de una base de datos de Amazon DynamoDB como destino para AWS Database Migration Service

Puede utilizar AWS DMS para migrar datos a una tabla de Amazon DynamoDB. Amazon DynamoDB es un servicio de base de datos NoSQL totalmente administrado que ofrece un desempeño rápido y previsible, así como una escalabilidad óptima. AWS DMS es compatible con la base de datos relacional o MongoDB como origen.

En DynamoDB se trabaja principalmente con tablas, elementos y atributos. Una tabla es una colección de elementos y cada elemento es una colección de atributos. DynamoDB utiliza claves principales, denominadas claves de partición, para identificar de forma exclusiva cada uno de los elementos de la tabla. También puede utilizar claves e índices secundarios para proporcionar más flexibilidad a la hora de realizar consultas.

Puede utilizar el mapeo de objetos para migrar sus datos desde una base de datos de origen a una tabla de DynamoDB de destino. El mapeo de objetos le permite determinar dónde se encuentran los datos de origen en el destino.

Cuando AWS DMS crea tablas en un punto de enlace de destino de Amazon DynamoDB, crea tantas tablas como haya en el punto de enlace de la base de datos de origen. AWS DMS también establece varios valores de parámetros de Amazon DynamoDB. El costo de la creación de la tabla depende de la cantidad de datos y del número de tablas que hay que migrar.

Cuando AWS DMS establece valores de parámetros de Amazon DynamoDB para una tarea de migración, el valor del parámetro para las unidades de capacidad de lectura (RCU) predeterminadas se establece en 200.

También se establece el valor del parámetro de las unidades de capacidad de escritura (WCU), pero su valor depende de otras configuraciones diferentes:

- El valor predeterminado para el parámetro WCU es 200.
- Si el parámetro parallelLoadThreads se establece en un valor superior a 1 (el valor predeterminado es 0), entonces el parámetro WCU se establece en un valor 200 veces el valor de parallelLoadThreads.
- En la región EE.UU. Este (Norte de Virginia) (us-east-1), el valor del parámetro WCU más alto posible es 40000. Si la región de AWS es us-east-1 y el valor del parámetro WCU es superior a 40000, el valor del parámetro WCU se establece en 40000.
- En regiones de AWS distintas a us-east-1, el valor del parámetro WCU más alto posible es 10000. Para cualquier región de AWS distinta a us-east-1, si el valor del parámetro WCU se establece en un valor superior a 10000, el valor del parámetro WCU se establece en 10000.

## Migración desde una base de datos relacional a una tabla de DynamoDB

AWS DMS permite la migración de datos a tipos de datos escalares de DynamoDB. Al migrar desde una base de datos relacional como Oracle o MySQL a DynamoDB, puede reestructurar la manera de almacenar dichos datos.

En la actualidad, AWS DMS admite la reestructuración de una tabla en otra tabla con los atributos del tipo escalar de DynamoDB. Si migra datos a DynamoDB desde una tabla de base de datos relacional, toma los datos de una tabla y cambia su formato por atributos de tipo de datos escalares de DynamoDB. Estos atributos pueden aceptar datos de varias columnas y puede mapear una columna en un atributo directamente.

AWS DMS admite los siguientes tipos de datos escalares de DynamoDB:

- Cadena
- Número
- Booleano

Note

Los datos NULL del origen se ignoran en el destino.

## Requisitos previos para utilizar DynamoDB como destino de AWS Database Migration Service

Antes de empezar a trabajar con una base de datos de DynamoDB como destino de AWS DMS, asegúrese de crear un rol de IAM. Este rol de IAM debería permitir a AWS DMS asumir y conceder acceso a las tablas de DynamoDB a las que se va a migrar. El conjunto mínimo de permisos de acceso se muestra en la siguiente política de roles de ejemplo.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": {  
                "Service": "dms.amazonaws.com"  
            },  
            "Action": "sts:AssumeRole"  
        }  
    ]  
}
```

La función que utilice para la migración a DynamoDB debe tener los siguientes permisos.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "dynamodb:PutItem",  
                "dynamodb:CreateTable",  
                "dynamodb:DescribeTable",  
                "dynamodb>DeleteTable",  
                "dynamodb:DeleteItem"  
            ],  
            "Resource": [  
                "arn:aws:dynamodb:us-west-2:account-id:table/Name1",  
                "arn:aws:dynamodb:us-west-2:account-id:table/OtherName*",  
            ]  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "dynamodb>ListTables"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

## Restricciones cuando se usa DynamoDB como destino para AWS Database Migration Service

Al utilizar Amazon DynamoDB como destino se aplican las siguientes restricciones:

- DynamoDB limita la precisión del tipo de datos Number a 38 espacios. Almacene todos los tipos de datos con una mayor precisión como cadena. Deberá indicarlo explícitamente empleando la característica de mapeo de objetos.
- Debido a que Amazon DynamoDB no tiene un tipo de datos Date, los datos que utilizan el tipo de datos Date se convierten en cadenas.
- Amazon DynamoDB no permite actualizaciones de los atributos de clave principal. Esta restricción es importante cuando se utiliza la replicación continua con captura de datos de cambio (CDC), ya que puede resultar en la presencia de datos no deseados en el destino. En función del mapeo de objetos, una operación de CDC que actualiza la clave principal puede hacer una de estas dos opciones. Puede producir un error o insertar un nuevo elemento con la clave principal actualizada y datos incompletos.
- AWS DMS solo admite la replicación de tablas con claves principales no compuestas. La excepción es si especifica un mapeo de objetos para la tabla de destino con una clave de partición personalizada, una clave de ordenación o ambas.
- AWS DMS no admite datos LOB salvo si se trata de un CLOB. AWS DMS convierte los datos CLOB en una cadena de DynamoDB al migrar los datos.
- Cuando se utiliza DynamoDB como destino, solo se admite la tabla de control Apply Exceptions (Aplicar excepciones) (`dmslogs.awsdms_apply_exceptions`). Para obtener más información sobre las tablas de control, consulte [Configuración de las tareas de la tabla de control \(p. 266\)](#).

## Uso del mapeo de objetos para migrar datos a DynamoDB

AWS DMS usa las reglas de mapeo de tablas para asignar datos del origen a la tabla de DynamoDB de destino. Para asignar datos a un destino de DynamoDB, se utiliza un tipo de regla de mapeo de tablas denominada object-mapping. El mapeo de objetos le permite definir los nombres de atributo y los datos que se les puede migrar. Debe tener reglas de selección cuando utilice el mapeo de objetos.

Amazon DynamoDB no tiene una estructura predeterminada, simplemente dispone de una clave de partición y una clave de ordenación opcional. Si dispone de una clave principal no compuesta, AWS DMS la utiliza. Si tiene una clave principal compuesta o desea utilizar una clave de ordenación, defina estas claves y el resto de los atributos de su tabla de DynamoDB de destino.

Para crear una regla de mapeo de objetos, debe especificar `rule-type` como `object-mapping`. Esta regla indica el tipo de mapeo de objetos que desea utilizar.

La estructura de la regla es la siguiente:

```
{ "rules": [
    {
        "rule-type": "object-mapping",
        "rule-id": "<id>",
        "rule-name": "<name>",
        "rule-action": "<valid object-mapping rule action>",
        "object-locator": {
            "schema-name": "<case-sensitive schema name>",
            "table-name": ""
        },
        "target-table-name": "<table_name>",
        ...
    }
]
```

AWS DMS actualmente admite `map-record-to-record` y `map-record-to-document` como únicos valores válidos para el parámetro `rule-action`. Estos valores especifican qué hace AWS DMS de forma predeterminada con los registros que no se excluyen como parte de la lista de atributos `exclude-columns`. Estos valores no afectan a los mapeos de atributos en modo alguno.

- Puede utilizar `map-record-to-record` al migrar desde una base de datos relacional a DynamoDB. Utiliza la clave principal de la base de datos relacional como la clave de partición en Amazon DynamoDB y crea un atributo para cada columna de la base de datos de origen. Cuando se utiliza `map-record-to-record`, para cada columna de la tabla de origen que no figura en la lista de atributos de `exclude-columns`, AWS DMS crea un atributo correspondiente en la instancia de DynamoDB de destino. Lo hace independientemente de si dicha columna de origen se utiliza en un mapeo de atributos.
- Utilice `map-record-to-document` para colocar columnas de origen en un mapeo de DynamoDB único y plano en el destino utilizando el nombre de atributo "`_doc`". Cuando utilice `map-record-to-document`, AWS DMS coloca los datos en un atributo de mapeo de DynamoDB único y plano en el origen. Este atributo se denomina "`_doc`". Esta colocación se aplica a cada columna de la tabla de origen que no se enumera en la lista de atributos `exclude-columns`.

Una forma de entender la diferencia entre los parámetros de `rule-action map-record-to-record` y `map-record-to-document` consiste en ver los dos parámetros en acción. En este ejemplo, imagine que empieza con una fila de una tabla de base de datos relacional con la estructura y los datos siguientes:

FirstName	LastName	NickName	WorkAddress	WorkPhone	HomeAddress	HomePhone
Daniel	Sheridan	Dan	101 Main St Cambridge, MA	800-867-5309	100 Secret St, Unknownville, MA	123-456-7890

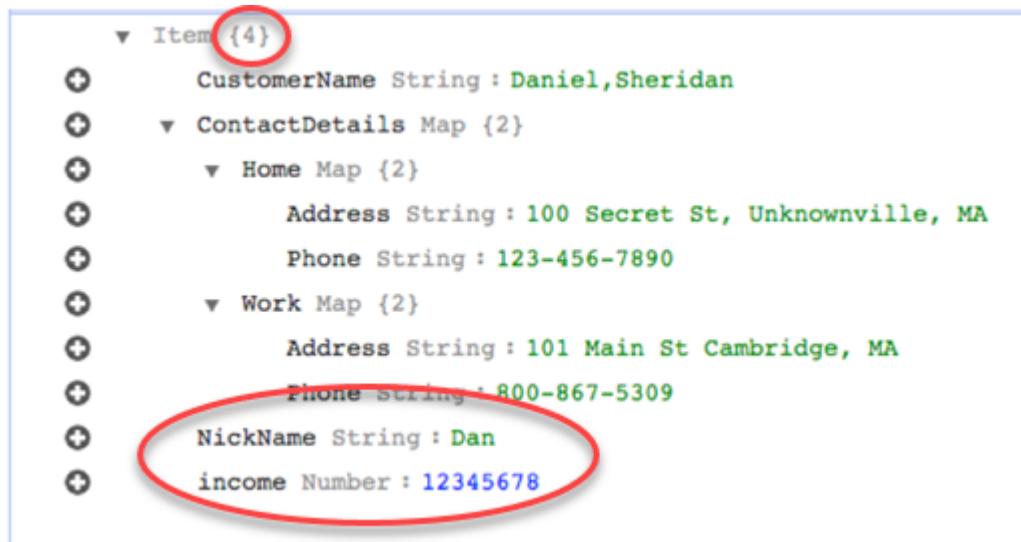
Para migrar esta información a DynamoDB, crea reglas para mapear los datos en un elemento de la tabla de DynamoDB. Tenga en cuenta las columnas listadas para el parámetro `exclude-columns`. Estas columnas no se mapean directamente en el destino. En su lugar, el mapeo de atributos se utiliza para combinar los datos en elementos nuevos como, por ejemplo, las columnas `FirstName` (Nombre) y `LastName` (Apellidos) que se agrupan en `CustomerName` (NombreCliente) en el destino de DynamoDB. `NickName` (Alias) e `income` (ingresos) no se excluyen.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "test",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "object-mapping",
            "rule-id": "1",
            "rule-name": "TransformToDDB",
            "rule-action": "map-record-to-record",
            "object-locator": {
                "schema-name": "test",
                "table-name": "customer"
            },
            "target-table-name": "customer_t",
            "mapping-parameters": {
                "partition-key-name": "CustomerName",
                "exclude-columns": [
                    "FirstName",
                    "LastName",
                    "HomeAddress",

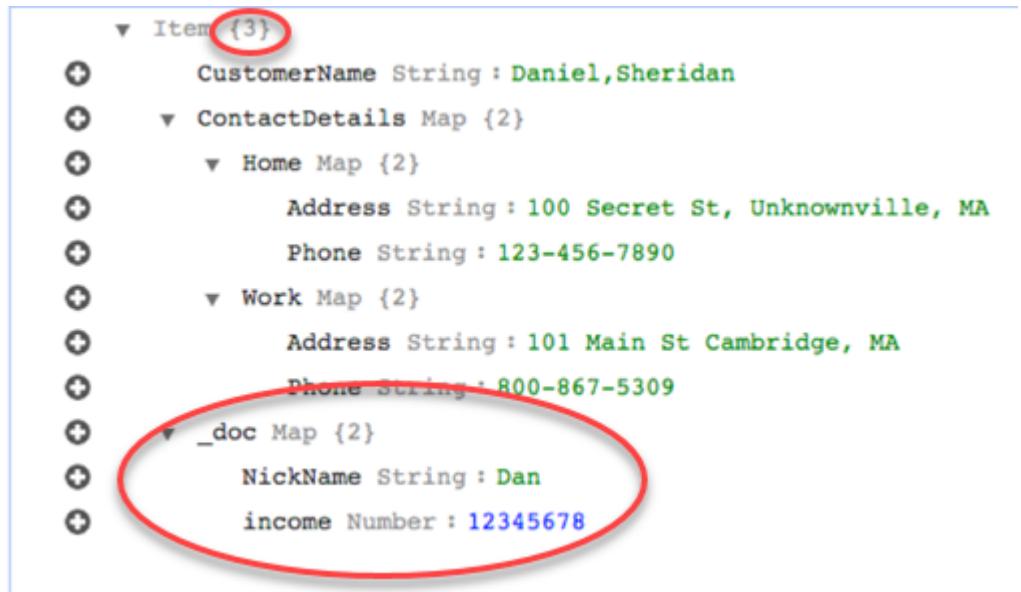
```

```
        "HomePhone",
        "WorkAddress",
        "WorkPhone"
    ],
    "attribute-mappings": [
        {
            "target-attribute-name": "CustomerName",
            "attribute-type": "scalar",
            "attribute-sub-type": "string",
            "value": "${FirstName},${LastName}"
        },
        {
            "target-attribute-name": "ContactDetails",
            "attribute-type": "document",
            "attribute-sub-type": "dynamodb-map",
            "value": {
                "M": {
                    "Home": {
                        "M": {
                            "Address": {
                                "S": "${HomeAddress}"
                            },
                            "Phone": {
                                "S": "${HomePhone}"
                            }
                        }
                    },
                    "Work": {
                        "M": {
                            "Address": {
                                "S": "${WorkAddress}"
                            },
                            "Phone": {
                                "S": "${WorkPhone}"
                            }
                        }
                    }
                }
            }
        }
    ]
}
```

Al utilizar el parámetro `rule-action` de la map-record-to-record, los datos de NickName e income se mapean en los elementos del mismo nombre en el destino de DynamoDB.



Sin embargo, supongamos que utiliza las mismas reglas pero cambia el parámetro `rule-action` a map-record-to-document. En este caso, las columnas que no aparecen en el parámetro `exclude-columns`, NickName (Alias) e income (ingresos), se asignan a un elemento `_doc`.



## Uso de expresiones de condición personalizadas con el mapeo de objetos

Puede utilizar una característica de Amazon DynamoDB denominada "expresiones de condición" para manipular los datos que se escriben en una tabla de DynamoDB. Para obtener más información sobre las expresiones de condición en DynamoDB, consulte [Expresiones de condición](#).

Un miembro de una expresión de condición consta de:

- una expresión (obligatorio)
- los valores de los atributos de expresión (opcional). Especifica una estructura json de DynamoDB del valor del atributo
- los nombres de los atributos de expresión (opcional)

- las opciones sobre cuándo utilizar la expresión de condición (opcional). El valor predeterminado es apply-during-cdc = false y apply-during-full-load = true

La estructura de la regla es la siguiente:

```
"target-table-name": "customer_t",
  "mapping-parameters": {
    "partition-key-name": "CustomerName",
    "condition-expression": {
      "expression": "<conditional expression>",
      "expression-attribute-values": [
        {
          "name": "<attribute name>",
          "value": "<attribute value>"
        }
      ],
      "apply-during-cdc": <optional Boolean value>,
      "apply-during-full-load": <optional Boolean value>
    }
}
```

En el siguiente ejemplo se destacan las secciones que se utilizan para la expresión de condición.

```
{
  "rules": [
    {
      "rule-type": "object-mapping",
      "rule-id": "1",
      "rule-name": "TransformToDDB",
      "rule-action": "map-record-to-record",
      "object-locator": {
        "schema-name": "test",
        "table-name": "customer",
      },
      "target-table-name": "customer_t",
      "mapping-parameters": {
        "partition-key-name": "CustomerName",
        "condition-expression": {
          "expression": "attribute_not_exists(version) or version <= :record_version",
          "expression-attribute-values": [
            {
              "name": ":record_version",
              "value": {"N": "${version}"}
            }
          ],
          "apply-during-cdc": true,
          "apply-during-full-load": true
        }
      },
      "attribute-mappings": [
        {
          "target-attribute-name": "CustomerName",
          "attribute-type": "scalar",
          "attribute-sub-type": "string",
          "value": "${FirstName}, ${LastName}"
        }
      ]
    }
  ]
}
```

Object mapping section defines name, rule-action, and object locator information

Condition expression

Options

## Uso del mapeo de atributos con el mapeo de objetos

El mapeo de atributos le permite especificar una cadena de ejemplo utilizando nombres de columna del origen para reestructurar los datos en el destino. El formato se modifica en función de lo que especifique el usuario en la plantilla.

El siguiente ejemplo muestra la estructura de la base de datos de origen y la estructura deseada de destino en DynamoDB. En primer lugar se muestra la estructura de origen, en este caso, una base de datos de Oracle y, a continuación, la estructura deseada de los datos en DynamoDB. El ejemplo concluye con la estructura JSON utilizada para crear la estructura de destino deseada.

La estructura de los datos de Oracle es la siguiente:

First	Last	Stc	HomeA	HomePl	WorkAddres	Work	DateOfBirth
Clave principal	N/D						
Randy	Mars	15	221B Baker Street	1234567890	Spooner Street, Quahog	9876543210	02/29/1988

La estructura de los datos de DynamoDB es la siguiente:

Custom	StoreId	ContactDetails	DateOfBirth	
Clave de partición	Clave de ordenación	N/D		
Randy	Mars	<pre>{     "Name": "Randy",     "Home": {         "Address": "221B Baker Street",         "Phone": 1234567890     },     "Work": {         "Address": "31 Spooner Street,         Quahog",         "Phone": 9876541230     } }</pre>	02/29/1988	

La siguiente estructura JSON muestra el mapeo de objetos y de columnas que se utiliza para conseguir la estructura de DynamoDB:

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "object-locator": {
        "schema-name": "test",
        "table-name": "Contact"
      }
    }
  ]
}
```

```

        "table-name": "%"
    },
    "rule-action": "include"
},
{
    "rule-type": "object-mapping",
    "rule-id": "2",
    "rule-name": "TransformToDDB",
    "rule-action": "map-record-to-record",
    "object-locator": {
        "schema-name": "test",
        "table-name": "customer"
    },
    "target-table-name": "customer_t",
    "mapping-parameters": {
        "partition-key-name": "CustomerName",
        "sort-key-name": "StoreId",
        "exclude-columns": [
            "FirstName",
            "LastName",
            "HomeAddress",
            "HomePhone",
            "WorkAddress",
            "WorkPhone"
        ],
        "attribute-mappings": [
            {
                "target-attribute-name": "CustomerName",
                "attribute-type": "scalar",
                "attribute-sub-type": "string",
                "value": "${FirstName},${LastName}"
            },
            {
                "target-attribute-name": "StoreId",
                "attribute-type": "scalar",
                "attribute-sub-type": "string",
                "value": "${StoreId}"
            },
            {
                "target-attribute-name": "ContactDetails",
                "attribute-type": "scalar",
                "attribute-sub-type": "string",
                "value": "{\"Name\": ${FirstName}, \"Home\": {\"Address\": \"${HomeAddress}\", \"Phone\": \"${HomePhone}\"}, \"Work\": {\"Address\": \"${WorkAddress}\", \"Phone\": \"${WorkPhone}\"}}"
            }
        ]
    }
}

```

Otro modo de utilizar el mapeo de columnas es utilizar el formato DynamoDB como su tipo de documento. El siguiente ejemplo de código utiliza dynamodb-map como el attribute-sub-type para el mapeo de atributos.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "rule-condition": {
                "operator": "EQ",
                "key": "CustomerName",
                "value": "JohnDoe"
            }
        }
    ]
}
```

```
"object-locator": {
    "schema-name": "test",
    "table-name": "%"
},
"rule-action": "include"
},
{
    "rule-type": "object-mapping",
    "rule-id": "2",
    "rule-name": "TransformToDDB",
    "rule-action": "map-record-to-record",
    "object-locator": {
        "schema-name": "test",
        "table-name": "customer"
    },
    "target-table-name": "customer_t",
    "mapping-parameters": {
        "partition-key-name": "CustomerName",
        "sort-key-name": "StoreId",
        "exclude-columns": [
            "FirstName",
            "LastName",
            "HomeAddress",
            "HomePhone",
            "WorkAddress",
            "WorkPhone"
        ],
        "attribute-mappings": [
            {
                "target-attribute-name": "CustomerName",
                "attribute-type": "scalar",
                "attribute-sub-type": "string",
                "value": "${FirstName},${LastName}"
            },
            {
                "target-attribute-name": "StoreId",
                "attribute-type": "scalar",
                "attribute-sub-type": "string",
                "value": "${StoreId}"
            },
            {
                "target-attribute-name": "ContactDetails",
                "attribute-type": "scalar",
                "attribute-sub-type": "string",
                "value": "{\"Name\": ${FirstName}, \"Home\": {\"Address\": ${HomeAddress}}, \"Phone\": ${HomePhone}, \"Work\": {\"Address\": ${WorkAddress}}, \"Phone\": ${WorkPhone}}"
            }
        ]
    }
}
```

### Ejemplo 1: Uso del mapeo de atributos con el mapeo de objetos

El siguiente ejemplo migra datos de dos tablas de la base de datos de MySQL, nfl\_data y sport\_team, a dos tablas de DynamoDB denominadas NFLTeams y SportTeams. A continuación se muestra la estructura de las tablas y la estructura JSON que se utilizan para mapear los datos de las tablas de la base de datos MySQL en las tablas de DynamoDB.

A continuación se muestra la estructura de la tabla de base de datos MySQL nfl\_data:

```
mysql> desc nfl_data;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| Position | varchar(5) | YES | | NULL | |
| player_number | smallint(6) | YES | | NULL | |
| Name | varchar(40) | YES | | NULL | |
| status | varchar(10) | YES | | NULL | |
| stat1 | varchar(10) | YES | | NULL | |
| stat1_val | varchar(10) | YES | | NULL | |
| stat2 | varchar(10) | YES | | NULL | |
| stat2_val | varchar(10) | YES | | NULL | |
| stat3 | varchar(10) | YES | | NULL | |
| stat3_val | varchar(10) | YES | | NULL | |
| stat4 | varchar(10) | YES | | NULL | |
| stat4_val | varchar(10) | YES | | NULL | |
| team | varchar(10) | YES | | NULL | |
+-----+-----+-----+-----+-----+
```

A continuación se muestra la estructura de la tabla de la base de datos MySQL sport\_team:

```
mysql> desc sport_team;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| id | mediumint(9) | NO | PRI | NULL | auto_increment |
| name | varchar(30) | NO | | NULL | |
| abbreviated_name | varchar(10) | YES | | NULL | |
| home_field_id | smallint(6) | YES | MUL | NULL | |
| sport_type_name | varchar(15) | NO | MUL | NULL | |
| sport_league_short_name | varchar(10) | NO | | NULL | |
| sport_division_short_name | varchar(10) | YES | | NULL | |
+-----+-----+-----+-----+-----+
```

A continuación, se muestran las reglas de mapeo de tablas que se utilizan para asignar las dos tablas a las dos tablas de DynamoDB:

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "object-locator": {
        "schema-name": "dms_sample",
        "table-name": "nfl_data"
      },
      "rule-action": "include"
    },
    {
      "rule-type": "selection",
      "rule-id": "2",
      "rule-name": "2",
      "object-locator": {
        "schema-name": "dms_sample",
        "table-name": "sport_team"
      },
      "rule-action": "include"
    }
  ]
}
```

```
{  
    "rule-type": "object-mapping",  
    "rule-id": "3",  
    "rule-name": "MapNFLData",  
    "rule-action": "map-record-to-record",  
    "object-locator": {  
        "schema-name": "dms_sample",  
        "table-name": "nfl_data"  
    },  
    "target-table-name": "NFLTeams",  
    "mapping-parameters": {  
        "partition-key-name": "Team",  
        "sort-key-name": "PlayerName",  
        "exclude-columns": [  
            "player_number", "team", "Name"  
        ],  
        "attribute-mappings": [  
            {  
                "target-attribute-name": "Team",  
                "attribute-type": "scalar",  
                "attribute-sub-type": "string",  
                "value": "${team}"  
            },  
            {  
                "target-attribute-name": "PlayerName",  
                "attribute-type": "scalar",  
                "attribute-sub-type": "string",  
                "value": "${Name}"  
            },  
            {  
                "target-attribute-name": "PlayerInfo",  
                "attribute-type": "scalar",  
                "attribute-sub-type": "string",  
                "value": "{\"Number\": \"${player_number}\", \"Position\": \"${Position}\",  
\"Status\": \"${status}\", \"Stats\": {\"Stat1\": \"${stat1}:${stat1_val}\", \"Stat2\":  
\"${stat2}:${stat2_val}\", \"Stat3\": \"${stat3}:${stat3_val}\", \"Stat4\": \"${stat4}:${stat4_val}\")"  
            }  
        ]  
    }  
},  
{  
    "rule-type": "object-mapping",  
    "rule-id": "4",  
    "rule-name": "MapSportTeam",  
    "rule-action": "map-record-to-record",  
    "object-locator": {  
        "schema-name": "dms_sample",  
        "table-name": "sport_team"  
    },  
    "target-table-name": "SportTeams",  
    "mapping-parameters": {  
        "partition-key-name": "TeamName",  
        "exclude-columns": [  
            "name", "id"  
        ],  
        "attribute-mappings": [  
            {  
                "target-attribute-name": "TeamName",  
                "attribute-type": "scalar",  
                "attribute-sub-type": "string",  
                "value": "${name}"  
            },  
            {  
                "target-attribute-name": "TeamInfo",  
                "attribute-type": "scalar",  
                "attribute-sub-type": "string",  
                "value": "${info}"  
            }  
        ]  
    }  
}
```

```
        "attribute-sub-type": "string",
        "value": "{\"League\": \"${sport_league_short_name}\", \"Division\":
\"${sport_division_short_name}\"}"
    }
}
]
}
}
```

A continuación se muestra el resultado de ejemplo de la tabla NFLTeams de DynamoDB:

```
"PlayerInfo": "{\"Number\": \"6\", \"Position\": \"P\", \"Status\": \"ACT\", \"Stats\":
{\"Stat1\": \"PUNTS:73\", \"Stat2\": \"AVG:46\", \"Stat3\": \"LNG:67\", \"Stat4\": \"IN
20:31\"}",
"PlayerName": "Allen, Ryan",
"Position": "P",
"stat1": "PUNTS",
"stat1_val": "73",
"stat2": "AVG",
"stat2_val": "46",
"stat3": "LNG",
"stat3_val": "67",
"stat4": "IN 20",
"stat4_val": "31",
"status": "ACT",
"Team": "NE"
}
```

A continuación se muestra el resultado de ejemplo de la tabla SportsTeams de DynamoDB:

```
{
    "abbreviated_name": "IND",
    "home_field_id": 53,
    "sport_division_short_name": "AFC South",
    "sport_league_short_name": "NFL",
    "sport_type_name": "football",
    "TeamInfo": "{\"League\": \"NFL\", \"Division\": \"AFC South\"}",
    "TeamName": "Indianapolis Colts"
}
```

## Tipos de datos de destino para DynamoDB

El punto de enlace de Amazon DynamoDB para AWS DMS es compatible con la mayoría de los tipos de datos de DynamoDB. La siguiente tabla muestra los tipos de datos de destino de Amazon AWS DMS que se admiten cuando se utiliza AWS DMS y la asignación predeterminada desde los tipos de datos de AWS DMS.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Cuando AWS DMS migra datos desde bases de datos heterogéneas, se mapean tipos de datos de la base de datos de origen en los tipos de datos intermedios llamados tipos de datos de AWS DMS. A continuación, se mapean los tipos de datos intermedios en los tipos de datos de destino. La tabla siguiente muestra cada tipo de datos de AWS DMS y el tipo de datos con el que se mapea en DynamoDB:

Tipo de datos de AWS DMS	Tipo de datos de DynamoDB
Cadena	Cadena
WString	Cadena
Booleano	Booleano
Fecha	Cadena
DateTime	Cadena
INT1	Número
INT2	Número
INT4	Número
INT8	Número
Numérico	Número
Real4	Número
Real8	Número
UINT1	Número
UINT2	Número
UINT4	Número
UINT8	Número
CLOB	Cadena

## Uso de Amazon Kinesis Data Streams como destino para AWS Database Migration Service

Puede utilizar AWS DMS para migrar datos a una secuencia de datos de Amazon Kinesis. Las secuencias de datos de Amazon Kinesis forman parte del servicio Amazon Kinesis Data Streams. Puede utilizar secuencias de datos de Kinesis para recopilar y procesar grandes secuencias de registros de datos en tiempo real.

Una secuencia de datos de Kinesis se compone de particiones. Las particiones son secuencias de registros de datos identificados inequívocamente en una secuencia. Para obtener más información acerca de las particiones Amazon Kinesis Data Streams, consulte el tema sobre [particiones](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

AWS Database Migration Service publica registros en una secuencia de datos de Kinesis utilizando JSON. Durante la conversión, AWS DMS serializa cada registro de la base de datos de origen en un par de atributo-valor en formato JSON.

Para migrar datos a Amazon Kinesis Data Streams, utilice la versión del motor de AWS Database Migration Service 3.1.2 o superior.

Puede utilizar el mapeo de objetos para migrar sus datos desde cualquier origen de datos admitido a una secuencia de destino. Con el mapeo de datos, se determina cómo se estructuran los registros de datos de

la secuencia. También debe definir una clave de partición para cada tabla, que Kinesis Data Streams utiliza para agrupar los datos en particiones.

## Requisitos previos para el uso de una secuencia de datos de Kinesis como destino para AWS Database Migration Service

Antes de configurar una secuencia de datos de Kinesis como destino para AWS DMS, asegúrese de crear un rol de IAM. Este rol debe permitir a AWS DMS asumir y conceder acceso a las secuencias de datos de Kinesis que se van a migrar. El conjunto mínimo de permisos de acceso se muestra en la siguiente política de roles de ejemplo.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "1",  
            "Effect": "Allow",  
            "Principal": {  
                "Service": "dms.amazonaws.com"  
            },  
            "Action": "sts:AssumeRole"  
        }  
    ]  
}
```

El rol que utilice para la migración a una secuencia de datos de Kinesis debe tener los siguientes permisos.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "kinesis:DescribeStream",  
                "kinesis:PutRecord",  
                "kinesis:PutRecords"  
            ],  
            "Resource": "arn:aws:kinesis:region:accountID:stream/streamName"  
        }  
    ]  
}
```

## Restricciones cuando se usa Kinesis Data Streams como destino para AWS Database Migration Service

Al utilizar Kinesis Data Streams como destino se aplican las siguientes restricciones:

- AWS DMS publica cada actualización en un solo registro de la base de datos de origen como un único registro de datos en una determinada secuencia de datos de Kinesis. Como resultado, las aplicaciones que consumen los datos de la secuencia pierden el rastreo de los límites de transacción.
- Las secuencias de datos de Kinesis no admiten la desduplicación. Las aplicaciones que consumen datos de una secuencia necesitan ocuparse de los registros duplicados. Para obtener más información, consulte [Administración de registros duplicados](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- AWS DMS admite los dos formatos siguientes para las claves de partición:
  - SchemaName . TableName: una combinación del nombre de esquema y de tabla.
  - \${AttributeName}: el valor de uno de los campos del archivo JSON o la clave principal de la tabla de la base de datos de origen.

## Uso del mapeo de objetos para migrar datos a una secuencia de datos de Kinesis

AWS DMS usa reglas de mapeo de tablas para asignar datos del origen a la secuencia de datos de Kinesis de destino. Para asignar datos a una secuencia de destino, se utiliza un tipo de regla de mapeo de tablas denominada "object mapping". Puede utilizar el mapeo de objetos para definir cómo los registros de datos del origen se asignan a los registros de datos publicados en la secuencia de datos de Kinesis.

Las secuencias de datos de Kinesis no tienen una estructura predeterminada distinta de una clave de partición.

Para crear una regla de mapeo de objetos, especifique `rule-type` como `object-mapping`. Esta regla indica el tipo de mapeo de objetos que desea utilizar.

La estructura de la regla es la siguiente.

```
{
  "rules": [
    {
      "rule-type": "object-mapping",
      "rule-id": "id",
      "rule-name": "name",
      "rule-action": "valid object-mapping rule action",
      "object-locator": {
        "schema-name": "case-sensitive schema name",
        "table-name": ""
      }
    }
  ]
}
```

AWS DMS admite actualmente `map-record-to-record` y `map-record-to-document` como los únicos valores válidos para el parámetro `rule-action`. `Map-record-to-record` y `map-record-to-document` especifican lo que AWS DMS hace de forma predeterminada con los registros que no están excluidos como parte de la lista de atributos de `exclude-columns`. Estos valores no afectan a los mapeos de atributos en modo alguno.

Utilice `map-record-to-record` al migrar desde una base de datos relacional a una secuencia de datos de Kinesis. Este tipo de regla utiliza el valor `taskResourceId.schemaName.tableName` de la base de datos relacional como la clave de partición en la secuencia de datos de Kinesis y crea un atributo para cada columna de la base de datos de origen. Cuando se utiliza `map-record-to-record`, para cada columna de la tabla de origen que no figura en la lista de atributos de `exclude-columns`, AWS DMS crea un atributo correspondiente en la secuencia de destino. Este atributo se crea independientemente de si dicha columna de origen se utiliza en un mapeo de atributos.

Una forma de entender `map-record-to-record` es verlo en acción. En este ejemplo, imagine que empieza con una fila de una tabla de base de datos relacional con la estructura y los datos siguientes.

FirstName	LastName	StoreId	HomeAddress	HomePhone	WorkAddress	WorkPhone	DateOfBirth
Randy	Marsh	5	221B Baker Street	1234567890	31 Spooner	9876543210	02/29/1988

FirstName	LastName	StoreId	HomeAddress	HomePhone	WorkAddress	WorkPhone	DateofBirth
					Street, Quahog		

Para migrar esta información a una secuencia de datos de Kinesis, crea reglas para asignar los datos a la secuencia de destino. La siguiente regla ilustra la operación de asignación.

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "rule-action": "include",
      "object-locator": {
        "schema-name": "Test",
        "table-name": "%"
      }
    },
    {
      "rule-type": "object-mapping",
      "rule-id": "2",
      "rule-name": "DefaultMapToKinesis",
      "rule-action": "map-record-to-record",
      "object-locator": {
        "schema-name": "Test",
        "table-name": "Customers"
      }
    }
  ]
}
```

A continuación, se ilustra el formato de registro resultante en la secuencia de datos de Kinesis.

- Nombre de secuencia: XXX
- Clave de partición: Test.Customers //schmaName.tableName
- Datos: //El siguiente mensaje JSON

```
{
  "FirstName": "Randy",
  "LastName": "Marsh",
  "StoreId": "5",
  "HomeAddress": "221B Baker Street",
  "HomePhone": "1234567890",
  "WorkAddress": "31 Spoooner Street, Quahog",
  "WorkPhone": "9876543210",
  "DateOfBirth": "02/29/1988"
}
```

## Reestructuración de datos con el mapeo de atributos

Puede reestructurar los datos mientras los migra a una secuencia de datos de Kinesis utilizando un mapa de atributos. Por ejemplo, es posible que desee combinar varios campos del origen en un único campo en el destino. El mapa de atributos siguiente ilustra cómo reestructurar los datos.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "rule-action": "include",
            "object-locator": {
                "schema-name": "Test",
                "table-name": "%"
            }
        },
        {
            "rule-type": "object-mapping",
            "rule-id": "2",
            "rule-name": "TransformToKinesis",
            "rule-action": "map-record-to-record",
            "target-table-name": "CustomerData",
            "object-locator": {
                "schema-name": "Test",
                "table-name": "Customers"
            },
            "mapping-parameters": {
                "partition-key-type": "attribute-name",
                "partition-key-name": "CustomerName",
                "exclude-columns": [
                    "firstname",
                    "lastname",
                    "homeaddress",
                    "homephone",
                    "workaddress",
                    "workphone"
                ],
                "attribute-mappings": [
                    {
                        "target-attribute-name": "CustomerName",
                        "attribute-type": "scalar",
                        "attribute-sub-type": "string",
                        "value": "${lastname}, ${firstname}"
                    },
                    {
                        "target-attribute-name": "ContactDetails",
                        "attribute-type": "document",
                        "attribute-sub-type": "json",
                        "value": {
                            "Home": {
                                "Address": "${homeaddress}",
                                "Phone": "${homephone}"
                            },
                            "Work": {
                                "Address": "${workaddress}",
                                "Phone": "${workphone}"
                            }
                        }
                    }
                ]
            }
        }
    ]
}
```

Para definir un valor constante para partition-key, especifique un valor `partition-key`. Tal vez desee hacer esto, por ejemplo, para obligar a que todos los datos se almacenen en una única partición. El siguiente mapeo ilustra este enfoque.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "1",  
            "rule-name": "1",  
            "object-locator": {  
                "schema-name": "Test",  
                "table-name": "%"  
            },  
            "rule-action": "include"  
        },  
        {  
            "rule-type": "object-mapping",  
            "rule-id": "1",  
            "rule-name": "TransformToKinesis",  
            "rule-action": "map-record-to-document",  
            "object-locator": {  
                "schema-name": "Test",  
                "table-name": "Customer"  
            },  
            "mapping-parameters": {  
                "partition-key": {  
                    "value": "ConstantPartitionKey"  
                },  
                "exclude-columns": [  
                    "FirstName",  
                    "LastName",  
                    "HomeAddress",  
                    "HomePhone",  
                    "WorkAddress",  
                    "WorkPhone"  
                ],  
                "attribute-mappings": [  
                    {  
                        "attribute-name": "CustomerName",  
                        "value": "${FirstName},${LastName}"  
                    },  
                    {  
                        "attribute-name": "ContactDetails",  
                        "value": {  
                            "Home": {  
                                "Address": "${HomeAddress}",  
                                "Phone": "${HomePhone}"  
                            },  
                            "Work": {  
                                "Address": "${WorkAddress}",  
                                "Phone": "${WorkPhone}"  
                            }  
                        }  
                    },  
                    {  
                        "attribute-name": "DateOfBirth",  
                        "value": "${DateOfBirth}"  
                    }  
                ]  
            }  
        }  
    ]  
}
```

## Formato de los mensajes de Kinesis Data Streams

La salida JSON es simplemente una lista de pares de clave-valor. AWS DMS proporciona los siguientes campos reservados para facilitar el consumo de los datos desde Kinesis Data Streams:

### RecordType

El tipo de registro puede ser de datos o de control. Los registros de datos representan las filas reales en el origen. Los registros de control son para eventos importantes de la secuencia como, por ejemplo, el reinicio de una tarea.

### Operation

Para los registros de datos, la operación puede ser `create`, `read`, `update` o `delete`.

Para los registros de control, la operación puede ser `TruncateTable` o `DropTable`.

### SchemaName

El esquema de origen del registro. Este campo puede estar vacío para un registro de control.

### TableName

La tabla de origen del registro. Este campo puede estar vacío para un registro de control.

### Marca temporal

La marca temporal que indica cuándo se creó el mensaje JSON. El campo está formateado con el formato ISO 8601.

### Note

El valor `partition-key` de un registro de control para una tabla específica es `TaskId.SchemaName.TableName`. El valor `partition-key` de un registro de control para una tabla específica es el `TaskId` de ese registro. La especificación de un valor `partition-key` en el mapeo de objetos no tiene ningún efecto en el elemento `partition-key` de un registro de control.

## Uso de un clúster de Amazon Elasticsearch Service como destino de AWS Database Migration Service

Puede utilizar AWS DMS para migrar datos a Amazon Elasticsearch Service (Amazon ES). Amazon ES es un servicio administrado que facilita la implementación, el uso y el escalado de un clúster de Elasticsearch.

En Elasticsearch, se trabaja con índices y documentos. Un índice es una colección de documentos y un documento es un objeto JSON que contiene valores escalares, matrices y otros objetos. Elasticsearch proporciona un lenguaje de consultas basado en JSON para que pueda consultar los datos de un índice y recuperar los documentos correspondientes.

Cuando AWS DMS crea índices para un punto de enlace de destino de Amazon Elasticsearch Service, crea un índice para cada tabla del punto de enlace de origen. El costo de la creación de un índice de Elasticsearch depende de varios factores. Estos factores son el número de índices creados, la cantidad total de datos en estos índices y la pequeña cantidad de metadatos que Elasticsearch almacena para cada documento.

Debe utilizar la versión del motor de AWS Database Migration Service 3.1.2 o superior para migrar datos a Amazon Elasticsearch Service.

Configure su clúster de Elasticsearch con recursos informáticos y de almacenamiento que sean adecuados para el ámbito de la migración. Le recomendamos que tenga en cuenta los factores siguientes, en función de la tarea de replicación que desee utilizar:

- Para una carga de datos completa, considere la cantidad total de datos que va a migrar, así como la velocidad de la transferencia.
- Para la replicación de cambios continuos, considere la frecuencia de las actualizaciones y los requisitos totales de latencia.

Asimismo, configure los ajustes del índice en su clúster de Elasticsearch, prestando especial atención al número de particiones y réplicas.

## Migración desde una tabla de una base de datos relacional a un índice de Amazon ES

AWS DMS permite migrar datos a tipos de datos escalares de Elasticsearch. Al migrar desde una base de datos relacional como Oracle o MySQL a Elasticsearch, puede reestructurar la manera de almacenar dichos datos.

AWS DMS admite los siguientes tipos de datos escalares de Elasticsearch:

- Booleano
- Fecha
- Float
- Int
- Cadena

AWS DMS convierte los datos de tipo Date en tipo String. Puede especificar la asignación personalizada para interpretar estas fechas.

AWS DMS no admite la migración de tipos de datos LOB.

## Requisitos previos para utilizar Amazon Elasticsearch Service como destino de AWS Database Migration Service

Antes de empezar a trabajar con una base de datos de Elasticsearch como destino de AWS DMS, asegúrese de crear un rol de AWS Identity and Access Management (IAM). Este rol debe permitir a AWS DMS obtener acceso a los índices de Elasticsearch en el punto de enlace de destino. El conjunto mínimo de permisos de acceso se muestra en la siguiente política de roles de ejemplo.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "1",  
            "Effect": "Allow",  
            "Principal": {  
                "Service": "dms.amazonaws.com"  
            },  
            "Action": "sts:AssumeRole"  
        }  
    ]  
}
```

El rol que utilice para la migración a Elasticsearch debe tener los siguientes permisos.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "es:ESHttpDelete",
                "es:ESHttpGet",
                "es:ESHttpHead",
                "es:ESHttpPost",
                "es:ESHttpPut"
            ],
            "Resource": "arn:aws:es:region:account-id:domain/domain-name/*"
        }
    ]
}
```

En el ejemplo anterior, sustituya **region** por el Identificador de la región de AWS, **account-id** por su ID de cuenta de AWS y **domain-name** por el nombre de su dominio de Amazon Elasticsearch Service. Un ejemplo es `arn:aws:es:us-west-2:123456789012:domain/my-es-domain`.

## Atributos de conexión adicionales al usar Elasticsearch como destino de AWS DMS

Cuando configure el punto de enlace de destino de Elasticsearch, puede especificar atributos de conexión adicionales. Estos atributos se especifican mediante pares de clave y valor y están separados por signos de punto y coma.

En la tabla siguiente se describen los atributos de conexión adicionales disponibles cuando se utiliza una instancia de Elasticsearch como un origen de AWS DMS.

Nombre de atributo	Valores válidos	Valor predeterminado y descripción
<code>fullLoadErrorThreshold</code>	Un número entero positivo mayor que 0, pero menor que 100.	10: para una tarea de carga completa, este atributo determina el umbral de errores permitidos antes de producirse un error en la tarea. Por ejemplo, suponga que hay 1 500 filas en el punto de enlace de origen y que este parámetro está establecido en 10. La tarea produce un error si AWS DMS encuentra más de 150 errores (10 por ciento de la fila "count") al escribir en el punto de enlace de destino.
<code>errorRetryDuration</code>	Un número entero positivo mayor que 0.	300: si se produce un error en el punto de enlace de destino, AWS DMS reintenta esta operación durante este número de segundos. De lo contrario, la tarea produce un error.

## Restricciones cuando se usa Amazon Elasticsearch Service como destino para AWS Database Migration Service

Al utilizar Amazon Elasticsearch Service como destino se aplican las siguientes restricciones:

- AWS DMS solo admite la replicación de tablas con claves principales no compuestas. La clave principal de la tabla de origen debe constar de una única columna.
- Elasticsearch utiliza el mapeo dinámico (conjetura automática) para determinar los tipos de datos que se utilizan para los datos migrados.
- Elasticsearch almacena cada documento con un identificador único. A continuación, se muestra un ID de ejemplo.

```
"_id": "D359F8B537F1888BC71FE20B3D79EAE6674BE7ACA9B645B0279C7015F6FF19FD"
```

Cada ID de documento tiene una longitud de 64 bytes, por lo que debe prever este requisito de almacenamiento. Por ejemplo, si migra 100 000 filas de un origen de AWS DMS, el índice de Elasticsearch resultante requiere almacenamiento para 6 400 000 bytes adicionales.

- Con Amazon ES, no puede realizar actualizaciones en los atributos de clave principales. Esta restricción es importante cuando se utiliza la replicación continua con captura de datos de cambio (CDC), ya que puede resultar en la presencia de datos no deseados en el destino. En modo de CDC, las claves principales se asignan a valores de SHA256, que tienen 32 bytes. Estos valores se convierten en cadenas de 64 bytes legibles por el usuario y se utilizan como identificadores de documento de Elasticsearch.
- Si AWS DMS encuentra algún elemento que no se puede migrar, escribe una lista de mensajes de error en Amazon CloudWatch Logs. Este comportamiento difiere del de otros puntos de enlace de destino de AWS DMS, que escriben los errores en una tabla de excepciones.

## Tipos de datos de destino para Amazon Elasticsearch Service

Cuando AWS DMS migra datos desde bases de datos heterogéneas, el servicio asigna los tipos de datos de la base de datos de origen a tipos de datos intermedios, llamados tipos de datos de AWS DMS. A continuación, el servicio asigna los tipos de datos intermedios a los tipos de datos de destino. La tabla siguiente muestra cada tipo de datos de AWS DMS y el tipo de datos al que se asigna en Elasticsearch.

Tipo de datos de AWS DMS	Tipo de datos de Elasticsearch
Booleano	booleano
Fecha	string
Time (Hora)	date
Marca temporal	date
INT4	integer
Real4	float
UINT4	integer

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

## Uso de Amazon DocumentDB como destino para AWS Database Migration Service

Puede usar AWS DMS para migrar datos a Amazon DocumentDB (con compatibilidad con MongoDB) desde cualquiera de los motores de datos de origen compatibles con AWS DMS. El motor de origen puede

residir en un servicio administrado por Amazon como Amazon RDS, Aurora o Amazon S3. Asimismo, el motor puede estar en una base de datos autoadministrada, como MongoDB ejecutándose en Amazon EC2 o localmente.

Puede utilizar AWS DMS para replicar los datos de origen en bases de datos, colecciones o documentos de Amazon DocumentDB.

Si el punto de enlace de origen es MongoDB, asegúrese de habilitar los siguientes atributos de conexión adicionales:

- `nestingLevel=NONE`
- `extractDocID=false`

Para obtener más información, consulte [Atributos de conexión adicionales al usar MongoDB como origen para AWS DMS \(p. 147\)](#).

MongoDB almacena los datos en formato JSON binario (BSON). AWS DMS admite todos los tipos de datos BSON compatibles con Amazon DocumentDB. Para ver una lista de estos tipos de datos, consulte [API, operaciones y tipos de datos de MongoDB admitidos](#) en la Guía para desarrolladores de Amazon DocumentDB.

Si el punto de enlace de origen es una base de datos relacional, AWS DMS establece las siguientes correspondencias de los objetos de la base de datos en Amazon DocumentDB:

- Una base de datos relacional, o un esquema de base de datos, se corresponde con una base de datos de Amazon DocumentDB.
- Las tablas de una base de datos relacional se corresponden con colecciones de Amazon DocumentDB.
- Los registros de una tabla relacional se corresponden con documentos de Amazon DocumentDB. Cada documento se crea a partir de los datos del registro de origen.

Si el punto de enlace de origen es Amazon S3, los objetos de Amazon DocumentDB resultantes se corresponden con reglas de mapeo de AWS DMS para Amazon S3. Considere, por ejemplo, el siguiente URI.

```
s3://mybucket/hr/employee
```

En este caso, AWS DMS asigna los objetos de `mybucket` a Amazon DocumentDB tal y como se indica a continuación:

- La parte del URI del nivel superior (`hr`) se corresponde con una base de datos de Amazon DocumentDB.
- La parte del URI siguiente (`employee`) se corresponde con una colección de Amazon DocumentDB.
- Cada objeto de `employee` se corresponde con un documento de Amazon DocumentDB.

Para obtener más información sobre las reglas de mapeo de Amazon S3, consulte [Uso de Amazon S3 como origen para AWS DMS \(p. 148\)](#).

Para obtener más información sobre cómo trabajar con Amazon DocumentDB como destino de AWS DMS, incluido un tutorial del proceso de migración, consulte las siguientes secciones.

#### Temas

- [Asignación de datos desde un origen a un destino de Amazon DocumentDB \(p. 234\)](#)
- [Replicación continua con Amazon DocumentDB como destino \(p. 237\)](#)
- [Restricciones al uso de Amazon DocumentDB como destino \(p. 238\)](#)

- [Tipos de datos de destino para Amazon DocumentDB \(p. 238\)](#)
- [Tutorial: Migración desde MongoDB a Amazon DocumentDB \(p. 239\)](#)

## Asignación de datos desde un origen a un destino de Amazon DocumentDB

AWS DMS lee los registros del punto de enlace de origen y crea documentos JSON en función de los datos que lee. Para cada documento JSON, AWS DMS debe determinar un campo `_id` que actúe como un identificador único. A continuación, escribe el documento JSON en una colección de Amazon DocumentDB, utilizando el campo `_id` como clave principal.

### Datos de origen que están en una sola columna

Si los datos de origen constan de una única columna, deben ser del tipo cadena. (En función del motor de origen, el tipo de datos real puede ser VARCHAR, NVARCHAR, TEXTO, LOB, CLOB o similar). AWS DMS presupone que los datos son un documento JSON válido y replica los datos en Amazon DocumentDB tal como están.

Si el documento JSON resultante contiene un campo llamado `_id`, entonces se utiliza ese campo como el `_id` único en Amazon DocumentDB.

Si el JSON no contiene un campo `_id`, Amazon DocumentDB genera un valor de `_id` automáticamente.

### Datos de origen que están en varias columnas

Si los datos de origen constan de varias columnas, AWS DMS crea un documento JSON a partir de todas estas columnas. Para determinar el campo `_id` del documento, AWS DMS procede de la siguiente manera:

- Si una de las columnas se llama `_id`, los datos de esa columna se utilizan como el `_id` de destino.
- Si no hay ninguna columna `_id`, pero los datos de origen tienen una clave principal o un índice único, AWS DMS utiliza el valor de esa clave o índice como el valor de `_id`. Los datos de la clave principal o el índice único también aparecen como campos explícitos en el documento JSON.
- Si no hay ninguna columna `_id` y tampoco hay una clave principal o un índice único, Amazon DocumentDB genera un valor de `_id` automáticamente.

### Forzar un tipo de datos en el punto de enlace de destino

AWS DMS puede modificar las estructuras de datos cuando escribe en un punto de enlace de destino de Amazon DocumentDB. Puede solicitar estos cambios modificando los nombres de las columnas y las tablas en el punto de enlace de origen o proporcionando reglas de transformación que se apliquen cuando se ejecute una tarea.

#### Uso de un documento JSON anidado (`json_Prefix`)

Para forzar un tipo de datos, puede incluir el prefijo `json_` en el nombre de la columna de origen (por ejemplo, `json_columnName`) manualmente o mediante una transformación. En este caso, la columna se crea como un documento JSON anidado dentro del documento de destino, en lugar de como un campo de cadena.

Suponga, por ejemplo, que desea migrar el siguiente documento desde un punto de enlace de origen de MongoDB.

```
{
```

```
{  
    "_id": "1",  
    "FirstName": "John",  
    "LastName": "Doe",  
    "ContactDetails": "{\"Home\": {\"Address\": \"Boston\", \"Phone\": \"11111111\"}, \"Work\":  
    { \"Address\": \"Boston\", \"Phone\": \"2222222222\"}}"  
}
```

Si no fuerza ningún tipo de datos de origen, el documento `ContactDetails` incrustado se migra como una cadena.

```
{  
    "_id": "1",  
    "FirstName": "John",  
    "LastName": "Doe",  
    "ContactDetails": "{\"Home\": {\"Address\": \"Boston\", \"Phone\": \"11111111\"}, \"Work\":  
    { \"Address\": \"Boston\", \"Phone\": \"2222222222\"}}"  
}
```

Sin embargo, puede añadir una regla de transformación para obligar a que `ContactDetails` sea un objeto JSON. Suponga, por ejemplo, que el nombre original de la columna de origen es `ContactDetails`. Suponga también que la columna de origen con el nuevo nombre va a ser `json_ContactDetails`. AWS DMS replica el campo `ContactDetails` como JSON anidado, tal y como se indica a continuación.

```
{  
    "_id": "1",  
    "FirstName": "John",  
    "LastName": "Doe",  
    "ContactDetails": {  
        "Home": {  
            "Address": "Boston",  
            "Phone": "1111111111"  
        },  
        "Work": {  
            "Address": "Boston",  
            "Phone": "2222222222"  
        }  
    }  
}
```

### Uso de una matriz JSON (array\_Prefix)

Para forzar un tipo de datos, puede incluir el prefijo `array_` en el nombre de la columna (por ejemplo, `array_columnName`) manualmente o mediante una transformación. En este caso, AWS DMS considera la columna como una matriz JSON y la crea como tal en el documento de destino.

Suponga que desea migrar el siguiente documento desde un punto de enlace de origen de MongoDB.

```
{  
    "_id" : "1",  
    "FirstName": "John",  
    "LastName": "Doe",  
  
    "ContactAddresses": ["Boston", "New York"],  
  
    "ContactPhoneNumbers": ["1111111111", "2222222222"]  
}
```

Si no fuerza ningún tipo de datos de origen, el documento `ContactDetails` incrustado se migra como una cadena.

```
{  
    "_id": "1",  
    "FirstName": "John",  
    "LastName": "Doe",  
  
    "ContactAddresses": "[\"Boston\", \"New York\"]",  
  
    "ContactPhoneNumbers": "[\"1111111111\", \"2222222222\"]"  
}
```

Sin embargo, puede añadir reglas de transformación para imponer el tipo de datos de matriz JSON a `ContactAddress` y `ContactPhoneNumbers`, tal y como se muestra en la siguiente tabla.

Nombre original de la columna de origen	Nuevo nombre de la columna de origen
<code>ContactAddress</code>	<code>array_ContactAddress</code>
<code>ContactPhoneNumbers</code>	<code>array_ContactPhoneNumbers</code>

AWS DMS replica `ContactAddress` y `ContactPhoneNumbers` de la manera siguiente.

```
{  
    "_id": "1",  
    "FirstName": "John",  
    "LastName": "Doe",  
    "ContactAddresses": [  
        "Boston",  
        "New York"  
    ],  
    "ContactPhoneNumbers": [  
        "1111111111",  
        "2222222222"  
    ]  
}
```

## Conexión a Amazon DocumentDB mediante TLS

De forma predeterminada, un clúster de Amazon DocumentDB recién creado solo acepta conexiones seguras mediante Transport Layer Security (TLS). Cuando TLS está habilitado, cada conexión a Amazon DocumentDB requiere una clave pública.

Puede descargar la clave pública de Amazon DocumentDB como el archivo `rds-combined-ca-bundle.pem` de un bucket de Amazon S3 alojado en AWS.

Después de descargar este archivo.pem, puede importar el archivo en AWS DMS como se describe a continuación.

### Consola de administración de AWS

Para importar el archivo de clave pública (.pem)

1. Abra la consola de AWS DMS en <https://console.aws.amazon.com/dms>.
2. En el panel de navegación, elija Certificates.
3. Elija Import certificate (Importar certificado) y haga lo siguiente:
  - En Certificate identifier (Identificador del certificado), escriba un nombre único para el certificado (por ejemplo `docdb-cert`).

- En Import file (Archivo de importación), desplácese hasta la ubicación en la que guardó el archivo .pem.

Cuando los ajustes sean los deseados, elija Add new CA certificate (Añadir nuevo certificado de CA).

## AWS CLI

Utilice el comando `aws dms import-certificate`, tal y como se muestra en el ejemplo siguiente.

```
aws dms import-certificate \
    --certificate-identifier docdb-cert \
    --certificate-pem file://./rds-combined-ca-bundle.pem
```

Cuando cree un punto de enlace de destino de AWS DMS, proporcione el identificador de certificado (por ejemplo, `docdb-cert`). A continuación, establezca el parámetro de modo de SSL en `verify-full`.

## Replicación continua con Amazon DocumentDB como destino

Si la replicación continua está habilitada, AWS DMS se asegura de que los documentos de Amazon DocumentDB se mantengan sincronizados con el origen. Cuando se crea o actualiza un registro de origen, AWS DMS debe determinar primero qué registro de Amazon DocumentDB resultará afectado mediante el siguiente procedimiento:

- Si el registro de origen tiene una columna denominada `_id`, el valor de esa columna determina el `_id` correspondiente en la colección de Amazon DocumentDB.
- Si no hay ninguna columna `_id`, pero los datos de origen tienen una clave principal o un índice único, AWS DMS utiliza el valor de esa clave o índice como el `_id` para la colección de Amazon DocumentDB.
- Si el registro de origen no tiene una columna `_id`, una clave principal ni un índice único, AWS DMS asigna todas las columnas de origen a los campos correspondientes de la colección de Amazon DocumentDB.

Cuando se crea un nuevo registro de origen, AWS DMS escribe un documento correspondiente en Amazon DocumentDB. Si se actualiza un registro de origen existente, AWS DMS actualiza los campos correspondientes del documento de destino de Amazon DocumentDB. Los campos que existen en el documento de destino pero no en el registro de origen se mantienen tal cual están.

Cuando se elimina un registro de origen, AWS DMS elimina el documento correspondiente de Amazon DocumentDB.

## Cambios estructurales (DDL) en el origen

Con la replicación continua, todos los cambios realizados en las estructuras de datos de origen (como tablas, columnas, etc.) se propagan a sus elementos correspondientes en Amazon DocumentDB. En las bases de datos relacionales, estos cambios se iniciaron con instrucciones del lenguaje de definición de datos (DDL). Puede ver cómo AWS DMS propaga estos cambios a Amazon DocumentDB en la siguiente tabla.

DDL en el origen	Efecto en el elemento de Amazon DocumentDB de destino
<code>CREATE TABLE</code>	Crea una colección vacía.
Instrucción que cambia el nombre de una tabla ( <code>RENAME TABLE</code> , <code>ALTER TABLE...RENAME</code> y similar)	Cambia el nombre de la colección.

DDL en el origen	Efecto en el elemento de Amazon DocumentDB de destino
TRUNCATE TABLE	Elimina todos los documentos de la colección, pero solo si <code>HandleSourceTableTruncated</code> es <code>true</code> . Para obtener más información, consulte <a href="#">Configuración de tareas para la administración de DDL del procesamiento de cambios (p. 272)</a> .
DROP TABLE	Elimina la colección, pero solo si <code>HandleSourceTableDropped</code> es <code>true</code> . Para obtener más información, consulte <a href="#">Configuración de tareas para la administración de DDL del procesamiento de cambios (p. 272)</a> .
Instrucción que añade una columna a una tabla ( <code>ALTER TABLE...ADD</code> y similar)	La instrucción DDL se omite y se envía una advertencia. Cuando se ejecuta la primera instrucción <code>INSERT</code> en el origen, se añade el nuevo campo al documento de destino.
ALTER TABLE...RENAME COLUMN	La instrucción DDL se omite y se envía una advertencia. Cuando se ejecuta la primera instrucción <code>INSERT</code> en el origen, el campo con el nuevo nombre se añade al documento de destino.
ALTER TABLE...DROP COLUMN	La instrucción DDL se omite y se envía una advertencia.
Instrucción que cambia el tipo de datos de las columnas ( <code>ALTER COLUMN...MODIFY</code> y similar)	La instrucción DDL se omite y se envía una advertencia. Cuando se ejecuta la primera instrucción <code>INSERT</code> en el origen con el nuevo tipo de datos, se crea el documento de destino con un campo de ese nuevo tipo de datos.

## Restricciones al uso de Amazon DocumentDB como destino

Al utilizar Amazon DocumentDB como destino de AWS DMS se aplican las siguientes restricciones:

- En Amazon DocumentDB, los nombres de las colecciones no pueden incluir el símbolo del dólar (\$). Además, los nombres de las bases de datos no pueden contener caracteres Unicode.
- AWS DMS no permite combinar varias tablas de origen en una sola colección de Amazon DocumentDB.
- Cuando AWS DMS procesa los cambios de una tabla de origen que no dispone de una clave principal, las columnas LOB de esa tabla se omiten.
- Si la opción Change table (Tabla de cambios) está habilitada y AWS DMS encuentra una columna de origen llamada `_id`, esa columna aparece como `_id` (con dos caracteres de subrayado) en la tabla de cambios.
- Si elige Oracle como punto de enlace de origen, el origen de Oracle debe tener el registro suplementario completo habilitado. De lo contrario, si hay columnas en el origen que no han cambiado, los datos se cargan en Amazon DocumentDB como valores nulos.

## Tipos de datos de destino para Amazon DocumentDB

En la tabla siguiente, se indican los tipos de datos de destino de Amazon DocumentDB que se admiten cuando se utiliza AWS DMS y la correspondencia predeterminada con los tipos de datos de AWS DMS.

Para obtener más información sobre los tipos de datos de AWS DMS, consulte [Tipos de datos para AWS Database Migration Service \(p. 393\)](#).

Tipo de datos de AWS DMS	Tipo de datos de Amazon DocumentDB
BOOLEANO	Booleano
BYTES	Datos binarios
FECHA	Fecha
TIME	Cadena (UTF8)
DATETIME	Fecha
INT1	Entero de 32 bits
INT2	Entero de 32 bits
INT4	Entero de 32 bits
INT8	Entero de 64 bits
NUMERIC	Cadena (UTF8)
REAL4	Double
REAL8	Double
STRING	Si los datos se reconocen como JSON, AWS DMS los migra a Amazon DocumentDB como un documento. De lo contrario, los datos se corresponden con una cadena (UTF8).
UINT1	Entero de 32 bits
UINT2	Entero de 32 bits
UINT4	Entero de 64 bits
UINT8	Cadena (UTF8)
WSTRING	Si los datos se reconocen como JSON, AWS DMS los migra a Amazon DocumentDB como un documento. De lo contrario, los datos se corresponden con una cadena (UTF8).
BLOB	Binario
CLOB	Si los datos se reconocen como JSON, AWS DMS los migra a Amazon DocumentDB como un documento. De lo contrario, los datos se corresponden con una cadena (UTF8).
NCLOB	Si los datos se reconocen como JSON, AWS DMS los migra a Amazon DocumentDB como un documento. De lo contrario, los datos se corresponden con una cadena (UTF8).

## Tutorial: Migración desde MongoDB a Amazon DocumentDB

Utilice el siguiente tutorial como guía durante el proceso de migración de MongoDB a Amazon DocumentDB (con compatibilidad con MongoDB). En este tutorial, hará lo siguiente:

- Instalar MongoDB en una instancia Amazon EC2
- Rellenar MongoDB con datos de ejemplo
- Crear una instancia de replicación de AWS DMS, un punto de enlace de origen (para MongoDB) y un punto de enlace de destino (para Amazon DocumentDB)
- Ejecutar una tarea de AWS DMS para migrar los datos del punto de enlace de origen al punto de enlace de destino

#### Important

Antes de comenzar, asegúrese de lanzar un clúster de Amazon DocumentDB en su nube virtual privada (VPC). Para obtener más información, consulte [Introducción](#) en la Guía para desarrolladores de Amazon DocumentDB.

#### Temas

- [Paso 1: Lanzar una instancia Amazon EC2 \(p. 240\)](#)
- [Paso 2: Instalar y configurar MongoDB Community Edition \(p. 241\)](#)
- [Paso 3: Crear una instancia de replicación de AWS DMS \(p. 242\)](#)
- [Paso 4: Crear los puntos de enlace de origen y destino \(p. 243\)](#)
- [Paso 5: Crear y ejecutar una tarea de migración \(p. 245\)](#)

## Paso 1: Lanzar una instancia Amazon EC2

En este tutorial, lanzará una instancia Amazon EC2 en la VPC predeterminada.

### Para lanzar una instancia Amazon EC2

1. Abra la consola de Amazon EC2 en <https://console.aws.amazon.com/ec2/>.
2. Elija Launch Instance (Lanzar instancia) y proceda del modo siguiente:
  - a. En la página Choose an Amazon Machine Image (AMI) (Elegir una imagen de máquina de Amazon (AMI)), en la parte superior de la lista de AMI, vaya a Amazon Linux AMI (AMI de Amazon Linux) y elija Select (Seleccionar).
  - b. En la página Choose an Instance Type (Elegir un tipo de instancia), en la parte superior de la lista de tipos de instancias, elija t2.micro. A continuación, elija Next: Configure Instance Details.
  - c. En la página Configure Instance Details (Configurar detalles de la instancia), en Network (Red), seleccione su VPC predeterminada. A continuación, elija Next: Add Storage (Siguiente: Añadir almacenamiento).
  - d. En la página Add Storage (Añadir almacenamiento), omita este paso y elija Next: Add Tags (Siguiente: Añadir etiquetas).
  - e. En la página Add Tags (Añadir etiquetas), omita este paso y elija Next: Configure Security Group (Siguiente: Configurar grupo de seguridad).
  - f. En la página Configure Security Group (Configurar grupo de seguridad), haga lo siguiente:
    1. Elija Select an existing security group.
    2. En la lista de grupos de seguridad, elija default. De esta forma, se elige el grupo de seguridad predeterminado para la VPC. De forma predeterminada, el grupo de seguridad acepta conexiones SSH (Secure Shell) entrantes en el puerto TPC 22. Si este no es el caso para su VPC, añada esta regla. Para obtener más información, consulte [¿Qué es Amazon VPC?](#) en la Guía del usuario de Amazon VPC.
    3. Elija Next: Review and Launch.
  - g. Revise la información y elija Launch (Lanzar).
3. En la ventana Select an existing key pair or create a new key pair, proceda del modo siguiente:

- Si no dispone de un par de claves de Amazon EC2, elija Create a new key pair (Crear un nuevo par de claves) y siga las instrucciones. Se le pedirá que descargue un archivo de clave privada (archivo .pem). Necesitará este archivo más tarde cuando inicie sesión en su instancia Amazon EC2.
  - Si ya dispone de un par de claves de Amazon EC2 en Select a key pair (Seleccionar un par de claves), elija ese par de claves en la lista. Debe disponer del archivo de clave privada (archivo .pem) para poder iniciar sesión en la instancia Amazon EC2.
4. Cuando haya configurado el par de claves, elija Launch Instances (Lanzar instancias).

En el panel de navegación de la consola, elija EC2 Dashboard (Panel de EC2) y, a continuación, seleccione la instancia que ha lanzado. En el panel inferior, en la pestaña Description (Descripción), busque el valor de Public DNS (DNS público) de su instancia (por ejemplo, ec2-11-22-33-44.us-west-2.compute.amazonaws.com).

La instancia Amazon EC2 tardará unos minutos en estar disponible.

5. Utilice el comando ssh para iniciar sesión en su instancia Amazon EC2, como se muestra en el siguiente ejemplo.

```
chmod 400 my-keypair.pem
ssh -i my-keypair.pem ec2-user@public-dns-name
```

Especifique el archivo de clave privada (archivo .pem) y el nombre DNS público de la instancia EC2. El identificador de inicio de sesión es ec2-user. No se requiere contraseña.

Para obtener más información acerca de la conexión a una instancia EC, consulte [Conexión a la instancia de Linux mediante SSH](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

## Paso 2: Instalar y configurar MongoDB Community Edition

Realice estos pasos en la instancia Amazon EC2 que lanzó en [Paso 1: Lanzar una instancia Amazon EC2 \(p. 240\)](#).

Para instalar y configurar MongoDB Community Edition en su instancia EC2

1. Consulte el tutorial de [instalación de MongoDB Community Edition en Amazon Linux](#) en la documentación de MongoDB y siga las instrucciones que se detallan allí.
2. De forma predeterminada, el servidor de MongoDB (`mongod`) solo permite conexiones de bucle invertido desde la dirección IP 127.0.0.1 (localhost). Para permitir las conexiones desde otro lugar de su Amazon VPC, haga lo siguiente:
  - a. Edite el archivo `/etc/mongod.conf` y busque las siguientes líneas.

```
# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1 # Enter 0.0.0.0,:: to bind to all IPv4 and IPv6 addresses or,
  alternatively, use the net.bindIpAll setting.
```

- b. Modifique la línea `bindIp` de forma que tenga el aspecto siguiente.

```
bindIp: public-dns-name
```

- c. Sustituya `public-dns-name` por el nombre DNS público real de su instancia (por ejemplo, `ec2-11-22-33-44.us-west-2.compute.amazonaws.com`).
- d. Guarde el archivo `/etc/mongod.conf` y después reinicie `mongod`.

```
sudo service mongod restart
```

3. Rellene su instancia de MongoDB con datos mediante alguno de los siguientes procedimientos:

- a. Utilice el comando `wget` para descargar un archivo JSON que contenga datos de ejemplo.

```
wget http://media.mongodb.org/zips.json
```

- b. Utilice el comando `mongoimport` para importar los datos en una nueva base de datos (`zips-db`).

```
mongoimport --host public-dns-name:27017 --db zips-db --file zips.json
```

- c. Despu s de que la importaci n haya terminado, utilice el shell de `mongo` para conectarse a MongoDB y verificar que los datos se cargaron correctamente.

```
mongo --host public-dns-name:27017
```

- d. Sustituya `public-dns-name` por el nombre DNS p blico real de su instancia.
  - e. En el s mbolo del sistema del shell de `mongo`, introduzca los siguientes comandos:

```
use zips-db
db.zips.count()
db.zips.aggregate([
  { $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },
  { $group: { _id: "$_id.state", avgCityPop: { $avg: "$pop" } } }
])
```

El resultado deber a ser el siguiente:

- El nombre de la base de datos (`zips-db`).
  - El n mero de documentos de la colecci n `zips` (29353)
  - La poblaci n media de las ciudades de cada estado
- f. Salga del shell de `mongo` y vuelva al s mbolo del sistema con el siguiente comando.

```
exit
```

## Paso 3: Crear una instancia de replicaci n de AWS DMS

Para realizar una replicaci n en AWS DMS, necesita una instancia de replicaci n.

Para crear una instancia de replicaci n de AWS DMS

1. Abra la consola de AWS DMS en <https://console.aws.amazon.com/dms/>.
2. En el panel de navegaci n, elija Replication instances.
3. Elija Create replication instance (Crear instancia de replicaci n) y escriba la siguiente informaci n:
  - En Name (Nombre), escriba `mongodb2docdb`.
  - En Description (Descripci n), escriba `MongoDB to Amazon DocumentDB replication instance`.
  - En Instance class (Clase de instancia), mantenga el valor predeterminado.

- En Engine version (Versión del motor), mantenga el valor predeterminado.
- En VPC, seleccione su VPC predeterminada.
- En Multi-AZ, elija No.
- En Publicly accessible (Accesible públicamente), habilite esta opción.

Cuando esté conforme con los ajustes, elija Create replication instance (Crear instancia de replicación).

#### Note

Puede comenzar a utilizar su instancia de replicación cuando su estado sea available (disponible). Esto puede tardar varios minutos.

## Paso 4: Crear los puntos de enlace de origen y destino

El punto de enlace de origen es el punto de enlace de la instalación de MongoDB que se ejecuta en su instancia Amazon EC2.

Para crear un punto de enlace de origen

1. Abra la consola de AWS DMS en <https://console.aws.amazon.com/dms/>.
2. En el panel de navegación, elija Endpoints.
3. Elija Create endpoint (Crear punto de enlace) y escriba la siguiente información:
  - En Endpoint Type (Tipo de punto de enlace), elija Source (Origen).
  - En Endpoint identifier (Identificador de punto de enlace), escriba un nombre que sea fácil de recordar, como `mongodb-source`.
  - En Source engine (Motor de origen), elija `mongodb`.
  - En Server name (Nombre de servidor), escriba el nombre DNS público de su instancia Amazon EC2 (por ejemplo, `ec2-11-22-33-44.us-west-2.compute.amazonaws.com`).
  - En Port (Puerto), escriba `27017`.
  - En SSL mode (Modo de SSL), elija `none` (ninguno).
  - En Authentication mode (Modo de autenticación), elija `none` (ninguno).
  - En Database name (Nombre de base de datos), escriba `zips-db`.
  - En Authentication mechanism (Mecanismo de autenticación), elija `default` (predeterminado).
  - En Metadata mode (Modo de metadatos), elija `document` (documento).

Cuando esté conforme con los ajustes, elija Create endpoint (Crear punto de enlace).

A continuación, creará un punto de enlace de destino. Este punto de enlace es para el clúster de Amazon DocumentDB, que ya debe estar ejecutándose. Para obtener más información sobre cómo lanzar su clúster de Amazon DocumentDB, consulte [Introducción](#) en la Guía para desarrolladores de Amazon DocumentDB.

#### Important

Antes de comenzar, haga lo siguiente:

- Tenga a mano el nombre de usuario y la contraseña maestros del clúster de Amazon DocumentDB.
- Tenga a mano el nombre DNS y el número de puerto del clúster de Amazon DocumentDB, de forma que AWS DMS pueda conectarse a él. Para determinar esta información, utilice el

siguiente comando de la AWS CLI, sustituyendo *cluster-id* por el nombre de su clúster de Amazon DocumentDB.

```
aws docdb describe-db-clusters \
--db-cluster-identifier cluster-id \
--query "DBClusters[*].[Endpoint,Port]"
```

- Descargue un paquete de certificados que Amazon DocumentDB pueda usar para verificar las conexiones SSL. Para ello, ejecute el siguiente comando.

```
wget https://s3.amazonaws.com/rds-downloads/rds-combined-ca-bundle.pem
```

Para crear un punto de enlace de destino

1. En el panel de navegación, elija Endpoints.
2. Elija Create endpoint (Crear punto de enlace) y escriba la siguiente información:
  - En Endpoint Type (Tipo de punto de enlace), elija Target (Destino).
  - En Endpoint identifier (Identificador de punto de enlace), escriba un nombre que sea fácil de recordar, como *docdb-target*.
  - En Target engine (Motor de destino), elija docdb.
  - En Server name (Nombre de servidor), escriba el nombre DNS de su clúster de Amazon DocumentDB.
  - En Port (Puerto), escriba el número de puerto de su clúster de Amazon DocumentDB.
  - En SSL mode (Modo de SSL), elija verify-full.
  - En CA certificate (Certificado de CA), elija Add new CA certificate (Añadir nuevo certificado de CA) y, en Certificate Identifier (Identificador del certificado), escriba *rds-combined-ca-bundle*. En Import file (Archivo de importación), desplácese hasta el archivo *rds-combined-ca-bundle.pem* que ha descargado. Cuando haya terminado, elija Add new CA certificate (Añadir nuevo certificado de CA).
  - En User name (Nombre de usuario), escriba el nombre de usuario maestro de su clúster de Amazon DocumentDB.
  - En Password (Contraseña), escriba la contraseña maestra de su clúster de Amazon DocumentDB.
  - En Database name (Nombre de base de datos), escriba *zips-db*.

Cuando esté conforme con los ajustes, elija Create endpoint (Crear punto de enlace).

Ahora que ha creado los puntos de enlace de origen y de destino, pruébelos para asegurarse de que funcionan correctamente. Además, para asegurarse de que AWS DMS pueda obtener acceso a los objetos de la base de datos en cada punto de enlace, actualice los esquemas de los puntos de enlace.

Para probar un punto de enlace

1. En el panel de navegación, elija Endpoints.
2. Elija el punto de enlace de origen (*mongodb-source*) y, a continuación, elija Test connection (Probar conexión).
3. Elija su instancia de replicación (*mongodb2docdb*) y después elija Run test (Ejecutar prueba). La prueba y el cambio de Status (Estado) a successful (correcto) tardan algunos minutos en completarse.

Si Status (Estado) cambia a failed (error), revise el mensaje de error. Corrija los errores y pruebe de nuevo el punto de enlace.

#### Note

Repita este procedimiento para el punto de enlace de destino (docdb-target).

#### Para actualizar los esquemas

1. En el panel de navegación, elija Endpoints.
2. Elija el punto de enlace de origen (mongodb-source) y, a continuación, elija Refresh schemas (Actualizar esquemas).
3. Elija su instancia de replicación (mongodb2docdb) y después elija Refresh schemas (Actualizar esquemas).

#### Note

Repita este procedimiento para el punto de enlace de destino (docdb-target).

### Paso 5: Crear y ejecutar una tarea de migración

Ahora está listo para lanzar una tarea de migración de AWS DMS para migrar los datos de zips de MongoDB a Amazon DocumentDB.

1. Abra la consola de AWS DMS en <https://console.aws.amazon.com/dms/>.
2. En el panel de navegación, elija Tasks.
3. Elija Create task (Crear tarea) y escriba la siguiente información:
  - En Task name (Nombre de tarea), escriba un nombre que sea fácil de recordar, como my-dms-task.
  - En Replication instance (Instancia de replicación), elija la instancia de replicación que creó en [Paso 3: Crear una instancia de replicación de AWS DMS \(p. 242\)](#).
  - En Source endpoint (Punto de enlace de origen), elija el punto de enlace de origen que creó en [Paso 4: Crear los puntos de enlace de origen y destino \(p. 243\)](#).
  - En Target endpoint (Punto de enlace de destino), elija el punto de enlace de destino que creó en [Paso 4: Crear los puntos de enlace de origen y destino \(p. 243\)](#).
  - En Migration type (Tipo de migración), elija Migrate existing data (Migrar datos existentes).
  - En Start task on create (Iniciar tarea al crear), habilite esta opción.

En la sección Task Settings (Configuración de la tarea), mantenga los valores predeterminados de todas las opciones.

En la sección Table mappings (Mapeos de tabla), elija la pestaña Guided (Guiados) y escriba la siguiente información:

- En Schema name is (El nombre de esquema es), elija Escribir un esquema.
- En Schema name is like (El nombre de esquema es como), mantenga el valor predeterminado (%).
- En Table name is like (El nombre de tabla es como), mantenga el valor predeterminado (%).

Elija Add selection rule (Añadir regla de selección) para confirmar que la información es correcta.

Cuando esté conforme con los ajustes, elija Create task (Crear tarea).

AWS DMS comienza ahora a migrar los datos de MongoDB a Amazon DocumentDB. El estado de la tarea cambia de Starting (Iniciándose) a Running (En ejecución). Puede monitorizar el progreso eligiendo Tasks

(Tareas) en la consola de AWS DMS. Al cabo de varios minutos, el estado cambia a Load complete (Carga completada).

Note

Una vez completada la migración, puede utilizar el shell de mongo para conectarse a su clúster de Amazon DocumentDB y ver los datos de zips. Para obtener más información, consulte [Obtener acceso a su clúster de Amazon DocumentDB mediante el shell de mongo](#) en la Guía para desarrolladores de Amazon DocumentDB.

## Creación de puntos de enlace de origen y de destino

Puede crear puntos de enlace de origen y de destino al crear su instancia de replicación o puede crear puntos de enlace después de que su instancia de replicación se haya creado. Los almacenes de datos de origen y de destino pueden residir en una instancia de Amazon Elastic Compute Cloud (Amazon EC2), una instancia de base de datos de Amazon Relational Database Service (Amazon RDS) o en una base de datos local.

En el procedimiento siguiente, se presupone que va a utilizar el asistente de la consola de AWS DMS. Tenga en cuenta que también puede efectuar este paso si selecciona Endpoints (Puntos de enlace) en el panel de navegación de la consola de AWS DMS y, a continuación, selecciona Create endpoint (Crear punto de enlace). Cuando se utiliza el asistente de la consola, debe crear los puntos de enlace de origen y de destino en la misma página. Si no utiliza el asistente de la consola, debe crear cada uno de los puntos de enlace por separado.

Para especificar los puntos de enlace de la base de datos de origen o de destino a con la consola de AWS

1. En la página Connect source and target database endpoints, especifique la información de conexión para la base de datos de origen o destino. La tabla siguiente describe la configuración.

Endpoint type\*  Source  Target  ⓘ

Select RDS DB Instance   ⓘ

Endpoint identifier\*   ⓘ

Source engine\* postgres  ⓘ

Server name\*   ⓘ

Port\* 8199  ⓘ

SSL mode\* - Select One -  ⓘ

User name\*   ⓘ

Password\*   ⓘ

Database name\*

» Advanced

Para esta opción	Haga lo siguiente
Tipo de punto de conexión	Elija si este punto de enlace es el punto de enlace de origen o de destino.
Select RDS DB Instance (Seleccionar instancia de base de datos de RDS)	Seleccione esta opción si el punto de enlace es una instancia de base de datos de Amazon RDS.
Endpoint identifier	Escriba el nombre que desea utilizar para identificar el punto de enlace. Es posible que le interese incluir en el nombre del tipo de punto de enlace, como <b>oracle-source</b> o <b>PostgreSQL-target</b> . El nombre debe ser único para todas las instancias de replicación.
Source engine y Target engine	Elija el tipo de motor de base de datos que vaya a ser el punto de enlace.
Server name	Escriba el nombre del servidor. Para las bases de datos locales, puede ser la dirección IP o el nombre de host público. Para una instancia de base de datos Amazon RDS, puede ser el punto de enlace (también denominado nombre de DNS) de la instancia de base de datos, como <b>mysqlsvrinst.abcd12345678.us-west-2.rds.amazonaws.com</b> .

Para esta opción	Haga lo siguiente
Puerto	Escriba el puerto que utiliza la base de datos.
SSL mode	Elija un modo SSL si desea habilitar el cifrado de conexión para este punto de enlace. En función del modo seleccionado, es posible que se le solicite proporcionar información sobre el certificado y el certificado de servidor.
Nombre de usuario	Escriba el nombre de usuario con los permisos necesarios para permitir la migración de datos. Para obtener información sobre los permisos necesarios, consulte la sección de seguridad para el motor de base de datos de origen o destino en esta guía del usuario.
Contraseña	Escriba la contraseña para la cuenta con los permisos necesarios. Si desea utilizar caracteres especiales en la contraseña, como "+" o "&", incluya la totalidad de la contraseña entre llaves "{}".
Nombre de base de datos	El nombre de la base de datos que desea utilizar como punto de enlace.

2. Seleccione la pestaña Advanced que se muestra a continuación para establecer los valores para la cadena de conexión y la clave de cifrado en caso de que sea necesario. Puede probar la conexión del punto de enlace si selecciona Run test.

▼ Advanced

Extra connection attributes  ⓘ

KMS master key  ⓘ

Description Default master key that protects my DMS replication instance volumes when no other key is defined

Account

Key ARN

▼ Test endpoint connection (optional)

Test your endpoint connection by selecting a replication instance within your desired VPC. After clicking with the details provided and attempt to connect to the instance. If the connection fails, you can edit or will be deleted.

VPC\*  ⓘ

Replication instance\*  ⓘ

Refresh schemas after successful ⓘ

Para esta opción	Haga lo siguiente
Atributos de conexión adicionales	Escriba los parámetros de conexión adicionales aquí. Para obtener más información sobre los atributos de conexión adicionales, consulte la sección de documentación relacionada con su almacén de datos.
Clave maestra de KMS	Elija la clave de cifrado que se utilizará para cifrar el almacenamiento de la replicación y la información de la conexión. Si elige (Default) aws/dms , se utilizará la clave de AWS Key Management Service (AWS KMS) asociada con su cuenta y región. Para obtener más información sobre cómo utilizar la clave de cifrado, consulte <a href="#">Configuración de una clave de cifrado y especificación de permisos KMS (p. 49)</a> .
Test endpoint connection (optional) (Probar la conexión del punto de enlace (opcional))	Añada la VPC y el nombre de instancia de replicación. Para probar la conexión, elija Run test (Ejecutar prueba).

# Trabajar con tareas de AWS DMS

En la tarea AWS Database Migration Service (AWS DMS) se desarrolla todo el trabajo. Especifica qué tablas y esquemas utilizar para su migración y cualquier procesamiento especial, como, por ejemplo, los requisitos de registro, los datos de la tabla de control y la gestión de errores.

Al crear una tarea de migración es necesario saber algunas cosas:

- Antes de crear una tarea de migración, tiene que crear un punto de enlace de origen, un punto de enlace de destino y una instancia de replicación.
- Puede especificar muchos valores de configuración de tareas para adaptar su tarea de migración. Puede configurar estas usando la Consola de administración de AWS, AWS Command Line Interface (AWS CLI) o la API de AWS DMS. Entre estas opciones se incluyen la especificación de cómo se tratan los errores de migración, el registro de errores y la información de la tabla de control.
- Una vez que creada una tarea, puede ejecutarla inmediatamente. Las tablas de destino con las definiciones de metadatos necesarias se crean y cargan automáticamente, y puede especificar que se inicie el proceso de replicación continua.
- De forma predeterminada, AWS DMS comenzará a ejecutar su tarea nada más la cree. Sin embargo, en algunas situaciones, es posible que desee aplazar el inicio de la tarea. Por ejemplo, al usar la AWS CLI, puede tener un proceso que cree una tarea y otro distinto que la comience cuando se produzca un evento de activación. Si es necesario, puede aplazar el inicio de la tarea.
- Puede monitorear, detener o reiniciar las tareas mediante la consola de AWS DMS, la AWS CLI o la API de AWS DMS.

A continuación, se indican las acciones que puede realizar cuando trabaja con una tarea de AWS DMS.

Tarea	Documentación relacionada
Creación de un informe de evaluación de tarea  Puede crear un informe de evaluación de tarea que muestra cualquier tipo de dato no admitido que podría provocar problemas durante la migración. Puede ejecutar este informe en la tarea antes de ejecutar la tarea para encontrar posibles problemas.	<a href="#">Creación de un informe de evaluación de tarea (p. 252)</a>
Creación de una tarea  Al crear una tarea, especifique el origen, el destino y la instancia de replicación, junto con cualquier configuración de migración.	<a href="#">Creación de una tarea (p. 254)</a>
Creación de una tarea de replicación continua  Puede configurar una tarea para proporcionar replicación continua entre el origen y el destino.	<a href="#">Creación de tareas para la replicación continua con AWS DMS (p. 277)</a>

Tarea	Documentación relacionada
Aplicación de la configuración de las tareas  Cada tarea tiene valores que pueden configurar en función de las necesidades de la migración de la base de datos. Puede crear estos ajustes en un archivo JSON o, con algunos ajustes, puede especificar la configuración con la consola de AWS DMS.	<a href="#">Especificación de la configuración de tareas para tareas de AWS Database Migration Service (p. 260)</a>
Validación de datos  La validación de datos es una configuración de la tarea que puede utilizar para que AWS DMS compare los datos en el almacén de datos de destino con los datos de su almacén de datos de origen.	<a href="#">Validación de tareas de AWS DMS (p. 334).</a>
Modificación de una tarea  Cuando se detiene una tarea, puede modificar sus valores de configuración.	<a href="#">Modificar una tarea (p. 280)</a>
Recarga de tablas durante una tarea  Puede volver a cargar una tabla mientras se realiza una tarea si se produce un error durante esta tarea.	<a href="#">Volver a cargar tablas durante una tarea (p. 280)</a>
Uso del asignación de tablas  La correspondencia de tablas utiliza diversos tipos de reglas para especificar el origen de datos, el esquema origen, los datos y las transformaciones que deben producirse durante la tarea.	Reglas de selección <a href="#">Reglas de selección y acciones (p. 288)</a>  Reglas de transformación <a href="#">Reglas y acciones de transformación (p. 291)</a>
Aplicación de filtros  Puede utilizar filtros de origen para limitar el número y el tipo de los registros transferidos desde el origen al destino. Por ejemplo, puede especificar que solo los trabajadores con una ubicación de sede central se trasladen a la base de datos de destino. Puede aplicar filtros en una columna de datos.	<a href="#">Usar filtros de origen (p. 317)</a>
Monitorización de una tarea  Hay varias formas de obtener información sobre el desempeño de una tarea y las tablas que utiliza esta tarea.	<a href="#">Monitorización de tareas de AWS DMS (p. 321)</a>

Tarea	Documentación relacionada
Administración de logs de tareas  Puede ver y eliminar registros de tareas utilizando la API de AWS DMS o la AWS CLI.	<a href="#">Administración de logs de tareas de AWS DMS (p. 328)</a>

## Creación de un informe de evaluación de tarea

La característica de evaluación de tareas identifica los tipos de datos que podrían no migrar correctamente. Durante una tarea de evaluación, AWS DMS lee el esquema de la base de datos de origen y crea una lista de tipos de datos. A continuación compara esta lista con una lista predefinida de tipos de datos admitidos por AWS DMS. AWS DMS crea un informe que puede examinar para ver si su tarea de migración tiene tipos de datos no compatibles.

El informe de evaluación de tareas incluye un resumen que enumera los tipos de datos no admitidos y el recuento de columna para cada uno de ellos. Incluye una lista de estructuras de datos en JSON para cada tipo de datos no compatible. Puede utilizar el informe para modificar los tipos de datos de origen y mejorar el éxito de la migración.

Hay dos niveles de tipos de datos no compatibles. Los tipos de datos que se muestran en el informe como "not supported" (no compatibles) no se pueden migrar. Los tipos de datos que se muestran en el informe como "partially supported" (se admiten parcialmente) pueden convertirse a otro tipo de datos y no migrar como espera.

Por ejemplo, el siguiente es un ejemplo de tarea de evaluación de informe.

```
{
  "summary": {
    "task-name": "test15",
    "not-supported": {
      "data-type": [
        "sql-variant"
      ],
      "column-count": 3
    },
    "partially-supported": {
      "data-type": [
        "float8",
        "jsonb"
      ],
      "column-count": 2
    }
  },
  "types": [
    {
      "data-type": "float8",
      "support-level": "partially-supported",
      "schemas": [
        {
          "schema-name": "schema1",
          "tables": [
            {
              "table-name": "table1",
              "columns": [
                "column1",
                "column2"
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

```
{  
    "table-name": "table2",  
    "columns": [  
        "column3",  
        "column4"  
    ]  
},  
}  
,  
{"  
    "schema-name": "schema2",  
    "tables": [  
        {  
            "table-name": "table3",  
            "columns": [  
                "column5",  
                "column6"  
            ]  
        },  
        {  
            "table-name": "table4",  
            "columns": [  
                "column7",  
                "column8"  
            ]  
        }  
    ]  
},  
}  
,  
{"  
    "datatype": "int8",  
    "support-level": "partially-supported",  
    "schemas": [  
        {  
            "schema-name": "schema1",  
            "tables": [  
                {  
                    "table-name": "table1",  
                    "columns": [  
                        "column9",  
                        "column10"  
                    ]  
                },  
                {  
                    "table-name": "table2",  
                    "columns": [  
                        "column11",  
                        "column12"  
                    ]  
                }  
            ]  
        }  
    ]  
},  
}
```

Puede ver el informe de evaluación de tareas más reciente en la pestaña Assessment (Evaluación) en la página Tasks (Tareas) en la consola de AWS. AWS DMS almacena informes de evaluación de tareas anteriores en un bucket de Amazon S3. El nombre del bucket de Amazon S3 está en el siguiente formato.

dms-<customerId>-<customerDNS>

El informe se guarda en el bucket en una carpeta que tiene el nombre de la tarea. El nombre del archivo del informe es la fecha de la evaluación en el formato aaaa-mm-dd-mm. Puede ver y comparar informes de evaluación de tareas anteriores de la consola de Amazon S3.

AWS DMS también crea una función de AWS Identity and Access Management (IAM) para permitir el acceso al bucket de S3; el nombre del rol es dms-access-for-tasks. La función utiliza la política `AmazonDMSRedshiftS3Role`.

Puede habilitar la característica de evaluación de tareas a través de la consola de AWS, la AWS CLI o la API de DMS:

- En la consola, elija Task Assessment al crear o modificar una tarea. Para ver el informe de evaluación de datos mediante la consola, elija la tarea en la página Tasks y seleccione la pestaña Assessment results en la sección de detalles.
- Los comandos de la CLI son `start-replication-task-assessment` para empezar una tarea de evaluación y `describe-replication-task-assessment-results` para recibir el informe de evaluación de tareas en formato JSON.
- La API de AWS DMS utiliza la acción `StartReplicationTaskAssessment` para empezar una tarea de evaluación y la acción `DescribeReplicationTaskAssessment` para recibir el informe de evaluación de tareas en formato JSON.

## Creación de una tarea

Tiene que hacer varias cosas para crear una tarea de migración de AWS DMS:

- Cree un punto de enlace de origen, un punto de enlace de destino y una instancia de replicación antes de crear una tarea de migración.
- Seleccione un método de migración:
  - Migrar datos a la base de datos de destino: este proceso crea archivos o tablas en la base de datos de destino y define automáticamente los metadatos que se necesitan en el destino. También rellena las tablas con datos del origen. Los datos de las tablas se cargan en paralelo para mejorar la eficacia. Este proceso es la opción `Migrate existing data` en la consola de AWS y se denomina `Full Load` en la API.
  - Capturar cambios durante la migración: Este proceso captura los cambios en la base de datos de origen que se producen mientras los datos migran desde el origen al destino. Cuando la migración de los datos solicitados originalmente se ha completado, el proceso de captura de datos de cambios (CDC) se aplica a los cambios capturados en la base de datos de destino. Los cambios se capturan y se aplican como unidades de transacciones confirmadas únicas y puede actualizar diversas tablas de destino como una sola confirmación de origen. Este enfoque garantiza la integridad de las transacciones en la base de datos de destino. Este proceso es la opción `Migrate existing data and replicate ongoing changes` en la consola de AWS y se denomina `full-load-and-cdc` en la API.
  - Replicar solo los cambios de los datos en la base de datos de origen: Este proceso lee el archivo del registro de recuperación del sistema de gestión de la base de datos de origen (sistema de gestión) y agrupa las entradas para cada transacción. En algunos casos, no AWS DMS puede aplicar los cambios en el destino en un plazo razonable (por ejemplo, si el destino no está accesible). En estos casos, AWS DMS almacena en el búfer los cambios en el servidor de replicación durante el tiempo que sea necesario. No vuelve a leer los registro DBMS del origen, lo que puede llevar mucho tiempo. Este proceso es la opción `Replicate data changes only` (`Replicar solo cambios de datos`) en la consola de AWS DMS.
- Determine cómo debería gestionar la tarea objetos binarios grandes (LOB) en el origen. Para obtener más información, consulte [Establecimiento de compatibilidad con LOB para bases de datos de origen en la tarea de AWS DMS \(p. 276\)](#).
- Especifique la configuración de tareas de migración. Se incluyen la configuración del registro, especificar qué datos se escriben en la tabla de control de migración, cómo se gestionan los errores y otras

configuraciones. Para obtener más información acerca de la configuración de tareas, consulte [Especificación de la configuración de tareas para tareas de AWS Database Migration Service \(p. 260\)](#).

- Configure el mapeo de tablas para definir reglas para seleccionar y filtrar los datos que está migrando. Para obtener más información sobre la asignación de tablas, consulte [Uso de la correspondencia de tablas para especificar la configuración de tareas \(p. 283\)](#). Antes de especificar las asignaciones, asegúrese de revisar la sección de la documentación sobre las asignaciones de tipos de datos para las bases de datos de origen y de destino.

Puede optar por iniciar una tarea tan pronto como termine de especificar la información para esa tarea en la página Create task (Crear tarea). De forma alternativa, puede iniciar la tarea de la página Panel después de que termine de especificar la información de la tarea.

En el procedimiento siguiente, se presupone que ha elegido el asistente de la consola de AWS DMS y ha especificado la información de la instancia de replicación y los puntos de enlace con el asistente de la consola. También puede efectuar este paso si selecciona Tasks (Tareas) en el panel de navegación de la consola de AWS DMS y, a continuación, selecciona Create Task (Crear tarea).

Para crear una tarea de migración

- En la página Create Task especifique las opciones de la tarea. La tabla siguiente describe la configuración.

## Create Task

A task can contain one or more table mappings which define what data is moved from the source to the target. If a table does not exist on the target, it can be created automatically.

Task name	ProdEndpoint-TestEndpoint	
Task description	e.g. migrate customer tables to RD	
Source endpoint	rdsoracle-endpoint	
Target endpoint	rdspostgres-endpoint	
Replication instance	replinstance1-26	
Migration type	Migrate existing data	
Start task on create	<input checked="" type="checkbox"/>	
<a href="#">Task Settings</a>		
<a href="#">Table mappings</a>		

Para esta opción	Haga lo siguiente
Task name	Escriba un nombre para la tarea.

Para esta opción	Haga lo siguiente
Task description	Escriba una descripción para la tarea.
Punto de enlace de origen	Muestra el punto de enlace de origen que se va a utilizar.
Punto de enlace de destino	Muestra el punto de enlace de destino que se va a utilizar.
Instancia de replicación	Muestra la instancia de replicación que se va a utilizar.
Migration type	Elija el método de migración que desea utilizar. Puede elegir que solo se migren los datos existentes a la base de datos de destino o que se envíen los cambios en curso a la base de datos de destino, además de los datos migrados.
Start task on create	Cuando esta opción esté seleccionada, la tarea comenzará tan pronto como se cree.

2. Seleccione la pestaña Task Settings que se muestra a continuación y especifique los valores para su tabla de destino, la compatibilidad con LOB y habilite el registro. La configuración de la tarea que se muestra dependerá del valor Migration type que seleccione. Por ejemplo, al seleccionar Migrate existing data, se muestran las siguientes opciones:

**Task Settings**

**Target table preparation mode**

- Do nothing
- Drop tables on target
- Truncate

**Include LOB columns in replication**

- Don't include LOB columns
- Full LOB mode
- Limited LOB mode

**Max LOB size (kb)\***

**Enable logging**

Para esta opción	Haga lo siguiente
Target table preparation mode	Do nothing (No hacer nada): en el modo Do nothing (No hacer nada), AWS DMS presupone que las tablas de destino se han creado previamente en el destino. Si la migración es una carga completa o una carga completa más CDC, debe asegurarse de que las tablas de destino estén vacías antes de iniciar la migración. El modo Do nothing es una opción adecuada para las tareas que solo incluyen CDC cuando las tablas de destino se han llenado previamente desde el origen y se aplica la replicación continua para mantener el origen y el destino sincronizados. Puede usar <a href="#">AWS Schema Conversion Tool</a> (AWS SCT) para que se creen previamente las tablas de destino de manera automática.

Para esta opción	Haga lo siguiente
	<p>Drop tables on target (Borrar tablas en el destino): en el modo Drop tables on target (Borrar tablas en el destino), AWS DMS borra las tablas de destino y las vuelve a crear antes de iniciar la migración. De este modo, se garantiza que las tablas de destino estén vacías cuando se inicie la migración. AWS DMS solo crea los objetos necesarios para migrar de forma eficaz los datos: tablas, claves primarias y, en algunos casos, índices únicos. AWS DMS no crea índices secundarios, restricciones de claves no principales ni valores predeterminados de datos de columna. Si realiza una tarea de carga completa más CDC o solo de CDC, le recomendamos que detenga la migración y cree índices secundarios que admitan el filtrado de instrucciones de actualización y eliminación.</p> <p>Es posible que tenga que realizar algunos ajustes en la base de datos de destino si utiliza el modo Drop tables on target (Borrar tablas en el destino). Por ejemplo, en el caso de un destino de Oracle, AWS DMS no puede crear un esquema (usuario de la base de datos) por motivos de seguridad. En tal caso, tiene que crear previamente el usuario del esquema para que AWS DMS pueda crear las tablas cuando comience la migración. Para la mayoría de los demás tipos de destino, AWS DMS crea el esquema y todas las tablas asociadas con los parámetros de configuración adecuados.</p> <p>Truncate (Truncar): en el modo Truncate (Truncar), AWS DMS trunca todas las tablas de destino antes de que comience la migración. El modo Truncate es adecuado para las migraciones de carga completa o de carga completa más CDC en las que el esquema de destino se ha creado previamente antes de que se inicie la migración. Puede usar <a href="#">AWS Schema Conversion Tool</a> (AWS SCT) para que se creen previamente las tablas de destino de manera automática.</p>
Include LOB columns in replication	<p>Don't include LOB columns (No incluir columnas LOB): las columnas LOB se excluyen de la migración.</p> <p>Full LOB mode (Modo LOB completo): se migran los LOB completos independientemente de su tamaño. AWS DMS migra los LOB por partes en porciones que se controlan con el parámetro Max LOB size (Tamaño máximo de LOB). Este modo es más lento que utilizar el modo de LOB limitado.</p> <p>Limited LOB mode (Modo de LOB limitado): los LOB se truncan de acuerdo con el valor del parámetro Max LOB size (Tamaño máximo de LOB). Este modo es más rápido que utilizar el modo de LOB completo.</p>

Para esta opción	Haga lo siguiente
Max LOB size (kb)	En Limited LOB Mode (Modo LOB limitado), las columnas de LOB que superen el valor configurado en Max LOB Size (Longitud máxima del LOB) se truncarán según el tamaño máximo especificado para los LOB.
Enable validation	Habilita la validación de datos, para verificar que los datos se migran con precisión del origen al destino. Para obtener más información, consulte <a href="#">Validación de tareas de AWS DMS (p. 334)</a> .
Enable logging	Habilita los registros de Amazon CloudWatch.

Al seleccionar Migrate existing data and replicate para Migration type, se muestran las siguientes opciones:

▼ Task Settings

**Target table preparation mode**

- Do nothing
- Drop tables on target
- Truncate

**Stop task after full load completes**

- Don't stop
- Stop Before Applying Cached Changes
- Stop After Applying Cached Changes

**Include LOB columns in replication**

- Don't include LOB columns
- Full LOB mode
- Limited LOB mode

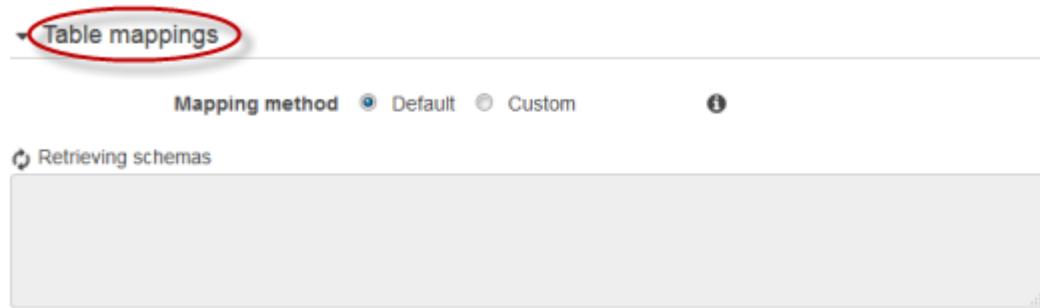
**Max LOB size (kb)\***

**Enable logging**

Para esta opción	Haga lo siguiente
Target table preparation mode	<p>Do nothing (No hacer nada): los datos y los metadatos de las tablas de destino no cambiarán.</p> <p>Drop tables on target (Borrar tablas en destino): las tablas existentes se eliminan y se crean otras para sustituirlas.</p> <p>Truncate (Truncar): las tablas se truncan sin que los metadatos de la tabla se vean afectados.</p>

Para esta opción	Haga lo siguiente
Stop task after full load completes	<p>Don't stop (No parar): no interrumpe la tarea, pero se aplican inmediatamente los cambios en la memoria caché y se continúa.</p> <p>Stop before applying cached changes (Detener antes de aplicar los cambios almacenados en caché): se detiene la tarea antes de aplicar los cambios de la memoria caché. Con este enfoque, puede agregar índices secundarios que podrían agilizar la aplicación de cambios.</p> <p>Stop after applying cached changes: se detiene la tarea después de aplicar los cambios de la memoria caché. Con este enfoque, puede agregar claves externas, disparadores, etc., si está utilizando la aplicación transaccional.</p>
Include LOB columns in replication	<p>Don't include LOB columns (No incluir columnas LOB): las columnas LOB se excluyen de la migración.</p> <p>Full LOB mode (Modo de LOB completo): se migran los LOB completos independientemente de su tamaño. Los LOB se migran por partes, en porciones que se controlan según el tamaño de la porción del LOB. Este método es más lento que utilizar el modo LOB limitado.</p> <p>Limited LOB mode (Modo de LOB limitado): se truncan los LOB al tamaño máximo permitido. Este método es más rápido que utilizar el modo LOB completo.</p>
Max LOB size (KB) (Longitud máxima del LOB)	En Limited LOB Mode (Modo LOB limitado), las columnas de LOB que superen el valor configurado en Max LOB Size (Longitud máxima del LOB) se truncarán según el tamaño máximo especificado para los LOB.
Enable validation	Habilita la validación de datos, para verificar que los datos se migran con precisión del origen al destino. Para obtener más información, consulte <a href="#">Validación de tareas de AWS DMS (p. 334)</a> .
Enable logging	Habilita los registros de Amazon CloudWatch.

3. Seleccione la pestaña Table mappings que se muestra a continuación para establecer valores para el mapeo de esquemas y el método de mapeo. Si elige Custom, puede especificar el esquema de destino y los valores de la tabla. Para obtener más información sobre la asignación de tablas, consulte [Uso de la correspondencia de tablas para especificar la configuración de tareas \(p. 283\)](#).



4. Una vez que haya completado la configuración de las tareas, seleccione Create task (Crear tarea).

## Especificación de la configuración de tareas para tareas de AWS Database Migration Service

Cada tarea tiene valores que pueden configurar en función de las necesidades de la migración de la base de datos. Puede crear estos ajustes en un archivo JSON o, con algunos ajustes, puede especificar la configuración con la consola de AWS DMS.

Existen diversos tipos principales de configuración de tareas, como se indica a continuación:

### Temas

- [Configuración de las tareas de los metadatos de destino \(p. 263\)](#)
- [Configuración de tareas de carga completa \(p. 264\)](#)
- [Configuración de las de tareas de los registros \(p. 265\)](#)
- [Configuración de las tareas de la tabla de control \(p. 266\)](#)
- [Configuración de las tareas del búfer de transmisión \(p. 269\)](#)
- [Configuración de los ajustes del procesamiento de cambios \(p. 269\)](#)
- [Configuración de tareas de validación de datos \(p. 270\)](#)
- [Configuración de tareas para la administración de DDL del procesamiento de cambios \(p. 272\)](#)
- [Configuración de las tareas de administración de errores \(p. 272\)](#)
- [Guardar la configuración de las tareas \(p. 274\)](#)

Configuración de tarea	Documentación relacionada
Creación de un informe de evaluación de tarea  Puede crear un informe de evaluación de tarea que muestra cualquier tipo de dato no admitido que podría provocar problemas durante la migración. Puede ejecutar este informe en la tarea antes de ejecutar la tarea para encontrar posibles problemas.	<a href="#">Creación de un informe de evaluación de tarea (p. 252)</a>
Creación de una tarea	<a href="#">Creación de una tarea (p. 254)</a>

Configuración de tarea	Documentación relacionada
Al crear una tarea, especifique el origen, el destino y la instancia de replicación, junto con cualquier configuración de migración.	
Creación de una tarea de replicación continua  Puede configurar una tarea para proporcionar replicación continua entre el origen y el destino.	<a href="#">Creación de tareas para la replicación continua con AWS DMS (p. 277)</a>
Aplicación de la configuración de las tareas  Cada tarea tiene valores que pueden configurar en función de las necesidades de la migración de la base de datos. Puede crear estos ajustes en un archivo JSON o, con algunos ajustes, puede especificar la configuración con la consola de AWS DMS.	<a href="#">Especificación de la configuración de tareas para tareas de AWS Database Migration Service (p. 260)</a>
Validación de datos  La validación de datos es una configuración de la tarea que puede utilizar para que AWS DMS compare los datos en el almacén de datos de destino con los datos de su almacén de datos de origen.	<a href="#">Validación de tareas de AWS DMS (p. 334)</a>
Modifying a Task (Modificación de una tarea)  Cuando se detiene una tarea, puede modificar sus valores de configuración.	<a href="#">Modificar una tarea (p. 280)</a>
Recarga de tablas durante una tarea  Puede volver a cargar una tabla mientras se realiza una tarea si se produce un error durante esta tarea.	<a href="#">Volver a cargar tablas durante una tarea (p. 280)</a>
Uso del asignación de tablas  La correspondencia de tablas utiliza diversos tipos de reglas para especificar la configuración de tareas del origen de datos, el esquema origen, los datos y las transformaciones que deben producirse durante la tarea.	Reglas de selección <a href="#">Reglas de selección y acciones (p. 288)</a>  Reglas de transformación <a href="#">Reglas y acciones de transformación (p. 291)</a>

Configuración de tarea	Documentación relacionada
<b>Aplicación de filtros</b>  Puede utilizar filtros de origen para limitar el número y el tipo de los registros transferidos desde el origen al destino. Por ejemplo, puede especificar que solo los trabajadores con una ubicación de sede central se trasladen a la base de datos de destino. Puede aplicar filtros en una columna de datos.	<a href="#">Usar filtros de origen (p. 317)</a>
<b>Monitorización de una tarea</b>  Hay varias formas de obtener información sobre el desempeño de una tarea y las tablas que utiliza esta tarea.	<a href="#">Monitorización de tareas de AWS DMS (p. 321)</a>
<b>Administración de logs de tareas</b>  Puede ver y eliminar registros de tareas utilizando la API de AWS DMS o la AWS CLI.	<a href="#">Administración de logs de tareas de AWS DMS (p. 328)</a>

Un archivo JSON de configuración de tareas puede tener el siguiente aspecto.

```
{
  "TargetMetadata": {
    "TargetSchema": "",
    "SupportLobs": true,
    "FullLobMode": false,
    "LobChunkSize": 64,
    "LimitedSizeLobMode": true,
    "LobMaxSize": 32,
    "BatchApplyEnabled": true
  },
  "FullLoadSettings": {
    "TargetTablePrepMode": "DO_NOTHING",
    "CreatePkAfterFullLoad": false,
    "StopTaskCachedChangesApplied": false,
    "StopTaskCachedChangesNotApplied": false,
    "MaxFullLoadSubTasks": 8,
    "TransactionConsistencyTimeout": 600,
    "CommitRate": 10000
  },
  "Logging": {
    "EnableLogging": false
  },
  "ControlTablesSettings": {
    "ControlSchema": "",
    "HistoryTimeslotInMinutes": 5,
    "HistoryTableEnabled": false,
    "SuspendedTablesTableEnabled": false,
    "StatusTableEnabled": false
  },
  "StreamBufferSettings": {
    "StreamBufferCount": 3,
    "StreamBufferSizeInMB": 8
  }
}
```

```

    "ChangeProcessingTuning": {
        "BatchApplyPreserveTransaction": true,
        "BatchApplyTimeoutMin": 1,
        "BatchApplyTimeoutMax": 30,
        "BatchApplyMemoryLimit": 500,
        "BatchSplitSize": 0,
        "MinTransactionSize": 1000,
        "CommitTimeout": 1,
        "MemoryLimitTotal": 1024,
        "MemoryKeepTime": 60,
        "StatementCacheSize": 50
    },
    "ChangeProcessingDdlHandlingPolicy": {
        "HandleSourceTableDropped": true,
        "HandleSourceTableTruncated": true,
        "HandleSourceTableAltered": true
    },
    "ValidationSettings": {
        "EnableValidation": true,
        "ThreadCount": 5
    },
    "ErrorBehavior": {
        "DataErrorPolicy": "LOG_ERROR",
        "DataTruncationErrorPolicy": "LOG_ERROR",
        "DataErrorEscalationPolicy": "SUSPEND_TABLE",
        "DataErrorEscalationCount": 50,
        "TableErrorPolicy": "SUSPEND_TABLE",
        "TableErrorEscalationPolicy": "STOP_TASK",
        "TableErrorEscalationCount": 50,
        "RecoverableErrorCount": 0,
        "RecoverableErrorInterval": 5,
        "RecoverableErrorThrottling": true,
        "RecoverableErrorThrottlingMax": 1800,
        "ApplyErrorDeletePolicy": "IGNORE_RECORD",
        "ApplyErrorInsertPolicy": "LOG_ERROR",
        "ApplyErrorUpdatePolicy": "LOG_ERROR",
        "ApplyErrorEscalationPolicy": "LOG_ERROR",
        "ApplyErrorEscalationCount": 0,
        "FullLoadIgnoreConflicts": true
    }
}

```

## Configuración de las tareas de los metadatos de destino

Estos son algunos de los valores de configuración de los metadatos:

- **TargetSchema:** El nombre del esquema de la tabla de destino. Si esta opción de metadatos está en blanco, se utiliza el esquema de la tabla de origen. añade AWS DMS automáticamente el prefijo del propietario a la base de datos de destino para todas las tablas si no se definió un esquema de origen. Esta opción debe estar en blanco para los puntos de enlace de destino de tipo MySQL.
- **Configuración de LOB:** la configuración determina cómo se administran los objetos grandes (LOB). Si establece `SupportLobs=true`, debe establecer una de las siguientes opciones en `true`:
  - **FullLobMode:** si establece esta opción en `true`, debe indicar un valor para la opción `LobChunkSize`. Indique el tamaño, en kilobytes, de los fragmentos del LOB que se van a utilizar cuando se repliquen los datos en el destino. La opción `FullLobMode` es más adecuada para los LOB de gran tamaño, pero suele ralentizar la carga.
  - **InlineLobMaxSize:** este valor determina qué LOBS AWS Database Migration Service transfiere en línea durante una carga completa. La transferencia de LOB pequeños es más eficaz que buscarlos de una tabla de origen. Durante una carga completa, AWS Database Migration Service comprueba todos los LOB y realiza una transferencia en línea para los LOB de tamaño inferior a `InlineLobMaxSize`.

AWS Database Migration Service transfiere todos los LOB de tamaño superior a `InlineLobMaxSize` en `FullLobMode`. El valor predeterminado `InlineLobMaxSize` es 0 y el rango es 1 kilobyte–2 gigabyte. Establezca un valor para `InlineLobMaxSize` solo si sabe que la mayor parte de los LOB tienen un tamaño inferior al valor especificado en `InlineLobMaxSize`.

- `LimitedSizeLobMode`: Si establece esta opción en `true`, debe indicar un valor para la opción `LobMaxSize`. Indique el tamaño máximo, en kilobytes, para un LOB específico.

Para obtener más información acerca de los criterios para utilizar esta configuración de LOB de tarea, consulte [Establecimiento de compatibilidad con LOB para bases de datos de origen en la tarea de AWS DMS \(p. 276\)](#). También puede controlar la administración de LOB para tablas individuales. Para obtener más información, consulte [Reglas y operaciones de table-settings \(p. 300\)](#).

- `LoadMaxFileSize`: una opción para los puntos de enlace de destino de PostgreSQL y MySQL que define el tamaño máximo en el disco de los datos almacenados que no se descargaron, como, por ejemplo, archivos CSV. Esta opción anula el atributo de conexión. Puede especificar valores a partir de 0, que indica que esta opción no anula el atributo de conexión, hasta 100 000 KB.
- `BatchApplyEnabled`: Determina si cada transacción se aplica individualmente o si los cambios se confirman en lotes. El valor predeterminado es `false`.

El parámetro `BatchApplyEnabled` se usa con el parámetro `BatchApplyPreserveTransaction`. Si `BatchApplyEnabled` está establecido en `true`, el parámetro `BatchApplyPreserveTransaction` determina la integridad de las transacciones.

Si `BatchApplyPreserveTransaction` está establecido en `true`, se mantiene la integridad de las transacciones y se garantiza que un lote contendrá todos los cambios en una transacción desde el origen.

Si `BatchApplyPreserveTransaction` está establecido en `false`, pueden producirse interrupciones temporales en la integridad de las transacciones para mejorar el desempeño.

El parámetro `BatchApplyPreserveTransaction` se aplica únicamente a los puntos de enlace de destino de Oracle y solo es pertinente cuando el parámetro `BatchApplyEnabled` está establecido en `true`.

Cuando se incluyen columnas de LOB en la replicación, `BatchApplyEnabled` solo se puede utilizar en modo de LOB limitado.

Para obtener más información acerca del uso de esta configuración para una carga de captura de datos de cambios (CDC), consulte [Configuración de los ajustes del procesamiento de cambios \(p. 269\)](#).

- `ParallelLoadThreads`: especifica el número de subprocessos que AWS DMS utiliza para cargar cada tabla en la base de datos de destino. El valor máximo para un destino MySQL es 16. El valor máximo para un destino DynamoDB es 32. Puede solicitar un aumento del límite máximo. Para obtener información acerca de la configuración de la carga paralela de las tablas individuales, consulte [Reglas y operaciones de table-settings \(p. 300\)](#).
- `ParallelLoadBufferSize`: especifica el número máximo de registros para almacenar en el búfer que los subprocessos de carga en paralelo utilizan para cargar datos en el destino. El valor predeterminado es 50. El valor máximo es 1000. Este campo actualmente solo es válido cuando DynamoDB es el destino. Utilice este parámetro con `ParallelLoadThreads`; `ParallelLoadBufferSize` solo es válido cuando hay más de un subprocesso. Para obtener información acerca de la configuración de la carga paralela de las tablas individuales, consulte [Reglas y operaciones de table-settings \(p. 300\)](#).

## Configuración de tareas de carga completa

La configuración de carga completa incluye lo siguiente:

- Para indicar cómo se gestiona la carga del destino con arranque en carga completa, especifique uno de los siguientes valores para la opción `TargetTablePrepMode`:

- **DO NOTHING:** Los datos y los metadatos de la tabla de destino existente no se verán afectados.
- **DROP\_AND\_CREATE:** La tabla existente se ha descartado y se creó una tabla nueva para sustituirla.
- **TRUNCATE\_BEFORE\_LOAD:** Los datos se truncan sin que los metadatos de la tabla se vean afectados.
- Para retrasar la creación de la clave principal o el índice único hasta que finalice la carga completa, establezca la opción `CreatePkAfterFullLoad`.  
Cuando se selecciona esta opción, no podrá reanudar las tareas de carga completa que no se hayan terminado.
- Para las tareas que acepten carga completa y CDC, puede configurar las siguientes opciones de `Stop task after full load completes`:
  - `StopTaskCachedChangesApplied`: Establezca esta opción en `true` para detener una tarea después de que finalice un proceso de carga completa y se apliquen los cambios en la memoria caché.
  - `StopTaskCachedChangesNotApplied`: Establezca esta opción en `true` para detener una tarea antes de que se apliquen cambios en la memoria caché.
- `MaxFullLoadSubTasks`: Establezca esta opción para indicar el número máximo de tablas que se pueden cargar en paralelo. El valor predeterminado es 8, el valor máximo es 49.
- Para establecer el número de segundos que debe esperar AWS DMS para que se cierren las transacciones antes de iniciar una operación de carga completa, si las transacciones están abiertas cuando se inicia la tarea, establezca la opción `TransactionConsistencyTimeout`. El valor predeterminado es 600 (10 minutos). AWS DMS inicia la carga completa después de haber alcanzado el valor de tiempo de espera, incluso aunque haya transacciones abiertas. Una tarea de carga completa solamente no espera durante 10 minutos sino que comienza inmediatamente.
- Para indicar el número máximo de eventos que se pueden transferir a la vez, establezca la opción `CommitRate`.

## Configuración de las de tareas de los registros

La configuración de tareas de registro se escriben en un archivo JSON y le permiten especificar qué actividades de los componentes se van a registrar y qué cantidad de información se escribirá en el registro. La función de registro utiliza Amazon CloudWatch para registrar información durante el proceso de migración.

Hay varias formas de habilitar los registros de Amazon CloudWatch. Puede seleccionar la opción `EnableLogging` en la Consola de administración de AWS cuando cree una tarea de migración o establezca la opción `EnableLogging` en `true` cuando cree una tarea con la API de AWS DMS. También puede especificar "EnableLogging": `true` en el JSON de la sección de registro en la configuración de tareas.

Para eliminar los logs de tareas, puede especificar "`DeleteTaskLogs`": `true` en el JSON de la sección de registro de la configuración de tareas.

Puede establecer el registro para las siguientes actividades de componentes:

- **SOURCE\_UNLOAD:** los datos se descargaron desde la base de datos de origen.
- **SOURCE\_CAPTURE:** los datos se capturan desde la base de datos de origen.
- **TARGET\_LOAD:** los datos se cargan en la base de datos de destino.
- **TARGET\_APPLY:** los datos e instrucciones de lenguaje de definición de datos (DDL) se aplican a la base de datos de destino.
- **TASK\_MANAGER:** el administrador de tareas desencadena un evento.

Después de especificar la actividad de un componente, puede especificar la cantidad de información que se registra. La siguiente lista se ordena en orden ascendente desde el nivel inferior de información hasta

el más elevado. Los niveles más elevados siempre incluyen información de los niveles inferiores. Estos valores de gravedad incluyen:

- **LOGGER\_SEVERITY\_ERROR**: los mensajes de error se escriben en el registro.
- **LOGGER\_SEVERITY\_WARNING**: las advertencias y mensajes de error se escriben en el registro.
- **LOGGER\_SEVERITY\_INFO**: los mensajes informativos, las advertencias y los mensajes de error se escriben en el registro.
- **LOGGER\_SEVERITY\_DEFAULT**: los mensajes informativos, las advertencias y los mensajes de error se escriben en el registro.
- **LOGGER\_SEVERITY\_DEBUG**: Los mensajes de depuración, los mensajes informativos, las advertencias y los mensajes de error se escriben en el registro.
- **LOGGER\_SEVERITY\_DETAILED\_DEBUG**: toda la información se escribe en el registro.

Por ejemplo, la siguiente sección JSON presenta la configuración de las tareas para el registro de todas las actividades de componentes.

```
...
  "Logging": {
    "EnableLogging": true,
    "LogComponents": [
      {
        "Id": "SOURCE_UNLOAD",
        "Severity": "LOGGER_SEVERITY_DEFAULT"
      },
      {
        "Id": "SOURCE_CAPTURE",
        "Severity": "LOGGER_SEVERITY_DEFAULT"
      },
      {
        "Id": "TARGET_LOAD",
        "Severity": "LOGGER_SEVERITY_DEFAULT"
      },
      {
        "Id": "TARGET_APPLY",
        "Severity": "LOGGER_SEVERITY_INFO"
      },
      {
        "Id": "TASK_MANAGER",
        "Severity": "LOGGER_SEVERITY_DEBUG"
      }
    ]
  },
...
}
```

## Configuración de las tareas de la tabla de control

Las tablas de control ofrecen información sobre la tarea de AWS DMS, así como útiles estadísticas que puede utilizar para planificar y administrar las tareas de migración y las tareas futuras. Puede aplicar estos valores de configuración de tareas en un archivo JSON o mediante el vínculo Advanced Settings (Configuración avanzada) en la página Create task (Crear tarea) de la consola de AWS DMS. Además de la tabla Apply Exceptions (Aplicar excepciones) (`dmslogs.awsdms_apply_exceptions`), que siempre se crea, puede crear otras tablas, entre las que se incluyen las siguientes:

- **Replication Status** (`dmslogs.awsdms_status`) (Estado de replicación): esta tabla ofrece información sobre el estado de la tarea actual. Esto incluye el estado de la tarea, la cantidad de memoria que consume la tarea, y el número de cambios que no se han aplicado aún en el destino. Esta tabla también ofrece a la posición en la base de datos de origen donde AWS DMS se está leyendo actualmente e indica si la tarea es una carga completa o una captura de datos de cambios (CDC).
- **Tablas suspendidas** (`dmslogs.awsdms_suspended_tables`): Esta tabla proporciona una lista de tablas suspendidas, así como el motivo por el que están suspendidas.
- **Replication History** (`dmslogs.awsdms_history`) (Historial de replicación): esta tabla proporciona información sobre el historial de replicación. Esta información incluye el número y el volumen de registros procesados durante la tarea, la latencia al final de una tarea de CDC y otras estadísticas.

La tabla Apply Exceptions (Aplicar excepciones) (`dmslogs.awsdms_apply_exceptions`) contiene los siguientes parámetros:

Columna	Tipo	Descripción
TASK_NAME	nvchar	El nombre de la tarea AWS DMS.
TABLE_OWNER	nvchar	El propietario de la tabla.
TABLE_NAME	nvchar	El nombre de la tabla.
ERROR_TIME	timestamp	La hora en que se produjo la excepción (error).
STATEMENT	nvchar	La instrucción que se estaba ejecutando cuando se produjo el error.
ERROR	nvchar	El nombre y la descripción del error.

La tabla Replication History (Historial de replicación) (`dmslogs.awsdms_history`) contiene los siguientes parámetros:

Columna	Tipo	Descripción
SERVER_NAME	nvchar	El nombre de la máquina donde se está ejecutando la tarea de replicación.
TASK_NAME	nvchar	El nombre de la tarea AWS DMS.
TIMESLOT_TYPE	varchar	Uno de los valores siguientes: <ul style="list-style-type: none"><li>• FULL LOAD</li><li>• CHANGE PROCESSING (CDC)</li></ul> Si la tarea se ejecuta en carga completa y CDC, se escriben dos registros con el historial en el slot de tiempo.
TIMESLOT	timestamp	La marca de tiempo de finalización del slot de tiempo.
TIMESLOT_DURATION	int	La duración del slot de tiempo.
TIMESLOT_LATENCY	int	La latencia de destino al final del slot de tiempo. Este valor solo se aplica a los slots de tiempo de CDC.
RECORDS	int	El número de registros procesados durante el slot de tiempo.
TIMESLOT_VOLUME	int	El volumen de datos procesados en MB.

La tabla Replication Status (Estado de replicación) (`dmslogs.awsdms_status`) contiene el estado actual de la tarea y la base de datos de destino. Tiene los siguientes valores de configuración:

Columna	Tipo	Descripción
SERVER_NAME	nvchar	El nombre de la máquina donde se está ejecutando la tarea de replicación.
TASK_NAME	nvchar	El nombre de la tarea AWS DMS.
TASK_STATUS	varchar	Uno de los valores siguientes: <ul style="list-style-type: none"><li>• FULL LOAD</li><li>• CHANGE PROCESSING (CDC)</li></ul> El estado de la tarea se establece en FULL LOAD siempre que haya al menos una tabla en carga completa. Después de que todas las tablas se hayan cargado, el estado de la tarea cambia a CHANGE PROCESSING si está habilitada la CDC.
STATUS_TIME	timestamp	La marca de tiempo del estado de la tarea.
PENDING_CHANGES	int	El número de registros de cambio que no se aplicaron al destino.
DISK_SWAP_SIZE	int	La cantidad de espacio en disco que utilizan las transacciones antiguas o descargadas.
TASK_MEMORY	int	Memoria actual utilizada, en MB.
SOURCE_CURRENT_POSITION	varchar	La posición en la base de datos de origen desde donde lee AWS DMS en estos momentos.
SOURCE_CURRENT_TIMESTAMP	timestamp	La marca de tiempo en la base de datos de origen desde donde lee AWS DMS en estos momentos.
SOURCE_TAIL_POSITION	varchar	La posición de la transacción de inicio más antigua que no se confirmó. Este valor es la posición más reciente a la que puede volver sin perder ningún cambio.
SOURCE_TAIL_TIMESTAMP	timestamp	La marca de tiempo de la transacción de inicio más antigua que no se confirmó. Este valor es la marca de tiempo más reciente a la que puede volver sin perder ningún cambio.
SOURCE_TIMESTAMP_APPLIED	timestamp	La marca de tiempo de la última confirmación de transacción. En un

Columna	Tipo	Descripción
		proceso de aplicación masiva, este valor es la marca de tiempo para la confirmación de la última transacción del lote.

La configuración adicional de la tabla de control incluye lo siguiente:

- **ControlSchema**: utilice esta opción para indicar el nombre del esquema de la base de datos para las tablas de control de destino de AWS DMS. Si no indica información en este campo, las tablas se copian a la ubicación predeterminada en la base de datos.
- **HistoryTimeslotInMinutes**: Utilice esta opción para indicar la longitud de cada slot de tiempo en la tabla Historial de replicación. El valor predeterminado es 5 minutos.

## Configuración de las tareas del búfer de transmisión

Puede configurar los valores del búfer de transmisión mediante AWS CLI, que incluye las siguientes opciones:

- **StreamBufferCount**: Utilice esta opción para especificar el número de búferes de transmisión de datos para la tarea de migración. El número predeterminado del búfer de transmisión es 3. Al aumentar el valor de esta configuración podría aumentar la velocidad de extracción de los datos. Sin embargo, este aumento del desempeño depende en gran medida del entorno de migración, incluido el sistema de origen y la clase de instancia del servidor de replicación. El valor predeterminado es suficiente para la mayoría de las situaciones.
- **StreamBufferSizeInMB**: Utilice esta opción para indicar el tamaño máximo de cada búfer de transmisión de datos. El valor predeterminado es 8 MB de tamaño. Es posible que necesite aumentar el valor de esta opción cuando trabaje con LOB muy grandes. También sería necesario aumentar el valor si recibe un mensaje en los archivos de registro de que el tamaño del búfer de transmisión no es suficiente. Al calcular el tamaño de esta opción, puede utilizar la siguiente ecuación: `[Max LOB size (or LOB chunk size)]*[number of LOB columns]*[number of stream buffers]*[number of tables loading in parallel per task(MaxFullLoadSubTasks)]*3`
- **CtrlStreamBufferSizeInMB**: Utilice esta opción para definir el tamaño del búfer de la transmisión de control. El valor se expresa en megabytes y puede ser de 1 a 8. El valor predeterminado es 5. Es posible que necesite incrementar esto si trabaja con un gran número de tablas, como, por ejemplo, decenas de miles de tablas.

## Configuración de los ajustes del procesamiento de cambios

Los siguientes ajustes determinan cómo AWS DMS se ocupa de los cambios para las tablas de destino durante la captura de datos de cambios (CDC). Varios de estos ajustes dependerán del valor del parámetro de los metadatos de destino `BatchApplyEnabled`. Para obtener más información sobre el parámetro `BatchApplyEnabled`, consulte [Configuración de las tareas de los metadatos de destino \(p. 263\)](#).

La configuración de los ajustes del procesamiento de cambios incluyen las siguientes opciones:

Se aplicarán exclusivamente los siguientes valores de configuración cuando el parámetro de metadatos `BatchApplyEnabled` se establezca en `true`.

- **BatchApplyPreserveTransaction**: Si está establecido en `true`, se mantiene la integridad de las transacciones y se garantiza que un lote contendrá todos los cambios en una transacción desde el origen. El valor predeterminado es `true`. Esta configuración se aplica solo a los puntos de enlace de destino de Oracle.

Si se establece en `false`, pueden producirse interrupciones temporales en la integridad de las transacciones para mejorar el desempeño. No se garantiza que todos los cambios en una transacción desde el origen se apliquen al destino en un solo lote.

- `BatchApplyTimeoutMin`: Establece la cantidad mínima de tiempo en segundos que espera AWS DMS entre cada aplicación de cambios por lotes. El valor predeterminado es 1.
- `BatchApplyTimeoutMax`: Establece la cantidad máxima de tiempo en segundos que espera AWS DMS entre cada aplicación de cambios por lotes antes de que se agote el tiempo. El valor predeterminado es 30.
- `BatchApplyMemoryLimit`: Establece la cantidad máxima de memoria en (MB) para utilizar en el procesamiento previo en modo de aplicación optimizado por lotes. El valor predeterminado es 500.
- `BatchSplitSize`: Establece el número máximo de cambios que se aplican en un solo lote. El valor predeterminado es 0, lo que significa que no se aplica ningún límite.

Se aplicarán exclusivamente los siguientes valores de configuración cuando el parámetro de metadatos `BatchApplyEnabled` se establezca en `false`.

- `MinTransactionSize`: Establece el número mínimo de cambios que se van a incluir en cada transacción. El valor predeterminado es 1000.
- `CommitTimeout`: Establece el tiempo máximo en segundos para que AWS DMS recopile las transacciones en lotes antes de declarar un tiempo de espera. El valor predeterminado es 1.
- `HandleSourceTableAltered`: Establezca esta opción en `true` para modificar la tabla de destino cuando se haya modificado la tabla de origen.

AWS DMS intenta conservar los datos de las transacciones en la memoria hasta que la transacción se haya confirmado en su totalidad en el origen o el destino. Sin embargo, las transacciones que superan el tamaño de la memoria asignada o que no se confirmaron en el límite de tiempo especificado se escriben en el disco.

La configuración siguiente se aplican para cambiar el ajuste de procesamiento independientemente del modo de procesamiento de cambios.

- `MemoryLimitTotal`: Establece el tamaño máximo (en MB) que pueden utilizar todas las transacciones en la memoria antes de que se escriban en el disco. El valor predeterminado es 1024.
- `MemoryKeepTime`: Establece el tiempo máximo en segundos que cada transacción puede permanecer en la memoria antes de que se escriba en el disco. La duración se calcula a partir del momento en que AWS DMS inicia la captura de la transacción. El valor predeterminado es 60.
- `StatementCacheSize`: Establece el número máximo de instrucciones preparadas para que se almacenen en el servidor para su posterior ejecución a la hora de aplicar los cambios en el destino. El valor predeterminado es 50. El valor máximo es 200.

## Configuración de tareas de validación de datos

Puede garantizar que los datos se han migrado de forma precisa del origen al destino. Si habilita la validación para una tarea, AWS DMS empieza a comparar los datos de origen y de destino inmediatamente después de realizar una carga completa para una tabla. Para obtener más información acerca de la validación de datos de tareas, sus requisitos, el ámbito de soporte de su base de datos y las métricas que notifica, consulte [Validación de tareas de AWS DMS \(p. 334\)](#).

La configuración de la validación de datos y sus valores incluye lo siguiente:

- `EnableValidation`: habilita la validación de datos cuando se establece en `true`. De lo contrario, la validación se deshabilita para la tarea. El valor predeterminado es `false`.

- **FailureMaxCount:** especifica el número máximo de registros que pueden generar un error de validación antes de que se suspenda la validación de la tarea. El valor predeterminado es 10,000. Si desea que la validación continúe con independencia del número de registros que generen un error de validación, defina un valor superior al número de registros que hay en el origen.
- **HandleCollationDiff:** cuando esta opción se establece en true, la validación tiene en cuenta las diferencias de intercalación de columna en los puntos de enlace de PostgreSQL cuando identifica los registros de origen y de destino que se comparan. De lo contrario, dichas diferencias en la intercalación de columna se pasan por alto para la validación. En puntos de enlace de PostgreSQL, las intercalaciones de columna pueden determinar el orden de las filas, lo cual es importante para la validación de datos. Al definir `HandleCollationDiff` en true se resuelven automáticamente estas diferencias de intercalación y se evitan falsos positivos en la validación de datos. El valor predeterminado es false.
- **RecordFailureDelayLimitInMinutes:** especifica el retraso antes de notificar los detalles de error de validación. Normalmente, AWS DMS utiliza la latencia de tareas para reconocer el retraso real en los cambios que hacer en el destino a fin de evitar falsos positivos. Esta configuración anula el valor de retraso real y le permite establecer un retraso mayor antes de notificar métricas de validación. El valor predeterminado es 0.
- **TableFailureMaxCount:** especifica el número máximo de tablas que pueden generar un error de validación antes de que se suspenda la validación de la tarea. El valor predeterminado es 1,000. Si desea que la validación continúe con independencia del número de tablas que generen un error de validación, defina un valor superior al número de tablas que hay en el origen.
- **ThreadCount:** especifica el número de subprocessos de ejecución que AWS DMS utiliza durante la validación. Cada subprocesso selecciona datos aún no validados del origen y del destino para compararlos y validarlos. El valor predeterminado es 5. Si establece `ThreadCount` en un número más elevado, AWS DMS puede completar la validación con mayor rapidez. Sin embargo, AWS DMS a continuación ejecuta más consultas simultáneas, consumiendo más recursos en el origen y el destino.
- **ValidationOnly:** cuando esta opción se establece en true, la ejecución de la tarea previsualiza la validación de datos sin realizar ninguna migración ni replicación de datos.

Para poder establecer esta opción, en la consola de AWS DMS establezca la tarea `Migration type` (Tipo de migración) en `Replicate data changes only` (Replicar solo los cambios en los datos). De forma alternativa, en la API AWS DMS establezca el tipo de migración en `cdc`.

Al adoptar este enfoque, puede ver los resultados de validación y resolver los errores antes de mover los datos realmente. Esta opción puede resultar más eficiente que esperar a resolver los errores después de que todos los datos de origen se hayan migrado al destino. El valor predeterminado es false.

Por ejemplo, el siguiente JSON permite la validación de datos con dos veces el número predeterminado de procesos. También contabiliza las diferencias en el orden de registros provocadas por diferencias de intercalación de columnas en puntos de enlace de PostgreSQL. Además, proporciona un retraso de informes de validación para incluir tiempo adicional para procesar los errores de validación.

```
"ValidationSettings": {  
    "EnableValidation": true,  
    "ThreadCount": 10,  
    "HandleCollationDiff": true,  
    "RecordFailureDelayLimitInMinutes": 30  
}
```

#### Note

Para un punto de enlace de Oracle, AWS DMS utiliza DBMS\_CRYPTO para la validación de BLOB. Si su punto de enlace de Oracle utiliza BLOB, conceda el permiso de ejecución para

DBMS\_CRYPTO a la cuenta de usuario que se utiliza para obtener acceso al punto de enlace de Oracle. Hágalo ejecutando la siguiente instrucción.

```
grant execute on sys.dbms_crypto to <dms_endpoint_user>;
```

## Configuración de tareas para la administración de DDL del procesamiento de cambios

Los siguientes ajustes determinan cómo AWS DMS se ocupa de los cambios del lenguaje de definición de datos (DDL) para las tablas de destino durante la captura de datos de cambios (CDC). La configuración de tareas para la administración de DDL de procesamiento de cambios, incluye las siguientes opciones:

- **HandleSourceTableDropped** – Establezca esta opción en `true` para dejar la tabla de destino cuando se ha dejado la tabla de origen.
- **HandleSourceTableTruncated**: Establezca esta opción en `true` para truncar la tabla de destino cuando se haya truncado la tabla de origen.
- **HandleSourceTableAltered**: Establezca esta opción en `true` para modificar la tabla de destino cuando se haya modificado la tabla de origen.

## Configuración de las tareas de administración de errores

Puede establecer el comportamiento de gestión de errores de la tarea de replicación durante la captura de datos de cambio (CDC) mediante los siguientes valores:

- **DataErrorPolicy**: Determina la acción que emprende AWS DMS cuando hay un error relacionado con el procesamiento de datos en el registro. Algunos ejemplos de errores en el procesamiento de datos incluyen los errores de conversión, los errores de transformación y los datos incorrectos. El valor predeterminado es `LOG_ERROR`.
  - **IGNORE\_RECORD**: La tarea continúa y los datos de este registro se omiten. El recuento de errores de la propiedad `DataErrorEscalationCount` se incrementa. Por lo tanto, si se establece un límite en los errores de una tabla, este error cuenta para el límite.
  - **LOG\_ERROR**: La tarea continúa y el error se escribe en el registro de tareas.
  - **SUSPEND\_TABLE**: La tarea continúa, pero los datos de la tabla con el registro de errores se pasan a un estado de error y los datos no se replican.
  - **STOP\_TASK**: La tarea se detiene y se necesitará intervención manual.
- **DataTruncationErrorPolicy**: Determina la acción que emprende AWS DMS cuando se truncan datos. El valor predeterminado es `LOG_ERROR`.
  - **IGNORE\_RECORD**: La tarea continúa y los datos de este registro se omiten. El recuento de errores de la propiedad `DataErrorEscalationCount` se incrementa. Por lo tanto, si se establece un límite en los errores de una tabla, este error cuenta para el límite.
  - **LOG\_ERROR**: La tarea continúa y el error se escribe en el registro de tareas.
  - **SUSPEND\_TABLE**: La tarea continúa, pero los datos de la tabla con el registro de errores se pasan a un estado de error y los datos no se replican.
  - **STOP\_TASK**: La tarea se detiene y se necesitará intervención manual.
- **DataErrorEscalationPolicy**: Determina la acción que emprende AWS DMS cuando se alcanza el número máximo de errores (establecido en el parámetro `DataErrorsEscalationCount`). El valor predeterminado es `SUSPEND_TABLE`.
  - **SUSPEND\_TABLE**: La tarea continúa, pero los datos de la tabla con el registro de errores se pasan a un estado de error y los datos no se replican.
  - **STOP\_TASK**: La tarea se detiene y se necesitará intervención manual.

- **DataErrorEscalationCount:** Establece el número máximo de errores que pueden producirse en los datos para un registro específico. Cuando se alcanza esta cifra, los datos de la tabla que contiene el registro de errores se administran de acuerdo con la política que se estableció en `DataErrorEscalationCount`. El valor predeterminado es 0.
- **TableErrorPolicy:** Determina la acción que emprende AWS DMS cuando se produce un error al procesar los datos o los metadatos para una tabla específica. Este error solo se aplica a los datos de la tabla general y no es un error que se refiera a un registro específico. El valor predeterminado es `SUSPEND_TABLE`.
  - `SUSPEND_TABLE`: La tarea continúa, pero los datos de la tabla con el registro de errores se pasan a un estado de error y los datos no se replican.
  - `STOP_TASK`: La tarea se detiene y se necesitará intervención manual.
- **TableErrorEscalationPolicy:** Determina la acción que emprende AWS DMS cuando se alcanza el número máximo de errores (establecida utilizando el parámetro `TableErrorEscalationCount`). El valor de configuración predeterminado y único del usuario es `STOP_TASK`, con el que la tarea se detiene y se requiere intervención manual.
- **TableErrorEscalationCount:** El número máximo de errores que pueden producirse en los datos o metadatos generales para una tabla específica. Cuando se alcanza esta cifra, los datos de la tabla se gestionan de acuerdo con la política establecida en `TableErrorEscalationPolicy`. El valor predeterminado es 0.
- **RecoverableErrorCount:** El número máximo de intentos para reiniciar una tarea cuando se produce un error del entorno. Después de que el sistema intenta reiniciar la tarea el número de veces establecido, la tarea se detiene y se requiere intervención manual. El valor predeterminado es -1, que indica a AWS DMS que intente reiniciar la tarea de forma indefinida. Establezca este valor en 0 para no intentar nunca reiniciar una tarea. Si se produce un error fatal, AWS DMS deja de intentar reiniciar la tarea después de seis intentos.
- **RecoverableErrorInterval:** El número de segundos que espera AWS DMS entre intentos para reiniciar una tarea. El valor predeterminado es 5.
- **RecoverableErrorThrottling:** Cuando está habilitado, se incrementa el intervalo entre los intentos de reiniciar una tarea cada vez que se hace una tentativa. El valor predeterminado es `true`.
- **RecoverableErrorThrottlingMax:** El número máximo de segundos que espera AWS DMS entre intentos para reiniciar una tarea si `RecoverableErrorThrottling` está habilitado. El valor predeterminado es 1800.
- **ApplyErrorDeletePolicy:** Determina la acción que va emprender AWS DMS cuando existe un conflicto con una operación `DELETE`. El valor predeterminado es `IGNORE_RECORD`.
  - `IGNORE_RECORD`: La tarea continúa y los datos de este registro se omiten. El recuento de errores de la propiedad `ApplyErrorEscalationCount` se incrementa. Por lo tanto, si se establece un límite en los errores de una tabla, este error cuenta para el límite.
  - `LOG_ERROR`: La tarea continúa y el error se escribe en el registro de tareas.
  - `SUSPEND_TABLE`: La tarea continúa, pero los datos de la tabla con el registro de errores se pasan a un estado de error y los datos no se replican.
  - `STOP_TASK`: La tarea se detiene y se necesitará intervención manual.
- **ApplyErrorInsertPolicy:** Determina la acción que va emprender AWS DMS cuando existe un conflicto con una operación `INSERT`. El valor predeterminado es `LOG_ERROR`.
  - `IGNORE_RECORD`: La tarea continúa y los datos de este registro se omiten. El recuento de errores de la propiedad `ApplyErrorEscalationCount` se incrementa. Por lo tanto, si se establece un límite en los errores de una tabla, este error cuenta para el límite.
  - `LOG_ERROR`: La tarea continúa y el error se escribe en el registro de tareas.
  - `SUSPEND_TABLE`: La tarea continúa, pero los datos de la tabla con el registro de errores se pasan a un estado de error y los datos no se replican.
  - `STOP_TASK`: La tarea se detiene y se necesitará intervención manual.
  - `INSERT_RECORD`: Si hay un registro de destino con la misma clave principal que el registro de origen que se insertó, el registro de destino se actualiza.

- **ApplyErrorUpdatePolicy:** Determina la acción que va emprender AWS DMS cuando existe un conflicto con una operación UPDATE. El valor predeterminado es LOG\_ERROR.
  - IGNORE\_RECORD: La tarea continúa y los datos de este registro se omiten. El recuento de errores de la propiedad ApplyErrorEscalationCount se incrementa. Por lo tanto, si se establece un límite en los errores de una tabla, este error cuenta para el límite.
  - LOG\_ERROR: La tarea continúa y el error se escribe en el registro de tareas.
  - SUSPEND\_TABLE: La tarea continúa, pero los datos de la tabla con el registro de errores se pasan a un estado de error y los datos no se replican.
  - STOP\_TASK: La tarea se detiene y se necesitará intervención manual.
  - UPDATE\_RECORD: si falta el registro de destino, este se inserta en la tabla de destino. Si se selecciona esta opción, es necesario habilitar el registro suplementario completo para todas las columnas en la tabla de origen cuando Oracle sea la base de datos de origen.
- **ApplyErrorEscalationPolicy:** Determina la acción que emprende AWS DMS cuando se alcanza el número máximo de errores (establecido utilizando el parámetro ApplyErrorsEscalationCount).
  - LOG\_ERROR: La tarea continúa y el error se escribe en el registro de tareas.
  - SUSPEND\_TABLE: La tarea continúa, pero los datos de la tabla con el registro de errores se pasan a un estado de error y los datos no se replican.
  - STOP\_TASK: La tarea se detiene y se necesitará intervención manual.
- **ApplyErrorEscalationCount:** Establece el número máximo de conflictos APPLY que pueden producirse en una tabla específica durante una operación del proceso de cambio. Cuando se alcanza esta cifra, los datos de la tabla se gestionan de acuerdo con la política establecida en el parámetro ApplyErrorEscalationPolicy. El valor predeterminado es 0.
- **ApplyErrorFailOnTruncationDdl:** Establezca esta variable en true para provocar el fracaso de la tarea cuando se realiza el truncado en cualquiera de las tablas a las que se ha hecho un seguimiento durante la CDC. El mensaje de error es: "Truncation DDL detected". El valor predeterminado es false.

Este método no funciona con PostgreSQL o cualquier otro punto de enlace de origen que no replica el truncado de la tabla DDL.

- **FailOnNoTablesCaptured:** Establezca esta variable en true para provocar el fracaso de la tarea cuando las reglas de transformación definidas para una tarea no encuentran tablas cuando la tarea comienza. El valor predeterminado es false.
- **FailOnTransactionConsistencyBreached:** Esta opción se aplica a tareas que utilizan Oracle como origen con CDC. Establezca esta variable en true para provocar el fracaso de la tarea cuando una transacción está abierta durante más tiempo que el tiempo de espera especificado y podría abandonarse.

Cuando una tarea de CDC comienza con Oracle, AWS DMS espera durante un periodo de tiempo limitado para que la transacción abierta más antigua se cierre antes de iniciar CDC. Si la transacción abierta más antigua no se cierra hasta que se alcanza el tiempo de espera, entonces normalmente iniciamos CDC igualmente, no haciendo caso a esa transacción. Si esta configuración está definida como true, la tarea falla.

- **FullLoadIgnoreConflicts:** establezca esta variable en false para que AWS DMS no haga caso a errores del tipo "zero rows affected" y "duplicates" al aplicar eventos almacenados en caché. Si se establece en true, AWS DMS informa todos los errores en lugar de no hacerles caso. El valor predeterminado es true.

## Guardar la configuración de las tareas

Puede guardar la configuración de una tarea como archivo JSON, en caso de que desee reutilizar la configuración para otra tarea.

Por ejemplo, el siguiente archivo JSON contiene valores de configuración que se guardaron para una tarea.

```
{  
    "TargetMetadata": {  
        "TargetSchema": "",  
        "SupportLobs": true,  
        "FullLobMode": false,  
        "LobChunkSize": 64,  
        "LimitedSizeLobMode": true,  
        "LobMaxSize": 32,  
        "BatchApplyEnabled": true  
    },  
    "FullLoadSettings": {  
        "TargetTablePrepMode": "DO_NOTHING",  
        "CreatePkAfterFullLoad": false,  
        "StopTaskCachedChangesApplied": false,  
        "StopTaskCachedChangesNotApplied": false,  
        "MaxFullLoadSubTasks": 8,  
        "TransactionConsistencyTimeout": 600,  
        "CommitRate": 10000  
    },  
    "Logging": {  
        "EnableLogging": false  
    },  
    "ControlTablesSettings": {  
        "ControlSchema": "",  
        "HistoryTimeslotInMinutes": 5,  
        "HistoryTableEnabled": false,  
        "SuspendedTablesTableEnabled": false,  
        "StatusTableEnabled": false  
    },  
    "StreamBufferSettings": {  
        "StreamBufferCount": 3,  
        "StreamBufferSizeInMB": 8  
    },  
    "ChangeProcessingTuning": {  
        "BatchApplyPreserveTransaction": true,  
        "BatchApplyTimeoutMin": 1,  
        "BatchApplyTimeoutMax": 30,  
        "BatchApplyMemoryLimit": 500,  
        "BatchSplitSize": 0,  
        "MinTransactionSize": 1000,  
        "CommitTimeout": 1,  
        "MemoryLimitTotal": 1024,  
        "MemoryKeepTime": 60,  
        "StatementCacheSize": 50  
    },  
    "ChangeProcessingDdlHandlingPolicy": {  
        "HandleSourceTableDropped": true,  
        "HandleSourceTableTruncated": true,  
        "HandleSourceTableAltered": true  
    },  
    "ErrorBehavior": {  
        "DataErrorPolicy": "LOG_ERROR",  
        "DataTruncationErrorPolicy": "LOG_ERROR",  
        "DataErrorEscalationPolicy": "SUSPEND_TABLE",  
        "DataErrorEscalationCount": 50,  
        "TableErrorPolicy": "SUSPEND_TABLE",  
        "TableErrorEscalationPolicy": "STOP_TASK",  
        "TableErrorEscalationCount": 50,  
        "RecoverableErrorCount": 0,  
        "RecoverableErrorInterval": 5,  
        "RecoverableErrorThrottling": true,  
        "RecoverableErrorThrottlingMax": 1800,  
        "ApplyErrorDeletePolicy": "IGNORE_RECORD",  
        "ApplyErrorInsertPolicy": "LOG_ERROR",  
        "ApplyErrorUpdatePolicy": "LOG_ERROR"  
    }  
}
```

```
        "ApplyErrorUpdatePolicy": "LOG_ERROR",
        "ApplyErrorEscalationPolicy": "LOG_ERROR",
        "ApplyErrorEscalationCount": 0,
        "FullLoadIgnoreConflicts": true
    }
}
```

## Establecimiento de compatibilidad con LOB para bases de datos de origen en la tarea de AWS DMS

A veces, migrar objetos binarios grandes (LOB) entre sistemas resulta complicado. AWS DMS ofrece una serie de opciones para ayudar a ajustar las columnas de LOB. Para saber qué y cuándo AWS DMS considera los tipos de datos como LOB, consulte la documentación de AWS DMS.

Al migrar datos desde una base de datos a otra, puede aprovechar para replantearse la manera en que quiere almacenar los LOB, especialmente en migraciones heterogéneas. Si desea hacerlo, no será necesario migrar los datos LOB.

Si decide incluir LOB, puede decidir a continuación las demás configuraciones para LOB:

- El modo LOB determina cómo se gestionan los LOB:
  - Full LOB mode (Modo de LOB completo): en el modo de LOB completo, AWS DMS migra todos los LOB del origen al destino sin importar el tamaño. En esta configuración, AWS DMS no tiene información sobre el tamaño máximo de los LOB. Por lo tanto, los LOB se migran de uno en uno. El modo LOB completo puede ser bastante lento.
  - Limited LOB mode (Modo de LOB limitado): en el modo de LOB limitado, usted establece el tamaño máximo de LOB que AWS DMS debe aceptar. Esto permite que AWS DMS reasigne previamente la memoria y cargue los datos LOB en masa. Los LOB que superen el tamaño máximo se truncarán y se emitirá una advertencia para el archivo de registro. En Limited LOB mode conseguirá una mejora significativa del desempeño frente a Full LOB mode. Le recomendamos que utilice Limited LOB mode siempre que sea posible.

### Note

Con Oracle, los LOB se tratan como tipos de datos VARCHAR, siempre que sea posible. Este enfoque significa que AWS DMS los recopila en masa desde la base de datos, lo que es significativamente más rápido que otros métodos. El tamaño máximo de un VARCHAR en Oracle es de 64 K. Por lo tanto un tamaño de LOB limitado de menos de 64 K es óptimo si Oracle es la base de datos de origen.

- Cuando se configura una tarea para que se ejecute en modo de LOB limitado, la opción Max LOB size (K) [Tamaño máximo de LOB (K)] establece el tamaño máximo del LOB que AWS DMS acepta. Cualquier LOB que sea superior a este valor se trunca en este valor.
- Cuando se configura una tarea para utilizar el modo de LOB completo, AWS DMS recupera los LOB en partes. La opción LOB chunk size (K) determina el tamaño de cada pieza. Al configurar esta opción, tenga en cuenta el tamaño máximo de paquete que permite su configuración de red. Si el tamaño del fragmento LOB supera el tamaño máximo permitido del paquete, es posible que vea errores de desconexión.

Para obtener más información sobre la configuración de las tareas para especificar estas opciones, consulte [Configuración de las tareas de los metadatos de destino \(p. 263\)](#)

## Crear varias tareas

En algunas situaciones de migración, es posible que tenga que crear varias tareas de migración. Las tareas funcionan de forma independiente y se pueden ejecutar a la vez. Cada tarea tiene su propia carga inicial, CDC y proceso de lectura de registros. Las tablas que se relacionan mediante lenguaje de manipulación de datos (DML) deben formar parte de la misma tarea.

Entre los motivos para crear varias tareas para una migración, se incluyen los siguientes:

- Las tablas de destino para las tareas residen en diferentes bases de datos, como, por ejemplo, cuando un sistema se divide en varios sistemas.
- Puede dividir la migración de una tabla grande en varias tareas con los filtros.

### Note

Dado que cada tarea tiene su propia captura de cambios y proceso de lectura de registros, los cambios no se coordinan en las tareas. Por tanto, cuando se utilizan varias tareas para realizar una migración, asegúrese de que las transacciones del origen se encuentran en su totalidad en una sola tarea.

## Creación de tareas para la replicación continua con AWS DMS

Puede crear una tarea de AWS DMS que capture los cambios en curso del almacén de base de datos de origen. Para ello, puede capturar al mismo tiempo que migra los datos. También puede crear una tarea que capture los cambios continuos después de completar la migración inicial a un almacén de datos de destino compatible. Este proceso se denomina replicación continua o captura de datos de cambios (CDC). AWS DMS utiliza este proceso al replicar los cambios continuos desde un almacén de datos de origen. Este proceso funciona recopilando los cambios en logs de la base de datos con la API nativa del motor de base de datos.

Cada motor origen tiene requisitos de configuración específicos para exponer el flujo de cambios a una determinada cuenta de usuario. La mayoría de los motores necesitan algún tipo de configuración adicional para que el proceso de captura pueda consumir los datos de forma provechosa, sin perder información. Por ejemplo, con Oracle es necesario añadir un modo de registro complementario y, con MySQL, un sistema de registro binario en el nivel de fila.

Para leer los cambios continuos de la base de datos de origen, AWS DMS utiliza acciones de la API específicas del motor para la lectura de los cambios de los logs de transacciones del motor de origen. A continuación se incluyen algunos ejemplos de cómo AWS DMS realiza esta operación:

- Para Oracle: AWS DMS utiliza la API de Oracle LogMiner o de Binary Reader (API bfile) para leer los cambios continuos. AWS DMS lee los cambios continuos de los registros REDO archivados u online en función del número de cambio del sistema (SCN).
- En Microsoft SQL Server, AWS DMS utiliza MS-Replication o MS-CDC para escribir información en el registro de transacciones de SQL Server. A continuación, utiliza la función `fnc_dblog()` o `fnc_dump_dblog()` de SQL Server para leer los cambios del registro de transacciones basándose en el número de secuencia del registro (LSN).
- En MySQL, AWS DMS lee los cambios de los registros binarios (binlogs) basados en filas y migra dichos cambios al destino.
- En PostgreSQL, AWS DMS configura ranuras de replicación lógica, utiliza el complemento `test_decoding` para leer los cambios del origen y migra dichos cambios al destino.

- Cuando use Amazon RDS como origen, le recomendamos que se asegure de que los backups están habilitados para configurar CDC. También le recomendamos que se asegure de que la base de datos de origen esté configurada para conservar los logs de cambios durante un tiempo suficiente (24 horas suele ser suficiente).

Hay dos tipos de tareas de replicación continua:

- Carga completa más CDC: la tarea migra los datos existentes y, a continuación, actualiza la base de datos de destino en función de los cambios realizados en la base de datos de origen.
- Solo CDC: la tarea migra los cambios continuos una vez que los datos están en la base de datos de destino.

## Realizar la replicación empezando desde un punto de inicio de CDC

Puede iniciar una tarea de replicación continua de AWS DMS (cambiar solo la captura de datos) desde varios puntos. Esto incluye lo siguiente:

- From a custom CDC start time (A partir de una hora de inicio de CDC personalizada): puede usar la Consola de administración de AWS o AWS CLI para proporcionar a AWS DMS una marca temporal en el punto donde desea que se inicie la replicación. AWS DMS inicia una tarea de replicación continua a partir de esta hora de inicio de CDC personalizada. AWS DMS convierte la marca temporal especificada (en UTC) en un punto de inicio nativo, como un LSN para SQL Server o un SCN para Oracle. AWS DMS utiliza métodos específicos del motor para determinar dónde debe iniciar exactamente la tarea de migración en función del flujo de cambios del motor de origen.

### Note

PostgreSQL como origen no admite una hora de inicio CDC personalizada. Esto se debe a que el motor de base de datos de PostgreSQL no tiene forma de asignar una marca temporal a un LSN o SCN como Oracle y SQL Server.

- From a CDC native start point (A partir de un punto de inicio nativo de CDC): también puede iniciar desde un punto nativo en el registro de transacciones del motor de origen. En algunos casos, es posible que prefiera este enfoque, ya que una marca temporal puede indicar varios puntos nativos en el registro de transacciones. AWS DMS admite esta característica para los siguientes puntos de enlace de origen:
  - SQL Server
  - Oracle
  - MySQL

## Determinar un punto de inicio de CDC nativo

Un punto de inicio nativo de CDC es un punto en el registro del motor de base de datos que define una hora a la que se puede iniciar la captura de datos de cambios (CDC). Por ejemplo, suponga que un volcado de datos masivos se ha aplicado al destino a partir de un punto en el tiempo. En este caso, puede consultar el punto de inicio nativo para la tarea solo de replicación continua a partir de un punto anterior a la realización del volcado.

Los siguientes ejemplos muestran cómo puede encontrar el punto de inicio de CDC nativo de motores de origen admitidos:

### SQL Server

En SQL Server, un número de secuencia de registro (LSN) tiene tres partes:

- Número de secuencia del archivo registro virtual (VLF)
- Desplazamiento inicial de un bloque del registro
- Número de ranura

Un LSN de ejemplo es: 00000014:00000061:0001

Para obtener el punto de inicio de una tarea de migración de SQL Server en función de la configuración de copias de seguridad de registros de transacciones, use la función `fn_dblog()` o `fn_dump_dblog()` de SQL Server.

Para utilizar un punto de inicio nativo de CDC con SQL Server, debe crear una publicación para cada tabla que participe en la replicación continua. Para obtener más información acerca de la creación de una publicación, consulte [Creación de una publicación de SQL Server para la replicación continua \(p. 112\)](#). AWS DMS crea la publicación automáticamente cuando se utiliza CDC sin un punto de inicio nativo.

#### Oracle

Un número de cambio del sistema (SCN) es una marca temporal interna lógica que utilizan las bases de datos de Oracle. Se usa para ordenar los eventos que se producen dentro de la base de datos, lo cual es necesario para satisfacer las propiedades de atomicidad, uniformidad, aislamiento y durabilidad (ACID) de una transacción. Las bases de datos de Oracle utilizan los SCN para marcar la ubicación donde todos los cambios se han escrito en el disco, de modo que una acción de recuperación no aplique los cambios que ya se han escrito. Oracle también utiliza los SCN para marcar el punto en el que no hay nada que rehacer en un conjunto de datos, de modo que la recuperación pueda detenerse. Para obtener más información acerca de Oracle SCNs, consulte la [documentación de Oracle](#).

Para obtener el SCN actual de una base de datos de Oracle, ejecute el siguiente comando:

```
SELECT current_scn FROM V$DATABASE
```

#### MySQL

Antes del lanzamiento de MySQL versión 5.6.3, el número de secuencia de registro (LSN) de MySQL era un número entero sin firmar de 4 bytes. En MySQL versión 5.6.3, en la que el límite de tamaño de archivos de registro redo aumentó de 4 GB a 512 GB, el LSN pasó a ser un número entero sin firmar de 8 bytes. El aumento refleja que se requerían bytes adicionales para almacenar información de tamaño adicional. Las aplicaciones creadas en MySQL 5.6.3 o versiones posteriores que utilizan valores LSN deben usar variables de 64 bits, en lugar de 32 bits, para almacenar y comparar los valores LSN. Para obtener más información acerca de los LSN de MySQL, consulte la [documentación de MySQL](#).

Para obtener el LSN actual de una base de datos de MySQL, ejecute el siguiente comando:

```
mysql> show master status;
```

La consulta devuelve un nombre de archivo binlog, la posición y otros valores. El punto de inicio nativo de CDC es una combinación del nombre y la posición del archivo binlogs, como `mysql-bin-changelog.000024:373`. En este ejemplo, `mysql-bin-changelog.000024` es el nombre de archivo binlogs y 373 es la posición en la que AWS DMS debe empezar a capturar los cambios.

## Uso de un punto de comprobación como punto de inicio de CDC

Una tarea de replicación continua migra los cambios y AWS DMS almacena en caché la información de los puntos de comprobación específica de AWS DMS de vez en cuando. El punto de comprobación que

AWS DMS crea información para que el motor de replicación sepa el punto de recuperación del flujo de cambios. Puede utilizar el punto de comprobación para retroceder en la escala de tiempo de los cambios y recuperar una tarea de migración que haya producido un error. También puede utilizar un punto de comprobación para iniciar otra tarea de replicación continua para otro destino en cualquier punto en el tiempo.

Puede obtener la información del punto de comprobación de una de las siguientes dos formas:

- Ejecute el comando de la API `DescribeReplicationTasks` y consulte los resultados. Puede filtrar la información por tarea y buscar el punto de comprobación. Puede recuperar el último punto de comprobación cuando la tarea está detenida o tenga el estado de error.
- Consulte la tabla de metadatos denominada `awsdms_txn_state` en la instancia de destino. Puede consultar la tabla para obtener información del punto de comprobación. Para crear la tabla de metadatos, establezca el parámetro `TaskRecoveryTableEnabled` en `Yes` cuando cree una tarea. Este valor hace que AWS DMS escriba la información del punto de comprobación en la tabla de metadatos de destino de forma continua. Esta información se pierde si se elimina la tarea.

Por ejemplo, el siguiente es un ejemplo de punto de comprobación en la tabla de metadatos:  
`checkpoint:V1#34#00000132/0F000E48#0#0#*#0#121`

## Detener una tarea en un punto en el tiempo de confirmación o del servidor

Con la introducción de los puntos de inicio nativos de CDC, AWS DMS también puede detener una tarea en los puntos siguientes:

- Un punto en el tiempo de confirmación en el origen
- Un punto en el tiempo de servidor en la instancia de replicación

Puede modificar una tarea y establecer una hora en UTC para detener una tarea según sea necesario. La tarea se detiene automáticamente en función de la hora de confirmación o de servidor que haya establecido. Además, si conoce un momento en el tiempo adecuado en el que detener la tarea de migración cuando crea la tarea, también puede establecer una hora de detención cuando la cree.

## Modificar una tarea

Puede modificar una tarea si necesita cambiar la configuración de las tareas, la correspondencia de tablas u otros ajustes. Las tareas se modifican en la consola de DMS y, para ello, debe seleccionar la tarea y elegir `Modify`. También puede utilizar la AWS CLI o el comando `ModifyReplicationTask` de la API de AWS DMS.

Existen algunas limitaciones a la hora de modificar una tarea. Entre ellas se incluyen:

- No puede modificar el punto de enlace del origen o el destino de una tarea.
- No puede cambiar el tipo de migración de una tarea.
- Una tarea que se haya ejecutado debe tener el estado `Stopped` o `Failed` para poder modificarla.

## Volver a cargar tablas durante una tarea

Mientras se ejecuta una tarea, puede volver a cargar una tabla de base de datos de destino con los datos del origen. Es posible que desee volver a cargar una tabla si, durante la tarea, se produce un error o

cambios de datos a causa de las operaciones de partición (por ejemplo, cuando se utiliza Oracle). Puede volver a cargar hasta 10 tablas desde una tarea.

Para volver a cargar una tabla, se aplicarán las siguientes condiciones:

- La tarea se debe estar ejecutando.
- El método de migración para la tarea debe ser Full Load o Full Load with CDC.
- No se permite duplicar tablas.
- AWS DMS conserva la definición de la tabla leída con anterioridad y no la vuelve a crear durante la operación de recarga. Si se han ejecutado instrucciones DDL como ALTER TABLE ADD COLUMN o DROP COLUMN en la tabla antes de que esta se vuelva a cargar, es posible que se produzca un error en la operación de recarga.

## Consola de administración de AWS

Para volver a cargar una tabla con la consola de AWS DMS

1. Inicie sesión en la Consola de administración de AWS y elija AWS DMS. Si está registrado como usuario de AWS Identity and Access Management (IAM), debe disponer de los permisos adecuados para acceder a AWS DMS. Para obtener más información sobre los permisos que se necesitan, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).
2. Elija Tasks en el panel de navegación.
3. Elija la tarea en ejecución que tiene la tabla que desea volver a cargar.
4. Elija la pestaña Table Statistics (Estadísticas de la tabla).

Create task   Assess   Modify   Start/Resume   Stop

Filter:  Filter

ID	Status	Source
move-data	Running	from-mysql-sou

move-data

Overview   Task monitoring   **Table statistics**   Logs   A

**Reload table data**

Revalidate

Filter:  Filter

Schema	Table	Load State	Inserts	Deletes
employees	departments	Table completed	0	0
employees	dept_emp	Table completed	0	0
employees	dept_manager	Table completed	0	0

5. Elija la tabla que desea volver a cargar. Si la tarea ya no se está ejecutando, no puede volver a cargar la tabla.
6. Elija Reload table data (Volver a cargar datos de la tabla).

Cuando AWS DMS se está preparando para volver a cargar una tabla, la consola cambia el estado de la tabla a Table is being reloaded (Se está volviendo a cargar la tabla).

## Uso de la correspondencia de tablas para especificar la configuración de tareas

La correspondencia de tablas utiliza diversos tipos de reglas para especificar el origen de datos, el esquema origen, los datos y las transformaciones que deben producirse durante la tarea. Puede utilizar la correspondencia de tablas para especificar las tablas individuales en una base de datos que se van a migrar y el esquema que se va a utilizar en la migración. Además, puede utilizar filtros para especificar los datos de una columna determinada de la tabla que desea replicar. Puede utilizar transformaciones para modificar los datos escritos en la base de datos de destino.

### Especificación de selección de tabla y transformaciones mediante correspondencia de tabla desde la consola

Puede utilizar la consola de Consola de administración de AWS para realizar la correspondencia de tablas, lo que incluye especificar la selección y la transformación de la tabla. En la consola, utilice la sección Where (Dónde) para especificar el esquema, la tabla y la acción (incluir o excluir). Utilice la sección Filter (Filtro) para especificar el nombre de la columna en una tabla y las condiciones que desea aplicar a una tarea de replicación. En su conjunto, estas dos acciones crean una regla de selección.

Puede incluir transformaciones en una correspondencia de tablas después de que haya especificado al menos una regla de selección. Puede utilizar transformaciones para cambiar el nombre de un esquema o tabla, añadir prefijos o sufijos a un esquema o tabla o eliminar una columna de la tabla.

El siguiente ejemplo ilustra cómo configurar las reglas de selección de una tabla denominada **Customers** en un esquema llamado **EntertainmentAgencySample**. Crea las reglas de selección y transformaciones en la pestaña Guided (Guiada). Esta pestaña solo aparece cuando dispone de un punto de enlace de origen que tiene información sobre el esquema y la tabla.

Para especificar una tabla de selección, filtre los criterios y las transformaciones con la consola de AWS

1. Inicie sesión en la Consola de administración de AWS y elija AWS DMS. Si está registrado como usuario de AWS Identity and Access Management (IAM), debe disponer de los permisos adecuados para acceder a AWS DMS. Para obtener más información sobre los permisos que se necesitan, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).
2. En la página Dashboard, elija Tasks.
3. Seleccione Create Task.
4. Escriba la información de la tarea, incluidos Task name, Replication instance, Source endpoint, Target endpoint y Migration type. Elija Guided (Guiada) en la sección Table mappings (Correspondencias de tabla).

## Create task

A task can contain one or more table mappings which define what data is moved from the source to the target. If a table does not exist on the target, it can be created automatically.

The screenshot shows the 'Create task' interface. At the top, there are five input fields: 'Task name\*' (sg-selecttransfrm), 'Replication instance\*' (repl-inst-or2pg - vpc-fd...), 'Source endpoint\*' (sg-mysql2pg-source2), 'Target endpoint\*' (postgres-target), and 'Migration type\*' (Migrate existing data). Below these is a checkbox 'Start task on create' with a checked status. Underneath the form, there are two navigation links: 'Task Settings' (highlighted with a blue arrow) and 'Table mappings'. At the bottom of the screen, there are two tabs: 'Guided' (which is selected) and 'JSON'.

Task name\* sg-selecttransfrm

Replication instance\* repl-inst-or2pg - vpc-fd...

Source endpoint\* sg-mysql2pg-source2

Target endpoint\* postgres-target

Migration type\* Migrate existing data

Start task on create

Task Settings

Table mappings

Guided JSON

5. En la sección Table mappings (Correspondencias de tabla), elija el nombre del esquema y el nombre de la tabla. Puede utilizar "%" como un valor comodín cuando especifique el nombre de la tabla. Especifique la acción que va a emprender, para incluir o excluir los datos que definió el filtro.

Start task on create

Task Settings

Table mappings

**Guided** **JSON**

Selection rules ?

At least one selection rule with an include action is required. Once you have one or more selection rules, you can add transformation rules.

Where ?

Schema name is EntertainmentAgencyExample

Table name is like %

Use % as a wildcard.

Action **Include** ?

Filter ?

Add column filter

Add selection rule

The screenshot shows the 'Selection rules' configuration screen in the AWS Database Migration Service. The 'Guided' tab is highlighted with a red circle. The 'Where' section contains a schema name and a table name pattern. The 'Action' is set to 'Include'. There is a 'Filter' section with a 'Add column filter' link. A blue 'Add selection rule' button is at the bottom right.

6. Especifique la información del filtro mediante los enlaces Add column filter (Añadir filtro de columna) y Add condition (Añadir condición).
  - a. Elija Add column filter (Añadir filtro de columna) para especificar una columna y las condiciones.
  - b. Elija Add condition (Añadir condición) para añadir otras condiciones adicionales.

El siguiente ejemplo muestra un filtro para la tabla **Customers** que incluye **AgencyIDs** entre **01** y **85**.

**Selection rules** ⓘ

At least one selection rule with an include action is required. Once you have one or more selection rules, you can add transformation rules.

**Where** ⓘ

Schema name is EntertainmentAgency... ▾  
Table name is like Customers  
Use % as a wildcard.

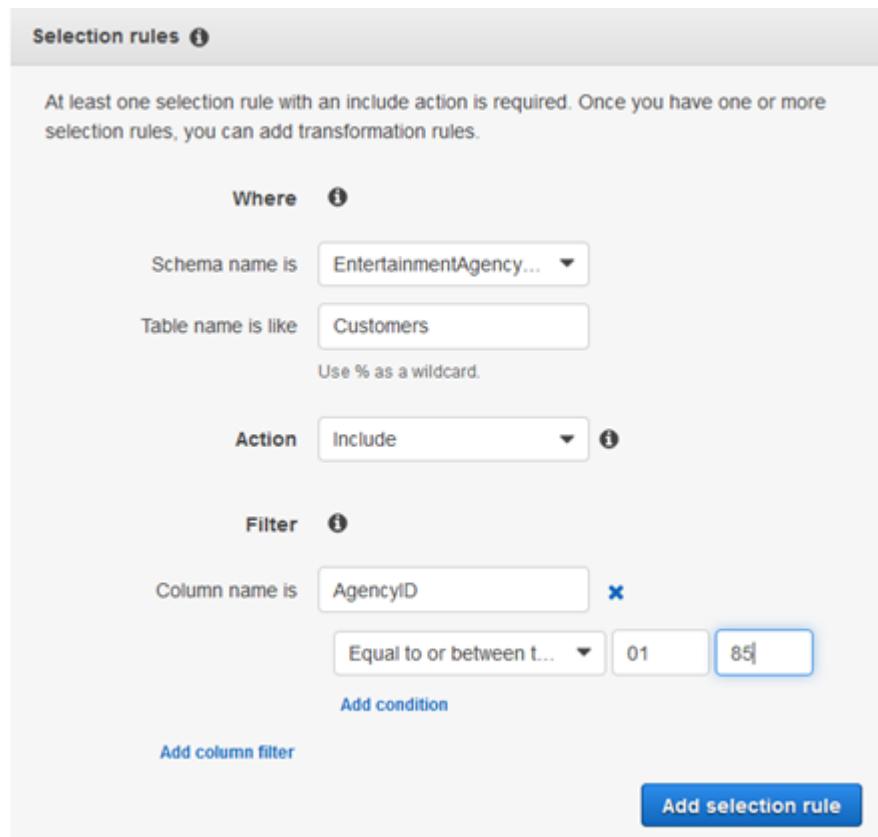
**Action** ⓘ

Include ▾ ⓘ

**Filter** ⓘ

Column name is AgencyID ✖  
Equal to or between t... ▾ 01 85  
Add condition  
Add column filter

Add selection rule



7. Cuando haya creado las selecciones que deseé, elija Add selection rule (Añadir regla de selección).
8. Una vez que haya creado al menos una regla de selección, puede añadir una transformación a la tarea. Elija Add transformation rule (Añadir regla de transformación).

▼ Table mappings

Guided JSON

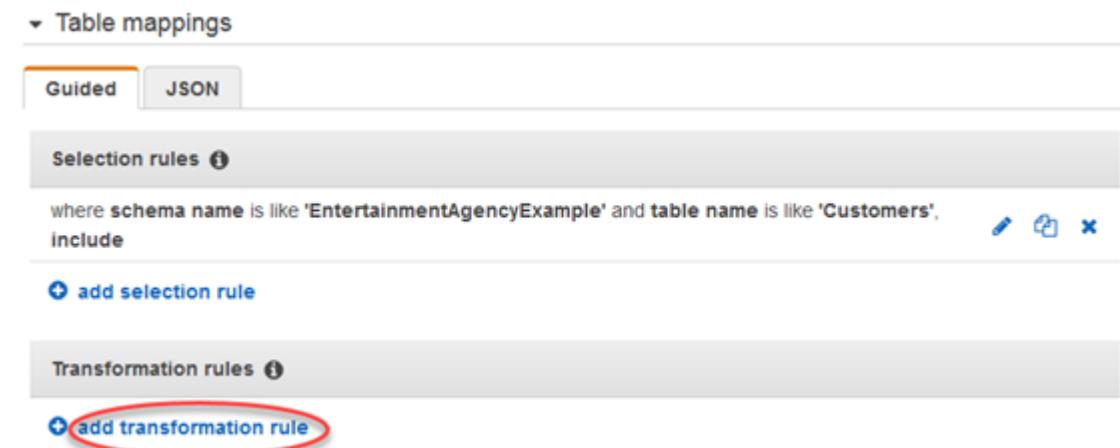
**Selection rules** ⓘ

where schema name is like 'EntertainmentAgencyExample' and table name is like 'Customers'.  
include

+ add selection rule

**Transformation rules** ⓘ

+ add transformation rule



9. Elija el destino que desea transformar e introduzca la información adicional solicitada. El siguiente ejemplo muestra una transformación que elimina la columna **AgencyStatus** de la tabla **Customer**.

The screenshot shows the 'Transformation rules' configuration screen. At the top, there is a 'Selection rules' section with a note: 'where schema name is like 'EntertainmentAgencyExample' and table name is like 'Customers''. Below it is a 'Transformation rules' section with a 'Target' dropdown set to 'Column'. The 'Where' section contains three filters: 'Schema name is EntertainmentAgencyExample', 'Table name is like Customer', and 'Column name is like AgencyStatus'. The 'Action' section has a dropdown set to 'Remove column'. At the bottom right are 'cancel' and 'Add transformation rule' buttons.

10. Elija Add transformation rule.
11. (Opcional) Añada transformaciones o reglas de selección adicionales eligiendo add selection rule (añadir selección de regla) o add transformation rule (añadir regla de transformación). Cuando haya terminado, seleccione Create task.

The screenshot shows the 'Create task' wizard interface. At the top, there are two tabs: 'Guided' (selected) and 'JSON'. Below the tabs, there are two sections: 'Selection rules' and 'Transformation rules'. The 'Selection rules' section contains a rule: 'where schema name is like 'EntertainmentAgencyExample' and table name is like 'Customer'', with an 'add selection rule' button. The 'Transformation rules' section contains a rule: 'For column where schema name is like 'EntertainmentAgencyExample' and table name is like 'Customer' and column name is like 'AgencyStatus', remove column', with an 'add transformation rule' button. Both the 'add selection rule' and 'add transformation rule' buttons are circled in red. At the bottom right, there are 'Cancel' and 'Create task' buttons.

## Especificación de selección de tabla y transformaciones mediante correspondencia de tabla con JSON

Puede crear correspondencias de tablas en formato JSON. Si crea una tarea de migración con la consola de administración de AWS DMS, puede ingresar el JSON directamente en el cuadro de mapeo de tablas. Si utiliza la CLI o la API para realizar migraciones, puede crear un archivo JSON para especificar los mapeos de tablas que deseé que se aplique durante la migración.

Puede especificar las tablas o los esquemas con los que desea trabajar y puede realizar transformaciones de esquemas y tablas. Puede crear reglas de correspondencia de tablas con los tipos de regla `selection` y `transformation`.

### Note

Para destinos de Amazon S3, también puede etiquetar los objetos de S3 asignados a tablas y esquemas seleccionados utilizando el tipo de regla `post-processing` y la acción de regla `add-tag`. Para obtener más información, consulte [Etiquetado de objetos de Amazon S3 \(p. 192\)](#).

## Reglas de selección y acciones

Si utiliza la correspondencia de tablas, puede especificar con qué tablas o esquemas desea trabajar mediante el uso de reglas de selección y acciones. Para las reglas de correspondencia de tablas utilice el tipo de regla de selección; se pueden aplicar los valores siguientes.

Parámetro	Valores posibles	Descripción
<code>rule-type</code>	<code>selection</code>	Defina al menos una regla de selección cuando especifique un mapeo de tablas.
<code>rule-id</code>	Un valor numérico.	Un único valor numérico para identificar la regla.

Parámetro	Valores posibles	Descripción
rule-name	Un valor alfanumérico.	Un nombre exclusivo para identificar la regla.
rule-action	include, exclude, explicit	Incluya o excluya el objeto u objetos que ha seleccionado la regla. Si se especifica explicit, puede seleccionar e incluir solo un objeto que corresponde a una tabla y esquema especificados de forma explícita.
object-locator	Un objeto con los siguientes parámetros: <ul style="list-style-type: none"><li>• schema-name: el nombre del esquema.</li><li>• table-name: el nombre de la tabla.</li></ul>	El nombre de cada esquema y tabla al que se aplica la regla. Si la rule-action es include o bien exclude, puede utilizar el signo de porcentaje "%" como comodín para todo o parte del valor de cada parámetro object-locator. Así, puede hacer coincidir estos elementos: <ul style="list-style-type: none"><li>• Una sola tabla en un esquema único</li><li>• Una sola tabla en varios o todos los esquemas</li><li>• Algunas o todas las tablas en un esquema único</li><li>• Algunas o todas las tablas en algunos o todos los esquemas</li></ul> Si la rule-action es explicit, solo puede especificar el nombre exacto de una única tabla y esquema (sin comodines).
load-order	Un número entero. El valor máximo es 2147483647.	Indica la prioridad para cargar tablas. Las tablas con los valores más altos se cargan en primer lugar.
filters	Una matriz de objetos.	Uno o más objetos para filtrar el origen. Especifique parámetros de objetos para filtrar en una única columna en el origen. Especifique varios objetos para filtrar en varias columnas. Para obtener más información, consulte <a href="#">Usar filtros de origen (p. 317)</a> .

#### Example Migrar todas las tablas de un esquema

El siguiente ejemplo migra todas las tablas desde un esquema denominado **Test** en el origen al punto de enlace de destino.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-action": "include",
            "schema-name": "Test",
            "table-name": "*"
        }
    ]
}
```

```
        "rule-name": "1",
        "object-locator": {
            "schema-name": "Test",
            "table-name": "%"
        },
        "rule-action": "include"
    }
]
}
```

#### Example Migrar algunos tablas en un esquema

En el siguiente ejemplo, se migran todas las tablas excepto aquellas que comienzan con **DMS** desde un esquema llamado **Test** en el origen hasta el punto de enlace de destino.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "Test",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "selection",
            "rule-id": "2",
            "rule-name": "2",
            "object-locator": {
                "schema-name": "Test",
                "table-name": "DMS%"
            },
            "rule-action": "exclude"
        }
    ]
}
```

#### Example Migrar una única tabla especificada en un solo esquema

El siguiente ejemplo migra la tabla **Customer** desde el esquema **NewCust** en el origen al punto de enlace de destino.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "NewCust",
                "table-name": "Customer"
            },
            "rule-action": "explicit"
        }
    ]
}
```

Note

Puede seleccionar de forma explícita en varias tablas y esquemas especificando varias reglas de selección.

Example Migrar tablas en un orden establecido

El siguiente ejemplo migra dos tablas. La tabla `loadfirst` (con prioridad 2) se migra antes que la tabla `loadsecond`.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "1",  
            "rule-name": "1",  
            "object-locator": {  
                "schema-name": "Test",  
                "table-name": "loadfirst"  
            },  
            "rule-action": "include",  
            "load-order": "2"  
        },  
        {  
            "rule-type": "selection",  
            "rule-id": "2",  
            "rule-name": "2",  
            "object-locator": {  
                "schema-name": "Test",  
                "table-name": "loadsecond"  
            },  
            "rule-action": "include",  
            "load-order": "1"  
        }  
    ]  
}
```

## Reglas y acciones de transformación

Puede utilizar las acciones de transformación para especificar las transformaciones que desea aplicar al esquema o la tabla seleccionados. Las reglas de transformación son opcionales.

Para las reglas de correspondencia de tablas utilice el tipo de regla de transformación; se pueden aplicar los valores siguientes.

Parámetro	Valores posibles	Descripción
<code>rule-type</code>	<code>transformation</code>	Un valor que aplica la regla a cada objeto especificado mediante la regla de selección. Utilice <code>transformation</code> a menos que se indique lo contrario.
<code>rule-id</code>	Un valor numérico.	Un único valor numérico para identificar la regla.
<code>rule-name</code>	Un valor alfanumérico.	Un nombre exclusivo para identificar la regla.
<code>object-locator</code>	Un objeto con los siguientes parámetros:	El nombre de cada esquema, tabla, espacio de tabla de tabla y espacio

Parámetro	Valores posibles	Descripción
	<ul style="list-style-type: none"> <li>• <code>schema-name</code>: el nombre del esquema.</li> <li>• <code>table-name</code>: el nombre de la tabla.</li> <li>• <code>table-tablespace-name</code>: el nombre de un espacio de tabla de una tabla existente.</li> <li>• <code>index-tablespace-name</code>: el nombre de un espacio de tabla de un índice existente.</li> </ul>	<p>de tabla de índice al que se aplica la regla. Puede utilizar el símbolo de porcentaje "%" como carácter comodín para la totalidad o parte del valor de cada parámetro <code>object-locator</code>. Así, puede hacer coincidir estos elementos:</p> <ul style="list-style-type: none"> <li>• Una sola tabla en un esquema único</li> <li>• Una sola tabla en varios o todos los esquemas</li> <li>• Algunas o todas las tablas en un esquema único</li> <li>• Algunas o todas las tablas en algunos o todos los esquemas</li> </ul> <p>Además, el parámetro <code>index-tablespace-name</code> o <code>table-tablespace-name</code> solo está disponible para que coincida con un punto de enlace de origen de Oracle. Puede especificar <code>table-tablespace-name</code> o bien <code>index-tablespace-name</code> en una regla única, pero no ambos. De este modo, puede asignar cualquiera de los elementos siguientes:</p> <ul style="list-style-type: none"> <li>• Uno, algunos o todos los espacios de tablas de tabla</li> <li>• Uno, algunos o todos los espacios de tablas de índice</li> </ul>
<code>rule-action</code>	<code>rename</code> <code>remove-column</code> <code>convert-lowercase, convert-uppercase</code> <code>add-prefix, remove-prefix, replace-prefix</code> <code>add-suffix, remove-suffix, replace-suffix</code> <code>define-primary-key</code>	La transformación que desea aplicar al objeto. Todas las acciones de las reglas de transformación distinguen entre mayúsculas y minúsculas.

Parámetro	Valores posibles	Descripción
rule-target	schema, table, column, table-tablespace, index-tablespace	El tipo de objeto que está transformando.  El valor table-tablespace o index-tablespace solo está disponible para un punto de enlace de destino de Oracle. Además, especifique el valor que coincide con el parámetro correspondiente que especifique como parte del object-locator: table-tablespace-name o index-tablespace-name.
value	Un valor alfanumérico que sigue las reglas de nomenclatura para el tipo de destino.	El nuevo valor para las acciones que requieren entradas, como rename.
old-value	Un valor alfanumérico que sigue las reglas de nomenclatura para el tipo de destino.	El valor antiguo para las acciones que requieran sustitución, como replace-prefix.
primary-key-def	Un objeto con los siguientes parámetros: <ul style="list-style-type: none"> <li>• name: el nombre de una nueva clave principal o índice único para la tabla.</li> <li>• (Opcional) origin: el tipo de clave única para definir: primary-key (la opción predeterminada) o unique-index.</li> <li>• columns: una matriz de cadenas que enumera los nombres de las columnas en el orden en el que aparecen en la clave principal o en el índice único.</li> </ul>	Con la acción de la regla establecida en define-primary-key y el destino de la regla establecido en table, este parámetro define el nombre, el tipo y el contenido de una clave única en la tabla transformada. De forma predeterminada, la clave única se define como una clave primaria.

#### Example Cambiar el nombre a un esquema

En el siguiente ejemplo, se cambia el nombre de un esquema de **Test** en el origen a **Test1** en el punto de enlace de destino.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "Test",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "transformation",
            "rule-id": "2",
            "rule-name": "2",
            "rule-action": "rename",
            "rule-expression": {
                "source": "schema-name",
                "target": "Test1"
            }
        }
    ]
}
```

```
        "rule-target": "schema",
        "object-locator": {
            "schema-name": "Test"
        },
        "value": "Test1"
    }
}
```

#### Example Cambiar el nombre a una tabla

En el siguiente ejemplo, se cambia el nombre de una tabla de **Actor** en el origen a **Actor1** en el punto de enlace de destino.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "Test",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "transformation",
            "rule-id": "2",
            "rule-name": "2",
            "rule-action": "rename",
            "rule-target": "table",
            "object-locator": {
                "schema-name": "Test",
                "table-name": "Actor"
            },
            "value": "Actor1"
        }
    ]
}
```

#### Example Cambiar el nombre de una columna

En el siguiente ejemplo, se cambia el nombre de una columna en una tabla **Actor** de **first\_name** en el origen a **fname** en el punto de enlace de destino.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "test",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "transformation",
            "rule-id": "4",
            "rule-name": "4",
            "rule-action": "rename",
            "rule-target": "column",
            "object-locator": {
                "schema-name": "test",
                "table-name": "Actor"
            },
            "value": "first_name"
        }
    ]
}
```

```
        "rule-target": "column",
        "object-locator": {
            "schema-name": "test",
            "table-name": "Actor",
            "column-name" : "first_name"
        },
        "value": "fname"
    }
]
```

#### Example Cambie el nombre de un espacio de tabla de tabla de Oracle

En el siguiente ejemplo se cambia el nombre del espacio de tabla de tabla denominado **SetSpace** para una tabla denominada **Actor** en su origen de Oracle a **SceneTblSpace** en su punto de enlace de destino de Oracle.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "Play",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "transformation",
            "rule-id": "2",
            "rule-name": "2",
            "rule-action": "rename",
            "rule-target": "table-tablespace",
            "object-locator": {
                "schema-name": "Play",
                "table-name": "Actor",
                "table-tablespace-name": "SetSpace"
            },
            "value": "SceneTblSpace"
        }
    ]
}
```

#### Example Cambie el nombre de un espacio de tabla de índice de Oracle

En el siguiente ejemplo se cambia el nombre del espacio de tabla de índice denominado **SetISpace** para una tabla denominada **Actor** en su origen de Oracle a **SceneIdxSpace** en su punto de enlace de destino de Oracle.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "Play",
                "table-name": "%"
            },
            "rule-action": "include"
        },

```

```
{  
    "rule-type": "transformation",  
    "rule-id": "2",  
    "rule-name": "2",  
    "rule-action": "rename",  
    "rule-target": "table-tablespace",  
    "object-locator": {  
        "schema-name": "Play",  
        "table-name": "Actor",  
        "table-tablespace-name: "SetISpace"  
    },  
    "value": "SceneIdxSpace"  
}  
]  
}
```

#### Example Quitar una columna

En el siguiente ejemplo, se transforma la tabla denominada **Actor** en el origen para que se quiten todas las columnas que empiecen por los caracteres **col** en el punto de enlace destino.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "1",  
            "rule-name": "1",  
            "object-locator": {  
                "schema-name": "test",  
                "table-name": "%"  
            },  
            "rule-action": "include"  
        },  
        {  
            "rule-type": "transformation",  
            "rule-id": "2",  
            "rule-name": "2",  
            "rule-action": "remove-column",  
            "rule-target": "column",  
            "object-locator": {  
                "schema-name": "test",  
                "table-name": "Actor",  
                "column-name": "col%"  
            }  
        }  
    ]  
}
```

#### Example Cambiar a minúsculas

En el siguiente ejemplo, se convierte el nombre de una tabla de **ACTOR** en el origen a **actor** en el punto de enlace de destino.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "1",  
            "rule-name": "1",  
            "object-locator": {  
                "schema-name": "test",  
                "table-name": "%"  
            },  
            "rule-action": "include"  
        },  
        {  
            "rule-type": "transformation",  
            "rule-id": "2",  
            "rule-name": "2",  
            "rule-action": "lowercase",  
            "rule-target": "table",  
            "object-locator": {  
                "schema-name": "test",  
                "table-name": "Actor"  
            }  
        }  
    ]  
}
```

```
"rule-name": "2",
"rule-action": "convert-lowercase",
"rule-target": "table",
"object-locator": {
  "schema-name": "test",
  "table-name": "ACTOR"
}
}]}
```

#### Example Cambiar a mayúsculas

En el siguiente ejemplo, se cambian todas las columnas de todas las tablas y todos los esquemas de minúsculas en el origen a mayúsculas en el punto de enlace de destino.

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "object-locator": {
        "schema-name": "test",
        "table-name": "%"
      },
      "rule-action": "include"
    },
    {
      "rule-type": "transformation",
      "rule-id": "2",
      "rule-name": "2",
      "rule-action": "convert-uppercase",
      "rule-target": "column",
      "object-locator": {
        "schema-name": "%",
        "table-name": "%",
        "column-name": "%"
      }
    }
  ]
}
```

#### Example Añadir un prefijo

En el siguiente ejemplo se transforman todas las tablas en el origen para añadirles el prefijo **DMS\_** en el punto de enlace de destino.

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "object-locator": {
        "schema-name": "test",
        "table-name": "%"
      },
      "rule-action": "include"
    },
    {
      "rule-type": "transformation",
      "rule-id": "2",
      "rule-name": "2",
      "rule-action": "add-prefix",
      "rule-target": "table",
      "prefix": "DMS_"
    }
  ]
}
```

```
    "object-locator": {  
        "schema-name": "test",  
        "table-name": "%"  
    },  
    "value": "DMS_"  
}  
}
```

#### Example Sustituir un prefijo

En el siguiente ejemplo, se transforman todas las columnas que contienen el prefijo **Pre\_** en el origen y se sustituye por **NewPre\_** en el punto de enlace de destino.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "1",  
            "rule-name": "1",  
            "object-locator": {  
                "schema-name": "test",  
                "table-name": "%"  
            },  
            "rule-action": "include"  
        },  
        {  
            "rule-type": "transformation",  
            "rule-id": "2",  
            "rule-name": "2",  
            "rule-action": "replace-prefix",  
            "rule-target": "column",  
            "object-locator": {  
                "schema-name": "%",  
                "table-name": "%",  
                "column-name": "%"  
            },  
            "value": "NewPre_",  
            "old-value": "Pre_"  
        }  
    ]  
}
```

#### Example Quitar un sufijo

En el siguiente ejemplo, se transforman todas las tablas en el origen para quitarles el sufijo **\_DMS** en el punto de enlace de destino.

```
{  
    "rules": [{  
        "rule-type": "selection",  
        "rule-id": "1",  
        "rule-name": "1",  
        "object-locator": {  
            "schema-name": "test",  
            "table-name": "%"  
        },  
        "rule-action": "include"  
    }, {  
        "rule-type": "transformation",  
        "rule-id": "2",  
        "rule-name": "2",  
        "rule-action": "remove-suffix",  
        "rule-target": "table"  
    }]  
}
```

```
"rule-target": "table",
"object-locator": {
  "schema-name": "test",
  "table-name": "%"
},
"value": "_DMS"
}]
```

#### Example Defina una clave principal

En el siguiente ejemplo se define una clave principal denominada `ITEM-primary-key` en tres columnas de la tabla `ITEM` migrada a su punto de enlace de destino.

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "object-locator": {
        "schema-name": "inventory",
        "table-name": "%"
      },
      "rule-action": "include"
    },
    {
      "rule-type": "transformation",
      "rule-id": "2",
      "rule-name": "2",
      "rule-action": "define-primary-key",
      "rule-target": "table",
      "object-locator": {
        "schema-name": "inventory",
        "table-name": "ITEM"
      },
      "primary-key-def": {
        "name": "ITEM-primary-key",
        "columns": [
          "ITEM-NAME",
          "BOM-MODEL-NUM",
          "BOM-PART-NUM"
        ]
      }
    }
  ]
}
```

#### Example Definir un índice único

En el siguiente ejemplo se define un índice único denominado `ITEM-unique-idx` en tres columnas de la tabla `ITEM` migrada a su punto de enlace de destino.

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "object-locator": {
        "schema-name": "inventory",
        "table-name": "%"
      },
      "rule-action": "include"
    },
    {
      "rule-type": "transformation",
      "rule-id": "2",
      "rule-name": "2",
      "rule-action": "define-unique-index",
      "rule-target": "table",
      "object-locator": {
        "schema-name": "inventory",
        "table-name": "ITEM"
      },
      "unique-index-def": {
        "name": "ITEM-unique-idx",
        "columns": [
          "ITEM-NAME",
          "BOM-MODEL-NUM",
          "BOM-PART-NUM"
        ]
      }
    }
  ]
}
```

```

    "rule-name": "2",
    "rule-action": "define-primary-key",
    "rule-target": "table",
    "object-locator": {
        "schema-name": "inventory",
        "table-name": "ITEM"
    },
    "primary-key-def": {
        "name": ITEM-unique-idx,
        "origin": unique-index,
        "columns": [
            "ITEM-NAME",
            "BOM-MODEL-NUM",
            "BOM-PART-NUM"
        ]
    }
}
]
}

```

## Reglas y operaciones de table-settings

Puede utilizar table-settings para especificar cualquier configuración que desee aplicar a la tabla seleccionada para una operación específica. Las reglas de table-settings son opcionales.

Para las reglas de mapeo de tablas que utilizan el tipo de regla table-settings, puede aplicar los parámetros siguientes.

Parámetro	Valores posibles	Descripción
<b>rule-type</b>	table-settings	Un valor que aplica la regla a una tabla especificada mediante la regla de selección.
<b>rule-id</b>	Un valor numérico.	Un único valor numérico para identificar la regla.
<b>rule-name</b>	Un valor alfanumérico.	Un nombre exclusivo para identificar la regla.
<b>object-locator</b>	Un objeto con los siguientes parámetros: <ul style="list-style-type: none"> <li><b>schema-name</b>: el nombre del esquema.</li> <li><b>table-name</b>: el nombre de la tabla.</li> </ul>	El nombre de un esquema y una tabla específicos (sin comodines).
<b>parallel-load</b>	Un objeto con los siguientes parámetros: <ul style="list-style-type: none"> <li><b>type</b>: especifica si la carga en paralelo está habilitada. Si lo está, este parámetro también especifica el mecanismo para identificar las particiones, subparticiones u otros segmentos de tabla que se van a cargar en paralelo. Las particiones son segmentos que ya están definidos e identificados por el nombre en la tabla de origen. Solo para los puntos de enlace de Oracle, las subparticiones son un nivel de</li> </ul>	Un valor que especifica una operación de carga en paralelo (varios subprocessos) en la tabla identificada por la opción <b>object-locator</b> . En este caso, puede cargar en paralelo en cualquiera de las siguientes formas: <ul style="list-style-type: none"> <li>Por segmentos especificados por todas las particiones o subparticiones disponibles.</li> <li>Por particiones y subparticiones seleccionadas.</li> <li>Por segmentos basados en rango que especifique.</li> </ul>

Parámetro	Valores posibles	Descripción
	<p>segmentos adicionales que ya están definidos e identificados por el nombre en la tabla de origen. Puede identificar otros segmentos en la regla table-settings especificando límites en el intervalo de valores para una o varias columnas de tabla.</p> <ul style="list-style-type: none"> <li>• <b>partitions:</b> cuando type es partitions-list, este valor especifica todas las particiones que se van a cargar en paralelo.</li> <li>• <b>subpartitions:</b> solo en el caso de los puntos de enlace de Oracle, cuando type es partitions-list, este valor especifica todas las subparticiones que se van a cargar en paralelo.</li> <li>• <b>columns:</b> cuando type es ranges, este valor especifica los nombres de columnas utilizadas para identificar segmentos de tabla basados en rango que se van a cargar en paralelo.</li> <li>• <b>boundaries:</b> cuando typees ranges, este valor especifica los valores de las columns utilizados para identificar segmentos basados en rango que se van a cargar en paralelo.</li> </ul>	<p>Para obtener más información sobre la carga en paralelo, consulte <a href="#">Uso de carga paralela para tablas seleccionadas (p. 305)</a>.</p>

Parámetro	Valores posibles	Descripción
<code>type</code>	<p>Una de las siguientes para <code>parallel-load</code>:</p> <ul style="list-style-type: none"> <li>• <code>partitions-auto</code>: todas las particiones de la tabla se cargan en paralelo. Cada partición se asigna a su propio subprocesso.</li> <li>• <code>subpartitions-auto</code>: (solo puntos de enlace de Oracle) Todas las subparticiones de la tabla se cargan en paralelo. Cada subpartición se asigna a su propio subprocesso.</li> <li>• <code>partitions-list</code>: todas las particiones especificadas de la tabla se cargan en paralelo. Solo para los puntos de enlace de Oracle, todas las subparticiones especificadas de la tabla se cargan en paralelo. Cada partición y subpartición que especifique se asigna a su propio subprocesso. Puede especificar las particiones y subparticiones que se cargan en paralelo por los nombres de particiones (<code>partitions</code>) y los nombres de subparticiones (<code>subpartitions</code>).</li> <li>• <code>ranges</code>: todos los segmentos especificados de la tabla se cargan en paralelo. Cada segmento de tabla que identifique se asigna a su propio subprocesso. Especifique estos segmentos por los nombres de las columnas (<code>columns</code>) y los valores de la columnas (<code>boundaries</code>).</li> <li>• <code>none</code>: la tabla se carga en una tarea de un solo subprocesso (predeterminada), con independencia de sus particiones o subparticiones. Para obtener más información, consulte <a href="#">Creación de una tarea (p. 254)</a>.</li> </ul>	Especifica el mecanismo para identificar las particiones, subparticiones o segmentos de tabla que se van a cargar en paralelo.
<code>partitions</code>	Cuando <code>type</code> es <code>partitions-list</code> , se trata de una serie de cadenas que especifican los nombres de las particiones que se cargan en paralelo.	Los nombres de las particiones que se van a cargar en paralelo.
<code>subpartitions</code>	(Solo puntos de enlace de Oracle) Cuando <code>type</code> es <code>partitions-list</code> , se trata de una matriz de cadenas que especifica los nombres de las subparticiones que se van a cargar en paralelo.	Los nombres de las subparticiones que se van a cargar en paralelo.

Parámetro	Valores posibles	Descripción
columns	Cuando type es ranges, se establece una matriz de cadenas en los nombres de columnas que identifican segmentos de tabla basados en rango que se cargan en paralelo.	Los nombres de las columnas utilizadas para identificar segmentos de tabla basados en rango que se van a cargar en paralelo.
boundaries	Cuando type es ranges, una matriz de matrices de valores de columna. Cada matriz de valores de columna contiene valores de columna en la cantidad y el orden especificados por columns. Una matriz de valores de columna especifica el límite superior de un segmento de tabla. Cada matriz de valores de columna adicional añade el límite superior de un segmento de tabla adicional. Todos estos segmentos de tabla basados en rango se cargan en paralelo.	Los valores de columna que identifican particiones de tabla basados en rango para cargar en paralelo.
lob-settings	Un objeto con los siguientes parámetros: <ul style="list-style-type: none"> <li>• mode: especifica el modo de gestionar la migración para los LOB.</li> <li>• bulk-max-size: especifica el tamaño máximo de los LOB, en función de la configuración de mode.</li> </ul>	Un valor que especifica la gestión de LOB para la tabla identificada por la opción object-locator. La gestión de LOB especificada anula cualquier configuración de tarea de LOB solo para esta tabla. Para obtener más información sobre el uso de los parámetros de configuración de LOB, consulte <a href="#">Especificación de la configuración de LOB para una tabla seleccionada (p. 308)</a> .

Parámetro	Valores posibles	Descripción
mode	<p>Especifica la gestión de migración para LOB en la tabla especificada utilizando los valores siguientes:</p> <ul style="list-style-type: none"> <li>• <b>limited</b> (predeterminado): este valor establece la migración en el modo LOB limitado, con todas las líneas de negocios migradas en línea junto con todos los demás tipos de datos de columna en la tabla. Utilice este valor al replicar LOB principalmente de tamaño pequeño (100 MB o menos). Además, especifique un valor <b>bulk-max-size</b> (cero no es válido). Todos los LOB migrados con un tamaño superior a <b>bulk-max-size</b> se truncan al tamaño que establezca.</li> <li>• <b>unlimited</b>: este valor establece la migración en modo de LOB completo. Utilice este valor cuando la totalidad o la mayor parte de los LOB que desea replicar tienen un tamaño superior a 1 GB. Si especifica un valor <b>bulk-max-size</b> de cero, todos los LOB se migran en modo de LOB completo estándar. En esta forma del modo <b>unlimited</b>, todos los LOB se migran aparte de otros tipos de datos de columna mediante una búsqueda desde la tabla de origen. Si especifica un valor <b>bulk-max-size</b> mayor que cero, todos los LOB se migran con el modo de LOB completo de combinación. En esta forma del modo <b>unlimited</b>, los LOB con un tamaño superior a <b>bulk-max-size</b> se migran utilizando una búsqueda de tabla de origen, similar al modo LOB completo estándar. De lo contrario, los LOB hasta este tamaño se migran en línea, similar al modo de LOB limitado. Ningún LOB se trunca en el modo <b>unlimited</b>, independientemente de la forma que utilice.</li> <li>• <b>none</b>: todos los LOB de tabla se migran de acuerdo a la configuración de LOB de tareas.</li> </ul> <p>Para obtener más información acerca de la configuración de LOB de tareas, consulte <a href="#">Configuración de las tareas de los metadatos de destino (p. 263)</a>.</p>	El mecanismo que se utiliza para migrar LOB.

Parámetro	Valores posibles	Descripción
	Para obtener más información acerca de cómo migrar LOB y cómo especificar esta configuración de LOB de tareas, consulte <a href="#">Establecimiento de compatibilidad con LOB para bases de datos de origen en la tarea de AWS DMS (p. 276)</a> .	
bulk-max-size	El efecto de este valor depende del mode.	El tamaño máximo de los LOB en incrementos de kilobytes. Especifique esta opción solo si necesita replicar LOB pequeños o si el punto de enlace de destino no admite el tamaño de LOB ilimitado.

## Uso de carga paralela para tablas seleccionadas

Para acelerar la migración y hacerla más eficiente, puede utilizar la carga paralela en tablas seleccionadas. En otras palabras, puede migrar una tabla de segmentación única con varios subprocessos en paralelo. Para ello, AWS DMS divide una tarea de carga completa en subprocessos y a cada segmento de la tabla se le asigna su propio subprocesso.

Con este proceso de carga en paralelo, primero puede hacer que varios subprocessos descarguen diversas tablas en paralelo desde el punto de enlace de origen. A continuación, puede hacer que varios subprocessos migren y carguen las mismas tablas en paralelo al punto de enlace de destino. Para algunos motores de base de datos, puede segmentar las tablas por las particiones o subparticiones existentes. De lo contrario, puede segmentar cualquier tabla por los rangos de valores de columna que especifique.

La carga en paralelo es compatible para los siguientes puntos de enlace de origen:

- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- IBM Db2 10.5
- SAP Adaptive Server Enterprise (ASE)

La carga en paralelo es compatible con los siguientes puntos de enlace de destino:

- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- Amazon Redshift
- SAP Adaptive Server Enterprise (ASE)

Especifique el número de subprocessos para una tarea de carga completa mediante la configuración de la tarea `ParallelLoadThreads`. Especifique el tamaño del búfer para una tarea de carga paralela mediante la configuración de la tarea `ParallelLoadBufferSize`. Esta configuración podría variar

en función del punto de enlace. Para obtener más información sobre esta configuración, consulte [Configuración de las tareas de los metadatos de destino \(p. 263\)](#).

Para utilizar la carga en paralelo, cree una regla de mapeo de tablas de tipo `table-settings` con la opción `parallel-load`. Dentro de la regla `table-settings`, puede especificar los criterios de segmentación para una única tabla que desea cargar en paralelo. Para ello, establezca el parámetro `type` de la opción `parallel-load` en una de las diversas opciones, en función de cómo desea segmentar la tabla para carga en paralelo:

- Por particiones: cargue todas las particiones de tabla existentes con el tipo `partitions-auto`. También puede cargar solamente las particiones seleccionadas con el tipo `partitions-list` y una matriz `partitions` especificada.
- (Solo los puntos de enlace de Oracle) Por subparticiones: cargue todas las subparticiones de tablas existentes con el tipo `subpartitions-auto`. También puede cargar solamente las subparticiones seleccionadas con el tipo `partitions-list` y una matriz `subpartitions` especificada.
- Por segmentos que defina: cargue los segmentos de tabla que defina mediante límites de valor de columna. Para ello, utilice el tipo `ranges` con las matrices `columns` y `boundaries` especificadas.

Para identificar otras tablas que desea cargar en paralelo, especifique objetos `table-settings` adicionales con las opciones `parallel-load`.

En los siguientes procedimientos, puede ver cómo programar el JSON para cada tipo de carga en paralelo, desde el más simple al más complejo.

Para especificar todas las particiones de tabla o todas las subparticiones de tabla

- Especifique `parallel-load` con el tipo `partitions-auto` o con el tipo `subpartitions-auto` (pero no ambos).

Cada partición o subpartición de tabla se asigna automáticamente a su propio subproceso.

**Note**

La carga en paralelo incluye particiones o subparticiones solo si ya están definidas para la tabla.

Para especificar particiones de tabla, subparticiones o ambas

1. Especifique `parallel-load` con el tipo `partitions-list`.
2. (Opcional) Para incluir particiones, especifique una matriz de nombres de partición como valor de `partitions`.

Cada partición especificada se asigna a continuación a su propio subproceso.

3. (Opcional, solo para puntos de enlace de Oracle) Para incluir subparticiones, especifique una matriz de nombres de subpartición como valor de `subpartitions`.

Cada subpartición especificada se asigna a continuación a su propio subproceso.

**Note**

La carga en paralelo incluye particiones o subparticiones solo si ya están definidas para la tabla.

Puede especificar segmentos de tabla como rangos de valores de columna Al hacerlo, tenga en cuenta estas características de las columnas:

- La especificación de columnas indexadas mejora de manera significativa el rendimiento.

- Puede especificar hasta 10 columnas.
- No puede utilizar columnas para definir límites de segmentos con los siguientes tipos de datos de AWS DMS: DOUBLE, FLOAT, BLOB, CLOB y NCLOB.
- Los registros con valores nulos no se replican.

Para especificar segmentos de tabla como rangos de valores de columna

1. Especifique `parallel-load` con el tipo `ranges`.
2. Defina un límite entre los segmentos de tabla; para ello, especifique una matriz de nombres de columnas de tabla como valor de `columns`. Repita esta operación con cada columna para la que desea definir un límite entre segmentos de tabla.

El orden de las columnas es importante. La primera columna es la más importante y la última es la menos importante para definir cada límite, tal como se describe a continuación.

3. Para definir los rangos de datos de todos los segmentos de la tabla, especifique una matriz de límites como valor de `boundaries`. Una matriz de límite es una matriz de matrices columna-valor. Para ello, siga estos pasos:
  - a. Especifique cada elemento de una matriz de valores de columna como un valor que corresponde a cada columna. Una matriz columna-valor representa el límite superior de cada segmento de tabla que desea definir. Especifique cada columna en el mismo orden en el que la especificó en la matriz `columns`.

**Note**

Introduzca valores para las columnas DATE en el formato admitido por el origen.

- b. Especifique cada matriz columna-valor como límite superior, en orden, de cada segmento desde la parte inferior hasta el segmento que esté situado junto al superior de la tabla. Si existe alguna fila de tabla sobre el límite superior que especifique, estas filas completan el segmento superior de la tabla. De este modo, el número de segmentos basados en rango es potencialmente uno más que el número de límites de segmentos en la matriz de límites. Cada uno de estos segmentos basados en rango se asigna a su propio subproceso.

**Note**

Todos los datos de la tabla no nulos se replican, incluso si no define rangos de datos para todas las columnas de la tabla.

Por ejemplo, suponga que define tres matrices columna-valor para las columnas COL1 COL2 y COL3, tal y como se indica a continuación.

COL1	COL2	COL3
10	30	105
20	20	120
100	12	99

Ha definido tres límites de segmentos para un total posible de cuatro segmentos.

Para identificar los rangos de filas que se van a replicar para cada segmento, la instancia de replicación aplica una búsqueda como la siguiente a estas tres columnas para cada uno de los cuatro segmentos:

#### Segmento 1

Replique todas las filas donde los valores de las dos primeras columnas sean iguales o inferiores a los valores de límite superior Segmento 1 correspondientes y los valores de la tercera columna sean inferiores a su valor de límite superior Segmento 1.

#### Segmento 2

Replique todas las filas (excepto las filas Segmento 1) donde los valores de las dos primeras columnas sean menores o iguales que los valores del límite superior de Segmento 2 correspondientes y los valores de la tercera columna sean inferiores al valor del límite superior de Segmento 2.

#### Segmento 3

Replique todas las filas (excepto las filas Segmento 2) donde los valores de las dos primeras columnas sean menores o iguales que los valores del límite superior de Segmento 3 correspondientes y los valores de la tercera columna sean inferiores al valor del límite superior de Segmento 3.

#### Segmento 4

Replique todas las filas restantes (excepto las filas Segmento 1, 2 y 3).

En este caso, la instancia de replicación crea una cláusula WHERE para cargar cada segmento tal y como se indica a continuación:

#### Segmento 1

```
((COL1 < 10) OR ((COL1 = 10) AND (COL2 < 30)) OR ((COL1 = 10) AND (COL2 = 30) AND (COL3 < 105)))
```

#### Segmento 2

```
NOT ((COL1 < 10) OR ((COL1 = 10) AND (COL2 < 30)) OR ((COL1 = 10) AND (COL2 = 30) AND (COL3 < 105))) AND ((COL1 < 20) OR ((COL1 = 20) AND (COL2 < 20)) OR ((COL1 = 20) AND (COL2 = 20) AND (COL3 < 120)))
```

#### Segmento 3

```
NOT ((COL1 < 20) OR ((COL1 = 20) AND (COL2 < 20)) OR ((COL1 = 20) AND (COL2 = 20) AND (COL3 < 120))) AND ((COL1 < 100) OR ((COL1 = 100) AND (COL2 < 12)) OR ((COL1 = 100) AND (COL2 = 12) AND (COL3 < 99)))
```

#### Segmento 4

```
NOT ((COL1 < 100) OR ((COL1 = 100) AND (COL2 < 12)) OR ((COL1 = 100) AND (COL2 = 12) AND (COL3 < 99)))
```

## Especificación de la configuración de LOB para una tabla seleccionada

Si desea establecer la configuración de las tareas de LOB para una o más tablas, cree una regla de mapeo de tablas de tipo table-settings con la opción lob-settings para uno o varios objetos table-settings.

La especificación de la configuración de LOB para tablas seleccionadas se admite para los siguientes puntos de enlace de origen:

- Oracle

- Microsoft SQL Server
- MySQL
- PostgreSQL
- IBM Db2, en función de los ajustes `mode` y `bulk-max-size`, que se describen a continuación
- SAP Adaptive Server Enterprise (ASE), en función de los ajustes `mode` y `bulk-max-size`, que se describen a continuación

La especificación de la configuración de LOB para tablas seleccionadas se admite para los siguientes puntos de enlace de destino:

- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- SAP ASE, en función de los ajustes `mode` y `bulk-max-size`, que se describen a continuación

#### Note

Solo puede utilizar tipos de datos BLOB en tablas que incluyan una clave principal.

Para utilizar la configuración de LOB para una tabla seleccionada, debe crear una regla de mapeo de tablas de tipo `table-settings` con la opción `lob-settings`. Esto especifica la gestión de LOB para la tabla identificada por la opción `object-locator`. En la regla `table-settings`, puede especificar un objeto `lob-settings` con los parámetros siguientes:

- `mode`: especifica el mecanismo para la gestión de LOB para la tabla seleccionada tal y como se indica a continuación:
  - `limited`: el modo de LOB limitado predeterminado es el modo más rápido y eficiente. Utilice este modo solo si todos los LOB son pequeños (hasta 100 MB de tamaño) o el punto de enlace de destino no es compatible con un tamaño de LOB ilimitado. Además, si utiliza `limited` todos los LOB deben respetar el tamaño que establezca para `bulk-max-size`.

En este modo para una tarea de carga completa, la instancia de replicación migra todos los LOB en línea junto con otros tipos de datos de columna como parte del almacenamiento de tabla principal. Sin embargo, la instancia trunca al tamaño especificado cualquier LOB migrado con un tamaño superior al valor `bulk-max-size`. Para una tarea de carga de captura de datos de cambios (CDC), la instancia migra todos los LOB utilizando una búsqueda de tabla de origen, como en modo LOB completo estándar (consulte a continuación). Lo hace con independencia del tamaño de LOB.

- `unlimited`: el mecanismo de migración para el modo de LOB completo depende del valor que defina para `bulk-max-size` del modo siguiente:
  - Modo de LOB completo estándar: cuando se establece `bulk-max-size` en cero, la instancia de replicación migra todos los LOB utilizando el modo de LOB completo estándar. Este modo requiere una búsqueda en la tabla de origen para migrar cada LOB, con independencia del tamaño. Este enfoque suele dar como resultado una migración mucho más lenta que para el modo de LOB limitado. Utilice este modo solo si todos o la mayor parte de los LOB son grandes (1 GB o más).
  - Modo de LOB completo de combinación: cuando se establece `bulk-max-size` en un valor distinto de cero, este modo de LOB completo utiliza una combinación de modo de LOB limitado y modo de LOB completo estándar. Es decir, para una tarea de carga completa, si el tamaño de LOB está dentro del valor `bulk-max-size`, la instancia migra el LOB en línea como en el modo de LOB limitado. Si el tamaño de LOB es superior a ese valor, la instancia migra el LOB mediante una búsqueda de tabla de origen como en el modo de LOB completo estándar. Para una tarea de carga de captura de datos de cambios (CDC), la instancia migra todos los LOB utilizando una búsqueda

de tabla de origen, como en modo LOB completo estándar (consulte a continuación). Lo hace con independencia del tamaño de LOB.

Este modo da lugar a una velocidad de migración que es un compromiso entre el modo de LOB limitado, más rápido y el modo de LOB completo estándar, más lento. Utilice este modo solo cuando tenga una combinación de LOB pequeños y grandes y la mayoría de los LOB sean pequeños.

Este modo LOB completo de combinación solo está disponible para los siguientes puntos de enlace:

- IBM Db2 como origen
- SAP ASE como origen o destino

Independientemente del mecanismo que especifique para el modo `unlimited`, la instancia migra todos los LOB por completo, sin truncado.

- `none`: la instancia de replicación migra los LOB en la tabla seleccionada utilizando la configuración de LOB de la tarea. Utilice esta opción para ayudar a comparar los resultados de migración con y sin la configuración de LOB para la tabla seleccionada.

Si la tabla especificada tiene LOB incluidos en la replicación, puede establecer la opción de tarea `BatchApplyEnabled` en `true` solo al utilizar el modo de LOB `limited`.

En algunos casos, puede establecer `BatchApplyEnabled` en `true` y `BatchApplyPreserveTransaction` en `false`. En estos casos, la instancia establece `BatchApplyPreserveTransaction` en `true` si la tabla tiene LOB y los puntos de enlace de origen y destino son Oracle.

- `bulk-max-size`: establezca este valor en un valor cero o no cero en kilobytes, en función del `mode`, tal como se ha descrito para los elementos anteriores. En modo `limited`, debe establecer un valor distinto de cero para este parámetro.

La instancia convierte los LOB a formato binario. Por lo tanto, para especificar el LOB mayor que tiene que replicar, multiplique su tamaño por tres. Por ejemplo, si el LOB más grande tiene 2 MB, defina `bulk-max-size` en 6000 (6 MB).

## Ejemplos de table-settings

A continuación, encontrará algunos ejemplos que muestran el uso de la configuración de tabla.

Example Cargue una tabla segmentada por particiones

En el siguiente ejemplo se carga una tabla **SALES** en el origen de manera más eficiente cargándola en paralelo en función de todas sus particiones.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "1",  
            "rule-name": "1",  
            "object-locator": {  
                "schema-name": "%",  
                "table-name": "%"   
            },  
            "rule-action": "include"  
        },  
        {  
            "rule-type": "table-settings",  
            "rule-id": "2",  
            "rule-name": "2",  
            "object-locator": {  
                "schema-name": "HR",  
                "table-name": "SALES"  
            },  
            "rule-action": "include"  
        }  
    ]  
}
```

```

        "table-name": "SALES"
    },
    "parallel-load": {
        "type": "partitions-auto"
    }
}
]
}
}
```

#### Example Cargar una tabla segmentada por subparticiones

En el siguiente ejemplo se carga una tabla **SALES** en el origen de Oracle de manera más eficiente cargándola en paralelo en función de todas sus subparticiones.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "%",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "table-settings",
            "rule-id": "2",
            "rule-name": "2",
            "object-locator": {
                "schema-name": "HR",
                "table-name": "SALES"
            },
            "parallel-load": {
                "type": "subpartitions-auto"
            }
        }
    ]
}
```

#### Example Cargar una tabla segmentada por una lista de particiones

En el siguiente ejemplo se carga una tabla **SALES** en su origen cargándola en paralelo por medio de una lista de particiones particular. Aquí, las particiones especificadas se nombran según valores que empiezan por partes del alfabeto, por ejemplo, **ABCD**, **EFGH**, etc.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "%",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "table-settings",
            "rule-id": "2",
            "rule-name": "2",
            "object-locator": {
                "partition-list": [
                    "ABCD", "EFGH"
                ],
                "schema-name": "%",
                "table-name": "%"
            },
            "parallel-load": {
                "type": "list"
            }
        }
    ]
}
```

```

        "schema-name": "HR",
        "table-name": "SALES"
    },
    "parallel-load": {
        "type": "partitions-list",
        "partitions": [
            "ABCD",
            "EFGH",
            "IJKL",
            "MNOP",
            "QRST",
            "UVWXYZ"
        ]
    }
}
]
}
}

```

Example Cargue una tabla de Oracle segmentada por una lista de particiones y subparticiones seleccionada

En el siguiente ejemplo se carga una tabla **SALES** en su origen de Oracle cargándola en paralelo por una lista de particiones y subparticiones seleccionada. Aquí, las particiones especificadas se nombran según valores que empiezan por partes del alfabeto, por ejemplo, **ABCD**, **EFGH**, etc. Las subparticiones especificadas se nombran según valores que empiezan por numerales, por ejemplo, **01234** y **56789**.

```

{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "%",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "table-settings",
            "rule-id": "2",
            "rule-name": "2",
            "object-locator": {
                "schema-name": "HR",
                "table-name": "SALES"
            },
            "parallel-load": {
                "type": "partitions-list",
                "partitions": [
                    "ABCD",
                    "EFGH",
                    "IJKL",
                    "MNOP",
                    "QRST",
                    "UVWXYZ"
                ],
                "subpartitions": [
                    "01234",
                    "56789"
                ]
            }
        }
    ]
}

```

### Example Cargar una tabla segmentada por rangos de valores de columna

En el siguiente ejemplo se carga una tabla **SALES** en el origen cargándola en paralelo por segmentos especificados por los rangos de los valores de columna **SALES\_NO** y **REGION**.

```
{  
    "rules": [ {  
        "rule-type": "selection",  
        "rule-id": "1",  
        "rule-name": "1",  
        "object-locator": {  
            "schema-name": "%",  
            "table-name": "%" }  
        },  
        { "rule-action": "include" }  
    },  
    {  
        "rule-type": "table-settings",  
        "rule-id": "2",  
        "rule-name": "2",  
        "object-locator": {  
            "schema-name": "HR",  
            "table-name": "SALES" }  
        },  
        "parallel-load": {  
            "type": "ranges",  
            "columns": [ "SALES_NO",  
                "REGION" ]  
            ],  
            "boundaries": [ [ [ "1000",  
                "NORTH" ]  
                ],  
                [ [ "3000",  
                    "WEST" ]  
                ]  
            ]  
        }  
    ]  
}
```

Aquí, se especifican dos columnas por rangos de segmento con los nombres **SALES\_NO** y **REGION**.

Se especifican dos límites con dos conjuntos de valores de columna (**["1000", "NORTH"]** y **["3000", "WEST"]**).

Estos dos límites, por lo tanto, identifican los tres segmentos de tabla siguientes para cargarlos en paralelo:

#### Segmento 1

Las filas con **SALES\_NO** menor o igual que 1000 y **REGION** menor que "NORTH". En otras palabras, cifras de ventas de hasta 1000 en la región EAST.

#### Segmento 2

Las filas distintas de Segmento 1 con **SALES\_NO** menor o igual que 3000 y **REGION** menor que "WEST". En otras palabras, las cifras de ventas de 1000 a 3000 en las regiones NORTH y SOUTH.

#### Segmento 3

Todas las filas restantes que no sean Segment 1 y Segment 2. En otras palabras, los números de ventas superiores a 3000 en la región WEST.

### Example Carga dos tablas: una segmentada por rangos y otra segmentada por particiones

En el siguiente ejemplo se carga una tabla **SALES** en paralelo por los límites de segmento que identifique. También se carga una tabla **ORDERS** en paralelo por todas sus particiones, como en ejemplos anteriores.

```
{  
    "rules": [  
        {  
            "rule-type": "selection",  
            "rule-id": "1",  
            "rule-name": "1",  
            "object-locator": {  
                "schema-name": "%",  
                "table-name": "%"   
            },  
            "rule-action": "include"  
        },  
        {  
            "rule-type": "table-settings",  
            "rule-id": "2",  
            "rule-name": "2",  
            "object-locator": {  
                "schema-name": "HR",  
                "table-name": "SALES"  
            },  
            "parallel-load": {  
                "type": "ranges",  
                "columns": [  
                    "SALES_NO",  
                    "REGION"  
                ],  
                "boundaries": [  
                    [  
                        "1000",  
                        "NORTH"  
                    ],  
                    [  
                        "3000",  
                        "WEST"  
                    ]  
                ]  
            }  
        },  
        {  
            "rule-type": "table-settings",  
            "rule-id": "3",  
            "rule-name": "3",  
            "object-locator": {  
                "schema-name": "HR",  
                "table-name": "ORDERS"  
            },  
            "parallel-load": {  
                "type": "partitions-auto"  
            }  
        }  
    ]  
}
```

### Example Cargar una tabla con LOB utilizando el modo de LOB limitado

En el siguiente ejemplo se carga una tabla **ITEMS** que incluye LOB en el origen utilizando el modo de LOB limitado (el predeterminado) con un tamaño máximo no truncado de 100 MB. Cualquier LOB que tenga un tamaño superior a este se trunca a 100 MB. Todos los LOB se cargan en línea con todos los demás tipos de datos de columna

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "%",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "table-settings",
            "rule-id": "2",
            "rule-name": "2",
            "object-locator": {
                "schema-name": "INV",
                "table-name": "ITEMS"
            },
            "lob-settings": {
                "bulk-max-size": "100000"
            }
        }
    ]
}
```

#### Example Cargar una tabla con LOB utilizando el modo de LOB completo estándar

En el siguiente ejemplo se carga una tabla **ITEMS** en su origen, incluidos todos sus LOB sin truncar, utilizando el modo de LOB completo estándar. Todos LOB, con independencia de su tamaño, se cargan por separado de otros tipos de datos mediante una búsqueda para cada LOB en la tabla de origen.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "%",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "table-settings",
            "rule-id": "2",
            "rule-name": "2",
            "object-locator": {
                "schema-name": "INV",
                "table-name": "ITEMS"
            },
            "lob-settings": {
                "mode": "unlimited"
                "bulk-max-size": "0"
            }
        }
    ]
}
```

#### Example Cargar una tabla con LOB utilizando el modo de LOB completo de combinación

En el siguiente ejemplo se carga una tabla **ITEMS** en su origen, incluidos todos sus LOB sin truncar, utilizando el modo de LOB completo de combinación. Todos los LOB de hasta 100 MB de tamaño se

cargan en línea junto con otros tipos de datos, como en el modo LOB limitado. Todos los LOB con más de 100 MB de tamaño se cargan por separado de otros tipos de datos utilizando una búsqueda para cada uno de estos LOB en la tabla de origen, como en modo de LOB completo estándar.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "%",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "table-settings",
            "rule-id": "2",
            "rule-name": "2",
            "object-locator": {
                "schema-name": "INV",
                "table-name": "ITEMS"
            },
            "lob-settings": {
                "mode": "unlimited"
                "bulk-max-size": "100000"
            }
        }
    ]
}
```

#### Example Cargar una tabla con LOB utilizando la configuración de LOB de tarea

En el ejemplo siguiente se carga una tabla **ITEMS** en el origen, incluidos todos los LOB, utilizando su configuración de LOB de tarea. La configuración de **bulk-max-size** de 100 MB se omite y se deja solo para un restablecimiento rápido al modo **limited** o **unlimited**.

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "%",
                "table-name": "%"
            },
            "rule-action": "include"
        },
        {
            "rule-type": "table-settings",
            "rule-id": "2",
            "rule-name": "2",
            "object-locator": {
                "schema-name": "INV",
                "table-name": "ITEMS"
            },
            "lob-settings": {
                "mode": "none"
                "bulk-max-size": "100000"
            }
        }
    ]
}
```

## Usar filtros de origen

Puede utilizar filtros de origen para limitar el número y el tipo de los registros transferidos desde el origen al destino. Por ejemplo, puede especificar que solo los trabajadores con una ubicación de sede central se trasladen a la base de datos de destino. Los filtros forman parte de una regla de selección. Puede aplicar filtros en una columna de datos.

Los filtros de origen presentan las restricciones siguientes:

- Una regla de selección puede no tener filtros o bien uno o más filtros.
- Cada filtro puede tener una o varias condiciones de filtro.
- Si se utiliza más de un filtro, la lista de filtros se combina como si utilizara un operador AND entre los filtros.
- Si se utiliza más de una condición de filtro en un único filtro, la lista de condiciones del filtro se combina como si se utilizara un operador OR entre las condiciones del filtro.
- Los filtros solo se aplican cuando `rule-action = 'include'`.
- Los filtros requieren un nombre de columna y una lista de condiciones de filtro. Las condiciones del filtro deben tener un operador y un valor de filtro.
- Los nombres de columna, los nombres de tabla y los nombres de esquema distinguen entre mayúsculas y minúsculas.

## Crear reglas de filtros de origen en JSON

Puede crear filtros de origen mediante el parámetro JSON `filters` de una regla de selección. El parámetro `filters` especifica una matriz de uno o varios objetos JSON. Cada objeto tiene parámetros que especifican el tipo de filtro de origen, el nombre de columna y las condiciones de filtro. Estas condiciones de filtro incluyen uno o varios operadores de filtro y valores de filtro.

La tabla siguiente muestra los parámetros para especificar el filtrado de origen en un objeto `filters`.

Parámetro	Valor
<code>filter-type</code>	<code>source</code>
<code>column-name</code>	El nombre de la columna de origen a la que desea aplicar el filtro. El nombre distingue entre mayúsculas y minúsculas.
<code>filter-conditions</code>	Una matriz de uno o más objetos que contiene un parámetro <code>filter-operator</code> y un parámetro de valor apropiado en función del valor <code>filter-operator</code> .
<code>filter-operator</code>	Este parámetro puede tener uno de los siguientes valores: <ul style="list-style-type: none"><li>• <code>lt</code>: menor que</li><li>• <code>lte</code>: menor o igual a</li><li>• <code>gt</code>: mayor que</li><li>• <code>gte</code>: mayor o igual que</li><li>• <code>eq</code>: igual que</li><li>• <code>between</code>: igual que o entre dos valores</li></ul>
<code>value</code> o <code>start-value</code> <code>end-value</code>	El valor del parámetro <code>filter-operator</code> . Si el <code>filter-operator</code> tiene un valor distinto de <code>between</code> , utilice <code>value</code> . Si el <code>filter-operator</code> se ha establecido en <code>between</code> , proporcione dos valores, uno para <code>start-value</code> y uno para <code>end-value</code> .

Los siguientes ejemplos muestran algunas de las formas más habituales de utilizar filtros de origen.

#### Example Filtro único

El siguiente filtro replica todos los empleados que `empid >= 100` en la base de datos de destino.

```
{  
    "rules": [  
        {"rule-type": "selection",  
         "rule-id": "1",  
         "rule-name": "1",  
         "object-locator": {  
             "schema-name": "test",  
             "table-name": "employee"  
         },  
         "rule-action": "include",  
         "filters": [  
             {"filter-type": "source",  
              "column-name": "empid",  
              "filter-conditions": [  
                  {"filter-operator": "gte",  
                   "value": "100"}  
              ]  
         ]  
     ]  
}
```

#### Example Operadores de múltiples filtros

El siguiente filtro se aplica a operadores de múltiples filtros en una única columna de datos. El filtro replica todos los empleados que (`empid <=10`) O (`empid is between 50 and 75`) O (`empid >= 100`) en la base de datos de destino.

```
{  
    "rules": [  
        {"rule-type": "selection",  
         "rule-id": "1",  
         "rule-name": "1",  
         "object-locator": {  
             "schema-name": "test",  
             "table-name": "employee"  
         },  
         "rule-action": "include",  
         "filters": [  
             {"filter-type": "source",  
              "column-name": "empid",  
              "filter-conditions": [  
                  {"filter-operator": "lte",  
                   "value": "10"},  
                  {"filter-operator": "between",  
                   "start-value": "50",  
                   "end-value": "75"},  
                  {"filter-operator": "gte",  
                   "value": "100"}  
              ]  
         ]  
     ]  
}
```

### Example Múltiples filtros

El siguiente filtro se aplica a varios filtros en dos columnas de una tabla. El filtro replica todos los empleados que (`empid <= 100`) Y (`dept= tech`) en la base de datos de destino.

```
{  
  "rules": [  
    {  
      "rule-type": "selection",  
      "rule-id": "1",  
      "rule-name": "1",  
      "object-locator": {  
        "schema-name": "test",  
        "table-name": "employee"  
      },  
      "rule-action": "include",  
      "filters": [  
        {  
          "filter-type": "source",  
          "column-name": "empid",  
          "filter-conditions": [  
            {  
              "filter-operator": "ste",  
              "value": "100"  
            }  
          ]  
        },  
        {  
          "filter-type": "source",  
          "column-name": "dept",  
          "filter-conditions": [  
            {  
              "filter-operator": "eq",  
              "value": "tech"  
            }  
          ]  
        }  
      ]  
    }  
  ]  
}
```

## Filtrar por fecha y hora

Cuando se seleccionan datos para su importación, puede especificar una fecha u hora como parte de los criterios de filtrado. AWS DMS utiliza la fecha en formato AAAA-MM-DD y la hora en formato AAAA-MM-DD HH: MM: SS para el filtrado. Las funciones de comparación de AWS DMS siguen las convenciones de SQLite. Para obtener más información sobre los tipos de datos y las comparaciones de fechas de SQLite, consulte la sección sobre [tipos de datos en SQLite, versión 3](#) en la documentación de SQLite.

En el siguiente ejemplo, se muestra cómo filtrar según una fecha. Replica todos los empleados donde `empstartdate >= January 1, 2002` en la base de datos de destino.

### Example Filtro de una sola fecha

```
{  
  "rules": [  
    {  
      "rule-type": "selection",  
      "rule-id": "1",  
      "rule-name": "1",  
      "object-locator": {  
        "schema-name": "test",  
        "table-name": "employee"  
      },  
      "rule-action": "include",  
      "filters": [  
        {  
          "filter-type": "source",  
          "column-name": "empstartdate",  
          "filter-conditions": [  
            {  
              "filter-operator": "ge",  
              "value": "2002-01-01"  
            }  
          ]  
        }  
      ]  
    }  
  ]  
}
```

```
"filters": [{  
    "filter-type": "source",  
    "column-name": "empstartdate",  
    "filter-conditions": [{  
        "filter-operator": "gte",  
        "value": "2002-01-01"  
    }]  
}]  
}]  
}
```

# Monitorización de tareas de AWS DMS

Puede monitorizar el progreso de su tarea comprobando su estado y monitorizando su tabla de control. Para obtener más información sobre las tablas de control, consulte [Configuración de las tareas de la tabla de control \(p. 266\)](#).

También puede monitorizar el avance de las tareas con Amazon CloudWatch. Mediante la consola de administración de AWS, la interfaz de línea de comandos de AWS (CLI) o la API de AWS DMS puede monitorizar el progreso de la tarea y los recursos y la conectividad de red utilizados.

Por último, puede monitorizar el estado de las tablas de origen en una tarea visualizando el estado de la tabla.

Tenga en cuenta que la columna de última actualización de la consola de DMS solo indica la hora en que AWS DMS realizó la última actualización del registro de estadísticas de una tabla. No indica la hora de la última actualización de la tabla.

Para obtener más información, consulte los siguientes temas.

## Temas

- [Estado de una tarea \(p. 321\)](#)
- [Estado de la tabla durante las tareas \(p. 322\)](#)
- [Uso de Amazon CloudWatch para la monitorización de tareas de replicación \(p. 323\)](#)
- [Métricas del servicio de migración de datos \(p. 325\)](#)
- [Administración de logs de tareas de AWS DMS \(p. 328\)](#)
- [Registro de llamadas al API de AWS DMS con AWS CloudTrail \(p. 329\)](#)

## Estado de una tarea

El estado de una tarea indica su situación. En la siguiente tabla se muestran los posibles estados que puede tener una tarea:

Estado de una tarea	Descripción
Creando	AWS DMS está creando la tarea.
En ejecución	La tarea está realizando los trabajos de migración especificados.
Stopped	La tarea se ha interrumpido.
Stopping	La tarea se está deteniendo. Suele reflejar la intervención del usuario en la tarea.
Eliminando	La tarea se está eliminando, normalmente por una solicitud de intervención por parte del usuario.

Estado de una tarea	Descripción
Failed	La tarea ha fallado. Consulte los archivos de registro de la tarea para obtener más información.
Starting	La tarea se está conectando a la instancia de replicación y a los puntos de enlace de origen y destino. Se están aplicando todos los filtros y las transformaciones.
Ready	La tarea está lista para ejecutarse. Este estado normalmente sigue al estado de la creación de la tarea.
Modificando	La tarea se está modificando; en general, debido a la acción de un usuario que ha modificado la configuración de la tarea.

La barra de estado de la tarea proporciona una estimación del avance de la tarea. La calidad de esta estimación depende de la calidad de las estadísticas de la tabla de la base de datos de origen; cuanto mejores sean las estadísticas de la tabla, más precisa será la estimación. Si una tarea solo tiene una tabla que carece de estimación de estadísticas de filas, no podemos proporcionar ningún tipo de cálculo sobre el porcentaje completado. En este caso, puede utilizar el estado de la tarea y la indicación de las filas cargadas para confirmar que la tarea está en ejecución y avanzando.

## Estado de la tabla durante las tareas

La consola de AWS DMS actualiza información sobre el estado de las tablas durante la migración. En la siguiente tabla se muestran los posibles valores de estado:

The screenshot shows the AWS DMS Task Monitoring interface. At the top, there are buttons for 'Create task', 'Start/Resume', 'Stop', 'Delete', and a refresh icon. Below this is a filter bar with a search icon and an 'X' button. The main table has columns: ID, Status, Source, Target, and Type. Two rows are listed: 'sg-mysql2pg-task1' (Status: Load complete, Source: sg-mysql2pg-so, Target: sg-mysql2pg-ta, Type: Full Load) and 'sg-mysql2pg-task2' (Status: Load complete, Source: sg-mysql2pg-so, Target: sg-mysql2pg-ta, Type: Full Load). Below the table, the task name 'sg-mysql2pg-task1' is displayed. At the bottom, there are tabs: 'Overview', 'Task monitoring' (selected), 'Table statistics' (circled in red), and 'Logs'. In the 'Table statistics' section, there is another filter bar. The table below has columns: Schema, Table, State (circled in red), Inserts, Deletes, Updates, and Deletes. Three rows are shown: 'entertainmentAgencyModify Agents' (State: Table completed), 'entertainmentAgencyModify Customers' (State: Table completed), and 'entertainmentAgencyModify Engagements' (State: Table completed).

Schema	Table	State	Inserts	Deletes	Updates	Deletes
entertainmentAgencyModify	Agents	Table completed	0	0	0	0
entertainmentAgencyModify	Customers	Table completed	0	0	0	0
entertainmentAgencyModify	Engagements	Table completed	0	0	0	0

Estado	Descripción
Table does not exist (La tabla no existe)	AWS DMS no puede encontrar la tabla en el punto de enlace de origen.
Before load (Antes de cargar)	El proceso de carga completa se ha habilitado, pero aún no se ha iniciado.
Full load	El proceso de carga completa está en curso.
Table completed	Se ha terminado la carga completa.
Table cancelled	Se ha cancelado la carga de la tabla.
Table error	Se ha producido un error al cargar la tabla.

## Uso de Amazon CloudWatch para la monitorización de tareas de replicación

Puede utilizar alarmas o eventos de Amazon CloudWatch para hacer un seguimiento más estrecho del proceso de migración. Para obtener más información sobre Amazon CloudWatch, consulte [¿Qué son Amazon CloudWatch, los eventos de Amazon CloudWatch y los registros de Amazon CloudWatch?](#) en la Guía del usuario de Amazon CloudWatch. Tenga en cuenta que se cobra por utilizar Amazon CloudWatch.

La consola de AWS DMS muestra estadísticas básicas de CloudWatch para cada tarea, incluido el estado de la tarea, el porcentaje completado, el tiempo transcurrido y estadísticas de tabla, como se muestra a continuación. Seleccione la tarea de replicación y, a continuación, la pestaña Task monitoring.

or2pg-task

Task monitoring

Time Range: Last Hour

Below are your CloudWatch metrics for the selected resources. Click on a graph to see an expanded view.

FullLoadThroughputBandwidth Source

FullLoadThroughputBandwidth Target

FullLoadThroughputRows Target

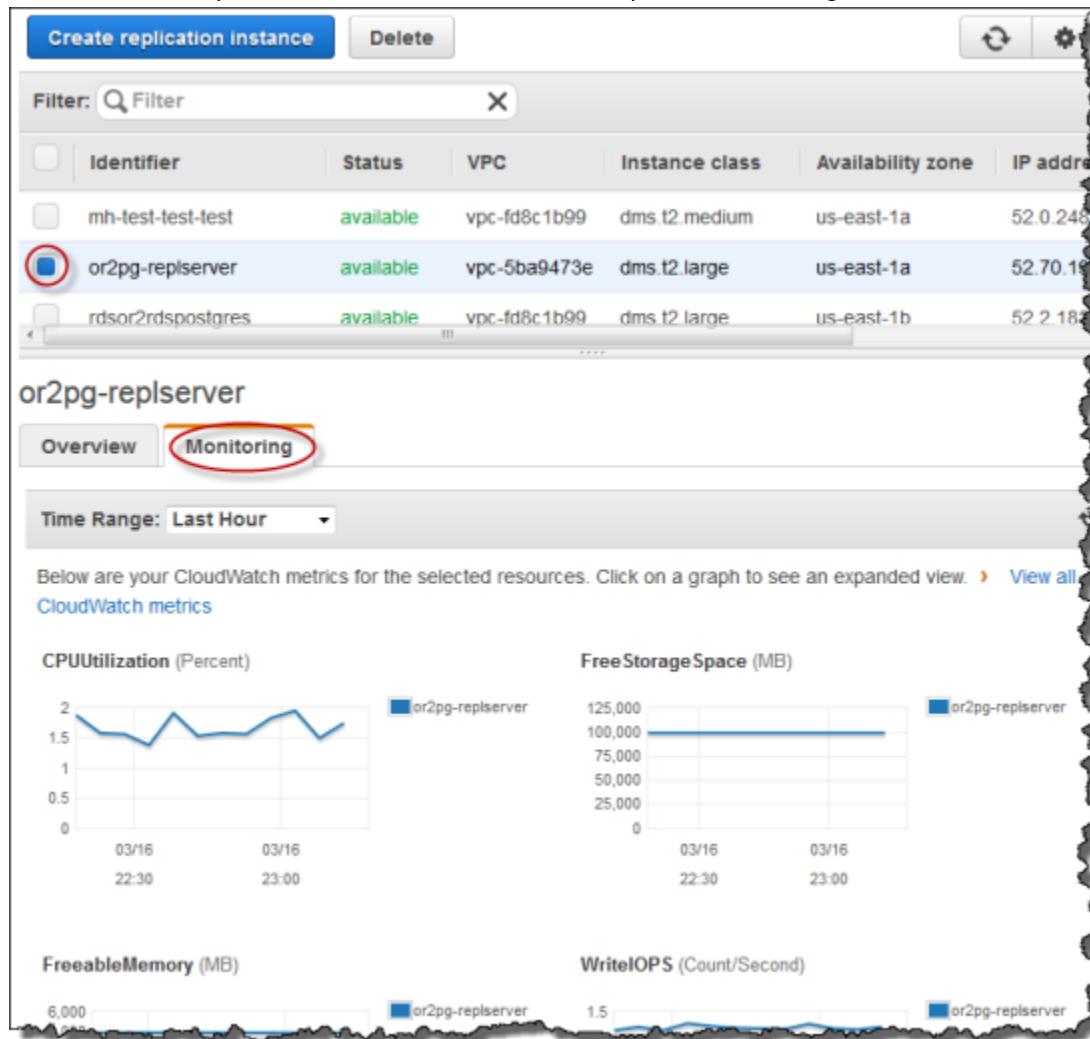
La consola de AWS DMS muestra estadísticas de desempeño de cada tabla, incluido el número de inserciones, eliminaciones y actualizaciones, al seleccionar la pestaña Table statistics (Estadísticas de tabla).

task-fkvacppsulxnqyv

Table statistics

Schema	Table	State	Inserts	Deletes	Updates	DDLs	Full Load Rows	Total	Last Update
aat	T20	Full load	0	0	0	0	16,080,394	16,080,394	3/18 19:30:00
aat	T21	Full load	0	0	0	0	16,079,437	16,079,437	3/18 19:30:00
aat	T22	Full load	0	0	0	0	15,804,000	15,804,000	3/18 19:30:00

Además, si selecciona una instancia de replicación en la página Replication Instance, podrá ver las métricas de desempeño de la instancia seleccionando la pestaña Monitoring.



## Métricas del servicio de migración de datos

AWS DMS proporciona estadísticas de lo siguiente:

- Métricas de host: estadísticas de desempeño y uso del host de replicación, las cuales proporciona Amazon CloudWatch. Para ver una lista completa de las métricas disponibles, consulte [Métricas de instancia de replicación \(p. 326\)](#).
- Métricas de tareas de replicación: las estadísticas para las tareas de replicación, incluidos los cambios entrantes y confirmados y la latencia entre el host de replicación y las bases de datos de origen y destino. Para ver una lista completa de las métricas disponibles, consulte [Métricas de tareas de replicación \(p. 327\)](#).
- Métricas de tablas: estadísticas de tablas que están en proceso de migración, incluido el número de inserciones, actualizaciones, eliminaciones e instrucciones DDL completadas.

Las métricas de tareas se dividen en estadísticas entre el host de replicación y el punto de enlace de origen, y estadísticas entre el host de replicación y el punto de enlace de destino. Puede determinar las estadísticas totales para una tarea agregando dos estadísticas relacionadas. Por ejemplo, puede determinar la latencia total o el retraso de réplica de una tarea combinando los valores CDCLatencySource y CDCLatencyTarget.

Los valores de las métricas de las tareas pueden verse afectados por la actividad en curso en la base de datos de origen. Por ejemplo, si se ha iniciado una transacción pero no se ha confirmado, la métrica CDCLatencySource seguirá creciendo hasta que la transacción se confirme.

En el caso de la instancia de replicación, es preciso realizar una aclaración para la métrica FreeableMemory. La memoria que se puede liberar no es una indicación de la memoria libre disponible real. Es la memoria que se encuentra actualmente en uso, que puede liberarse y que otros usuarios pueden utilizar; se trata de una combinación de búferes y caché en uso en la instancia de replicación.

Aunque la métrica FreeableMemory no refleja la cantidad de memoria libre real disponible, la combinación de las métricas SwapUsage y FreeableMemory puede indicar si la instancia de replicación está sobrecargada.

Monitoree estas dos métricas para detectar las siguientes condiciones.

- La métrica FreeableMemory está a punto de agotarse.
- La métrica SwapUsage aumenta o fluctúa.

Si detecta alguna de estas dos condiciones, debe sopesar la posibilidad de trasladarse a una instancia de replicación de mayor tamaño. También debe considerar la posibilidad de reducir el número y el tipo de tareas que se ejecutan en la instancia de replicación. Las tareas de carga completa requieren más memoria que las tareas que solo replican los cambios.

## Métricas de instancia de replicación

La monitorización de las instancias de replicación incluye métricas de Amazon CloudWatch para las siguientes estadísticas:

### CPUUtilization

La cantidad de CPU utilizada.

Unidades: bytes

### FreeStorageSpace

La cantidad de espacio de almacenamiento disponible.

Unidades: bytes

### FreeableMemory

La cantidad de memoria de acceso aleatorio disponible.

Unidades: bytes

### WriteIOPS

El número medio de operaciones de E/S en el disco por segundo.

Unidades: recuento/segundo

### ReadIOPS

El número medio de operaciones de E/S en el disco por segundo.

Unidades: recuento/segundo

**WriteThroughput**

Número medio de bytes que se escriben en el disco por segundo.

Unidades: bytes/segundo

**ReadThroughput**

El número medio de bytes leídos del disco por segundo.

Unidades: bytes/segundo

**WriteLatency**

Tiempo medio de cada operación de E/S (salida) en disco.

Unidades: milisegundos

**ReadLatency**

Tiempo medio de cada operación de E/S (entrada) en disco.

Unidades: milisegundos

**SwapUsage**

Cantidad de espacio de intercambio utilizada en la instancia de replicación.

Unidades: bytes

**NetworkTransmitThroughput**

El tráfico de red de salida (transferencia) en la instancia de replicación, incluidos el tráfico de base de datos del cliente y el tráfico de AWS DMS utilizado en monitorización y en replicación.

Unidades: bytes/segundo

**NetworkReceiveThroughput**

El tráfico de red de entrada (recepción) en la instancia de replicación, incluidos el tráfico de base de datos del cliente y el tráfico de AWS DMS utilizado en monitorización y en replicación.

Unidades: bytes/segundo

## Métricas de tareas de replicación

La monitorización de tareas de replicación incluye métricas para las estadísticas siguientes:

**FullLoadThroughputBandwidthSource**

El ancho de banda de red entrante de una carga completa desde el origen en kilobytes (KB) por segundo.

**FullLoadThroughputBandwidthTarget**

El ancho de banda de red saliente de una carga completa para el destino en KB por segundo.

**FullLoadThroughputRowsSource**

Cambios entrantes de una carga completa desde el origen en filas por segundo.

**FullLoadThroughputRowsTarget**

Cambios salientes de una carga completa para el destino en filas por segundo.

#### CDCIncomingChanges

Número total de eventos de cambio en un punto en el tiempo que están a la espera de aplicarse en el destino. Tenga en cuenta que esto no es lo mismo que una medida de la velocidad de cambio de las transacciones del punto de enlace de origen. Cuando el valor de esta métrica es alto, normalmente significa que AWS DMS no puede aplicar los cambios capturados con puntualidad, lo que provoca una latencia de destino más elevada.

#### CDCChangesMemorySource

Cantidad de filas que se acumulan en una memoria y que esperan a confirmarse desde el origen.

#### CDCChangesMemoryTarget

Cantidad de filas que se acumulan en una memoria y que esperan a confirmarse en el destino.

#### CDCChangesDiskSource

Cantidad de filas que se acumulan en el disco y que esperan a confirmarse desde el origen.

#### CDCChangesDiskTarget

Cantidad de filas que se acumulan en el disco y que esperan a confirmarse en el destino.

#### CDCThroughputBandwidthSource

Ancho de banda de red del destino en KB por segundo. CDCThroughputBandwidth registra el ancho de banda en puntos de muestreo. Si no se encuentra tráfico de red, el valor es cero. Como CDC no emite transacciones de larga duración, puede que el tráfico de red no se registre.

#### CDCThroughputBandwidthTarget

Ancho de banda de red del destino en KB por segundo. CDCThroughputBandwidth registra el ancho de banda en puntos de muestreo. Si no se encuentra tráfico de red, el valor es cero. Como CDC no emite transacciones de larga duración, puede que el tráfico de red no se registre.

#### CDCThroughputRowsSource

Cambios de tareas entrantes del origen en filas por segundo.

#### CDCThroughputRowsTarget

Cambios de tareas salientes para el destino en filas por segundo.

#### CDCLatencySource

La diferencia, en segundos, entre el último evento capturado en el punto de enlace de origen y la marca temporal del sistema actual de la instancia de AWS DMS. Si no se ha capturado ningún cambio en el origen debido al alcance de la tarea, AWS DMS establece este valor en cero.

#### CDCLatencyTarget

La diferencia, en segundos, entre la primera marca temporal de evento que espera confirmación en el destino y la marca temporal actual de la instancia de AWS DMS. Este valor se produce si hay transacciones que no gestiona el destino. De lo contrario, la latencia de destino es la misma que la latencia de origen si se aplican todas las transacciones. La latencia de destino nunca debe ser inferior a la latencia de origen.

## Administración de logs de tareas de AWS DMS

AWS DMS utiliza Amazon CloudWatch para registrar información sobre las tareas realizadas durante el proceso de migración. Puede utilizar la CLI de AWS o la API de AWS DMS para ver información sobre los

logs de tareas. Para ello, utilice el comando `describe-replication-instance-task-logs` de la CLI de AWS o la acción `DescribeReplicationInstanceTaskLogs` de la API de AWS DMS.

Por ejemplo, el siguiente comando de la CLI de AWS muestra los metadatos del log de tareas con formato JSON.

```
$ aws dms describe-replication-instance-task-logs \
--replication-instance-arn arn:aws:dms:us-east-1:237565436:rep:CDSFSFSFFFSSUFCAY
```

A continuación se muestra un ejemplo de respuesta del comando.

```
{
    "ReplicationInstanceTaskLogs": [
        {
            "ReplicationTaskArn": "arn:aws:dms:us-
east-1:237565436:task:MY34U6Z4MSY52GRTIX3O4AY",
            "ReplicationTaskName": "mysql-to-ddb",
            "ReplicationInstanceTaskLogSize": 3726134
        }
    ],
    "ReplicationInstanceArn": "arn:aws:dms:us-east-1:237565436:rep:CDSFSFSFFFSSUFCAY"
}
```

En esta respuesta, existe un único log de tareas (`mysql-to-ddb`) asociado a la instancia de replicación. El tamaño de este log es de 3 726 124 bytes.

Puede utilizar la información que devuelve `describe-replication-instance-task-logs` para diagnosticar y solucionar problemas con los logs de tareas. Por ejemplo, si habilita el registro de depuración detallado para una tarea, el log de tareas aumentará de tamaño rápidamente, lo que podría consumir todo el almacenamiento disponible en la instancia de replicación y hacer que el estado de la instancia cambie a `storage-full`. Al describir los registros de tareas, puede determinar cuáles son los que ya no necesita y eliminarlos para liberar espacio de almacenamiento.

Para eliminar los logs de tareas para una tarea, defina la opción `DeleteTaskLogs` de la tarea en `true`. Por ejemplo, el JSON siguiente elimina los logs de tareas cuando se modifica una tarea utilizando el comando `modify-replication-task` de la CLI de AWS o la acción `ModifyReplicationTask` de la API de AWS DMS.

```
{
    "Logging": {
        "DeleteTaskLogs":true
    }
}
```

## Registro de llamadas al API de AWS DMS con AWS CloudTrail

AWS DMS está integrado con AWS CloudTrail, un servicio que proporciona un registro de las acciones realizadas por un usuario, una función o un servicio de AWS en AWS DMS. CloudTrail captura todas las llamadas a la API de AWS DMS como eventos, incluidas las llamadas procedentes de la consola de AWS.

DMS y de las llamadas del código a las API de AWS DMS. Si crea un registro de seguimiento, puede habilitar la entrega continua de eventos de CloudTrail a un bucket de Amazon S3, incluidos los eventos de AWS DMS. Si no configura un registro de seguimiento, puede ver los eventos más recientes en la consola de CloudTrail en el Event history (Historial de eventos). Mediante la información que recopila CloudTrail, se puede determinar la solicitud que se envió a AWS DMS, la dirección IP desde la que se realizó la solicitud, quién la realizó, cuándo la realizó y los detalles adicionales.

Para obtener más información sobre CloudTrail, consulte la [AWS CloudTrail User Guide](#).

## Información de AWS DMS en CloudTrail

CloudTrail se habilita en una cuenta de AWS al crearla. Cuando se produce una actividad en AWS DMS, dicha actividad se registra en un evento de CloudTrail junto con los eventos de los demás servicios de AWS en el Event history (Historial de eventos). Puede ver, buscar y descargar los últimos eventos de la cuenta de AWS. Para obtener más información, consulte [Visualización de eventos con el historial de eventos de CloudTrail](#).

Para mantener un registro continuo de los eventos de la cuenta de AWS, incluidos los eventos de AWS DMS, cree un registro de seguimiento. Un registro de seguimiento permite a CloudTrail enviar archivos de registro a un bucket de Amazon S3. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las regiones. El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de Amazon S3 especificado. También puede configurar otros servicios de AWS para analizar y actuar en función de los datos de eventos recopilados en los registros de CloudTrail. Para obtener más información, consulte:

- [Introducción a la creación de registros de seguimiento](#)
- [Servicios e integraciones compatibles con CloudTrail](#)
- [Configuración de notificaciones de Amazon SNS para CloudTrail](#)
- [Recibir archivos de registro de CloudTrail de varias regiones](#) y [Recepción de archivos de registro de CloudTrail de varias cuentas](#)

Todas las acciones de AWS DMS las registra CloudTrail y están documentadas en la [AWS Database Migration Service API Reference](#). Por ejemplo, las llamadas a las acciones `CreateReplicationInstance`, `TestConnection` y `StartReplicationTask` generan entradas en los archivos de registro de CloudTrail.

Cada entrada de registro o evento contiene información acerca de quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con las credenciales raíz o del usuario de IAM.
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro servicio de AWS.

Para obtener más información, consulte el elemento `userIdentity` de [CloudTrail](#).

## Descripción de las entradas de archivos de registro de AWS DMS

Un registro de seguimiento es una configuración que permite la entrega de eventos como archivos de registro al bucket de Amazon S3 que se especifique. Los archivos de registro de CloudTrail contienen una o varias entradas de registro. Un evento representa una única solicitud de cualquier origen e incluye información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud,

etcétera. Los archivos de registro de CloudTrail no son un rastro de la pila ordenada de las llamadas a la API públicas, por lo que no aparecen en ningún orden específico.

En el ejemplo siguiente, se muestra una entrada de registro de CloudTrail que ilustra la acción RebootReplicationInstance.

```
{  
    "eventVersion": "1.05",  
    "userIdentity": {  
        "type": "AssumedRole",  
        "principalId": "AKIAIOSFODNN7EXAMPLE:johndoe",  
        "arn": "arn:aws:sts::123456789012:assumed-role/admin/johndoe",  
        "accountId": "123456789012",  
        "accessKeyId": "ASIAZYFI33SINADOJJEZW",  
        "sessionContext": {  
            "attributes": {  
                "mfaAuthenticated": "false",  
                "creationDate": "2018-08-01T16:42:09Z"  
            },  
            "sessionIssuer": {  
                "type": "Role",  
                "principalId": "AKIAIOSFODNN7EXAMPLE",  
                "arn": "arn:aws:iam::123456789012:role/admin",  
                "accountId": "123456789012",  
                "userName": "admin"  
            }  
        }  
    },  
    "eventTime": "2018-08-02T00:11:44Z",  
    "eventSource": "dms.amazonaws.com",  
    "eventName": "RebootReplicationInstance",  
    "awsRegion": "us-east-1",  
    "sourceIPAddress": "72.21.198.64",  
    "userAgent": "console.amazonaws.com",  
    "requestParameters": {  
        "forceFailover": false,  
        "replicationInstanceArn": "arn:aws:dms:us-  
east-1:123456789012:rep:EX4MBJ2NMRDL3BMAYJOXUGYPUE"  
    },  
    "responseElements": {  
        "replicationInstance": {  
            "replicationInstanceId": "replication-instance-1",  
            "replicationInstanceStatus": "rebooting",  
            "allocatedStorage": 50,  
            "replicationInstancePrivateIpAddresses": [  
                "172.31.20.204"  
            ],  
            "instanceCreateTime": "Aug 1, 2018 11:56:21 PM",  
            "autoMinorVersionUpgrade": true,  
            "engineVersion": "2.4.3",  
            "publiclyAccessible": true,  
            "replicationInstanceClass": "dms.t2.medium",  
            "availabilityZone": "us-east-1b",  
            "kmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-  
e8494fa3921a",  
            "replicationSubnetGroup": {  
                "vpcId": "vpc-1f6a9c6a",  
                "subnetGroupStatus": "Complete",  
                "replicationSubnetGroupArn": "arn:aws:dms:us-  
east-1:123456789012:subgrp:EDHRVRBAAAPONQAIYWP4NUW22M",  
                "subnets": [  
                    {  
                        "subnetIdentifier": "subnet-cbfff283",  
                        "subnetAvailabilityZone": {  
                            "name": "us-east-1b"  
                        }  
                    }  
                ]  
            }  
        }  
    }  
}
```

```
        "name": "us-east-1b"
    },
    "subnetStatus": "Active"
},
{
    "subnetIdentifier": "subnet-d7c825e8",
    "subnetAvailabilityZone": {
        "name": "us-east-1e"
    },
    "subnetStatus": "Active"
},
{
    "subnetIdentifier": "subnet-6746046b",
    "subnetAvailabilityZone": {
        "name": "us-east-1f"
    },
    "subnetStatus": "Active"
},
{
    "subnetIdentifier": "subnet-bac383e0",
    "subnetAvailabilityZone": {
        "name": "us-east-1c"
    },
    "subnetStatus": "Active"
},
{
    "subnetIdentifier": "subnet-42599426",
    "subnetAvailabilityZone": {
        "name": "us-east-1d"
    },
    "subnetStatus": "Active"
},
{
    "subnetIdentifier": "subnet-da327bf6",
    "subnetAvailabilityZone": {
        "name": "us-east-1a"
    },
    "subnetStatus": "Active"
}
],
    "replicationSubnetGroupIdentifier": "default-vpc-1f6a9c6a",
    "replicationSubnetGroupDescription": "default group created by console for
vpc id vpc-1f6a9c6a"
},
    "replicationInstanceEniId": "eni-0d6db8c7137cb9844",
    "vpcSecurityGroups": [
        {
            "vpcSecurityGroupId": "sg-f839b688",
            "status": "active"
        }
    ],
    "pendingModifiedValues": {},
    "replicationInstancePublicIpAddresses": [
        "18.211.48.119"
    ],
    "replicationInstancePublicIpAddress": "18.211.48.119",
    "preferredMaintenanceWindow": "fri:22:44-fri:23:14",
    "replicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:EX4MBJ2NMRDL3BMAYJOXUGYPUE",
    "replicationInstanceEniIds": [
        "eni-0d6db8c7137cb9844"
    ],
    "multiAZ": false,
    "replicationInstancePrivateIpAddress": "172.31.20.204",
    "patchingPrecedence": 0
}
```

```
},
"requestID": "a3c83c11-95e8-11e8-9d08-4b8f2b45bfd5",
"eventID": "b3c4adb1-e34b-4744-bdeb-35528062a541",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}
```

# Validación de tareas de AWS DMS

## Temas

- [Estadísticas de la tarea de replicación \(p. 335\)](#)
- [Revalidación de tablas durante una tarea \(p. 337\)](#)
- [Solución de problemas \(p. 337\)](#)
- [Limitaciones \(p. 338\)](#)

AWS DMS ofrece soporte para la validación de datos, para garantizar que los datos se han migrado de forma precisa del origen al destino. Si lo habilita para la tarea, entonces AWS DMS empieza a comparar los datos de origen y de destino inmediatamente después de realizar una carga completa para una tabla.

La validación de datos es opcional. AWS DMS compara los registros de origen y de destino y notifica cualquier discrepancia. Además, para una tarea habilitada para CDC, AWS DMS compara los cambios incrementales y notifica cualquier discrepancia.

Durante la validación de datos, AWS DMS compara cada fila en el origen con su fila correspondiente en el destino y verifica que esas filas contienen los mismos datos. Para conseguir esto, AWS DMS realiza consultas adecuadas para recuperar los datos. Tenga en cuenta que estas consultas consumirán recursos adicionales en el origen y el destino, así como recursos de red adicionales.

La validación de datos funciona con las siguientes bases de datos siempre que AWS DMS las admite como puntos de enlace de origen y de destino:

- Oracle
- PostgreSQL
- MySQL
- MariaDB
- Microsoft SQL Server
- Amazon Aurora (MySQL)
- Amazon Aurora (PostgreSQL)

Para obtener más información acerca de los puntos de enlace admitidos, consulte [Trabajo con puntos de enlace de AWS DMS \(p. 90\)](#).

La validación de datos requiere más tiempo, más allá de la cantidad necesaria para la migración en sí. El tiempo extra necesario depende de la cantidad de datos que se ha migrado.

La configuración de la validación de datos incluye lo siguiente:

- `EnableValidation`: Habilita o deshabilita la validación de datos.
- `FailureMaxCount`: especifica el número máximo de registros que pueden generar un error de validación antes de que se suspenda la validación de la tarea.
- `HandleCollationDiff`: cuentas para diferencias de intercalación de columna en puntos de enlace de PostgreSQL cuando se identifican registros de origen y de destino para comparar.
- `RecordFailureDelayLimitInMinutes`: especifica el retraso antes de notificar los detalles de error de validación.
- `TableFailureMaxCount`: especifica el número máximo de tablas que pueden generar un error de validación antes de que se suspenda la validación de la tarea.
- `ThreadCount`: ajusta el número de subprocessos de ejecución que AWS DMS utiliza durante la validación.

- **ValidationOnly**: ofrece una vista previa de la validación para la tarea sin tener que realizar ninguna para tarea de migración o replicación de datos.

Por ejemplo, el siguiente JSON activa la validación, aumenta el número de subprocessos a ocho y suspende la validación si alguna tabla tiene un error de validación.

```
ValidationSettings": {  
    "EnableValidation":true,  
    "ThreadCount":8,  
    "TableFailureMaxCount":1  
}
```

Para obtener más información sobre estas opciones, consulte [Configuración de tareas de validación de datos \(p. 270\)](#).

## Estadísticas de la tarea de replicación

Cuando se habilita la validación de datos, AWS DMS ofrece las siguientes estadísticas en la tabla:

- **ValidationState**: el estado de validación de la tabla. El parámetro puede tener los siguientes valores:
  - Not enabled (No habilitada): la validación no está habilitada para la tabla en la tarea de migración.
  - Pending records (Registros pendientes): algunos registros en la tabla están a la espera de validación.
  - Mismatched records (Discrepancia en los registros): algunos registros de la tabla no coinciden entre origen y destino. Se puede producir una discrepancia por varios motivos. Para obtener más información, consulte la tabla `awsdms_validation_failures` en el punto de enlace de destino.
  - Suspended records (Registros suspendidos): no se han podido validar algunos registros de la tabla.
  - No primary key (No hay clave principal): no se ha podido validar la tabla porque no tenía clave principal.
  - Table error (Error de tabla): la tabla no se ha validado porque estaba en un estado de error y no se han migrado algunos datos.
  - Validated (Validada): se han validado todas las filas de la tabla. Si se actualiza la tabla, el estado puede cambiar de Validated.
  - Error: no se ha podido validar la tabla debido a un error inesperado.
- **ValidationPending**: el número de registros que se han migrado al destino, pero que aún no se han validado.

**ValidationSuspended**: el número de registros que AWS DMS no puede comparar. Por ejemplo, si un registro en el origen se actualiza constantemente, AWS DMS no puede comparar el origen y el destino. Para obtener más información, consulte [Configuración de las tareas de administración de errores \(p. 272\)](#)

- **ValidationFailed**: el número de registros que no ha superado la fase de validación de datos. Para obtener más información, consulte [Configuración de las tareas de administración de errores \(p. 272\)](#).
- **ValidationSucceededRecordCount**: el número de filas que AWS DMS ha validado por minuto.
- **ValidationAttemptedRecordCount**: el número de filas en las que se intentó realizar la validación, por minuto.
- **ValidationFailedOverallCount**: el número de filas en las que falló la validación.
- **ValidationSuspendedOverallCount**: el número de filas en las que se suspendió la validación.
- **ValidationPendingOverallCount**: el número de filas en las que la validación todavía está pendiente.

- ValidationBulkQuerySourceLatency: AWS DMS puede realizar la validación de datos por lotes, sobre todo en ciertos escenarios durante una carga completa o una replicación continua en las que hay muchos cambios. Esta métrica indica la latencia necesaria para leer un conjunto masivo de datos desde el punto de enlace de origen.
- ValidationBulkQueryTargetLatency: AWS DMS puede realizar la validación de datos por lotes, sobre todo en ciertos escenarios durante una carga completa o una replicación continua en las que hay muchos cambios. Esta métrica indica la latencia necesaria para leer un conjunto masivo de datos en el punto de enlace de destino.
- ValidationItemQuerySourceLatency: durante la replicación continua, la validación de datos puede identificar los cambios en curso y validarlos. Esta métrica indica la latencia para leer esos cambios desde el origen. La validación puede ejecutar más consultas de las necesarias en función del número de cambios, si hay errores durante la validación.
- ValidationItemQueryTargetLatency: durante la replicación continua, la validación de datos puede identificar los cambios en curso y validarlos fila por fila. Esta métrica nos indica la latencia para leer esos cambios desde el destino. La validación puede ejecutar más consultas de las necesarias en función del número de cambios, si hay errores durante la validación.

Puede ver la información de validación de datos a través de la consola de , la CLI de AWS o la API de AWS DMS.

- En la consola, puede elegir validar una tarea cuando crea o modifica la tarea. Para ver el informe de validación de datos mediante la consola, elija la tarea en la página Tasks y seleccione la pestaña Table statistics en la sección de detalles.
- Establezca, con la interfaz de línea de comandos (CLI), el parámetro `EnableValidation` en true al crear o modificar una tarea para iniciar la validación de datos. En el siguiente ejemplo se crea una tarea y permite la validación de datos.

```
create-replication-task
  --replication-task-settings '{"ValidationSettings":{"EnableValidation":true}}'
  --replication-instance-arn arn:aws:dms:us-east-1:5731014:
    rep:36KWVMB7Q
  --source-endpoint-arn arn:aws:dms:us-east-1:5731014:
    endpoint:CSZAEFQURFYMM
  --target-endpoint-arn arn:aws:dms:us-east-1:5731014:
    endpoint:CGPP7MF6WT4JQ
  --migration-type full-load-and-cdc
  --table-mappings '{"rules": [{"rule-type": "selection", "rule-id": "1",
    "rule-name": "1", "object-locator": {"schema-name": "data_types", "table-name": "%"}, "rule-action": "include"}]}'
```

Utilice el comando `describe-table-statistics` para recibir el informe de validación de datos en formato JSON. El siguiente comando muestra el informe de validación de datos.

```
aws dms describe-table-statistics --replication-task-arn arn:aws:dms:us-east-1:5731014:
  rep:36KWVMB7Q
```

El informe debería ser similar al siguiente.

```
{
  "ReplicationTaskArn": "arn:aws:dms:us-west-2:5731014:task:VFPFTYKK2RYSI",
  "TableStatistics": [
    {
      "ValidationPendingRecords": 2,
      "Inserts": 25,
      "ValidationState": "Pending records",
      "ValidationSuspendedRecords": 0,
```

```
        "LastUpdateTime": 1510181065.349,
        "FullLoadErrorRows": 0,
        "FullLoadCondtnlChkFailedRows": 0,
        "Ddls": 0,
        "TableName": "t_binary",
        "ValidationFailedRecords": 0,
        "Updates": 0,
        "FullLoadRows": 10,
        "TableState": "Table completed",
        "SchemaName": "d_types_s_sqlserver",
        "Deletes": 0
    }
}
```

- Con la API de AWS DMS, cree una tarea mediante la acción `CreateReplicationTask` y establezca el parámetro `EnableValidation` en `true` para validar los datos migrados por la tarea. Utilice la acción `DescribeTableStatistics` para recibir el informe de validación de datos en formato JSON.

## Revalidación de tablas durante una tarea

Mientras se ejecuta una tarea, puede solicitar a AWS DMS que realice la validación de datos.

### Consola de administración de AWS

1. Inicie sesión en la Consola de administración de AWS y elija AWS DMS. Si está registrado como usuario de AWS Identity and Access Management (IAM), debe disponer de los permisos adecuados para acceder a AWS DMS. Para obtener más información sobre los permisos que se necesitan, consulte [Permisos de IAM necesarios para utilizar AWS DMS \(p. 35\)](#).
2. Elija Tasks en el panel de navegación.
3. Elija la tarea en ejecución que tiene la tabla que desea volver a validar.
4. Elija la pestaña Table Statistics (Estadísticas de la tabla).
5. Elija la tabla que desea volver a validar. Si la tarea ya no se está ejecutando, no puede volver a validar la tabla.
6. Elija Revalidate (Volver a validar).

## Solución de problemas

Durante la validación, AWS DMS crea una tabla nueva en el punto de enlace de destino: `awsdms_validation_failures_v1`. Si cualquier registro entra en estado `ValidationSuspended` o `ValidationFailed` AWS DMS escribe información de diagnóstico en `awsdms_validation_failures_v1`. Puede consultar esta tabla para ayudar a solucionar errores de validación.

A continuación se muestra una descripción de la tabla `awsdms_validation_failures_v1`:

Nombre de la columna	Tipo de datos	Descripción
<code>TASK_NAME</code>	<code>VARCHAR(128) NOT NULL</code>	Identificador de tareas de AWS DMS
<code>TABLE_OWNER</code>	<code>VARCHAR(128) NOT NULL</code>	Esquema (propietario) de la tabla.

Nombre de la columna	Tipo de datos	Descripción
TABLE_NAME	VARCHAR( 128 ) NOT NULL	Nombre de la tabla.
FAILURE_TIME	DATETIME( 3 ) NOT NULL	Hora cuando se produjo el error.
KEY	TEXT NOT NULL	Esta es la clave principal para el tipo de registro de fila.
FAILURE_TYPE	VARCHAR(128) NOT NULL	Gravedad de error de validación. Puede ser Failed o Suspended.

La siguiente consulta le mostrará todos los errores para una tarea consultando la tabla `awsdms_validation_failures_v1`. El nombre de la tarea debe ser el ID del recurso externo de la tarea. El ID del recurso externo de la tarea es el último valor en el ARN de la tarea. Por ejemplo, en el caso de una tarea con un valor de ARN `arn:aws:dms:us-west-2:5599:task: VFPFKH4FJR3FTYKK2RYSI`, el ID del recurso externo de la tarea sería `VFPFKH4FJR3FTYKK2RYSI`.

```
select * from awsdms_validation_failures_v1 where TASK_NAME = 'VFPFKH4FJR3FTYKK2RYSI'
```

Una vez que tenga la clave principal del registro con error, puede consultar los puntos de enlace de origen y destino para ver qué parte del registro no coincide.

## Limitaciones

- La validación de datos requiere que la tabla tenga una clave principal o índice único.
- Las columnas de clave principal no pueden ser del tipo CLOB, BLOB o BYTE.
- Para las columnas de clave principal de tipo VARCHAR o CHAR, la longitud debe ser inferior a 1024.
- Si la intercalación de la columna de clave principal en la instancia de PostgreSQL de destino no se ha establecido en "C", el orden de clasificación de la clave principal será diferente en comparación con el de Oracle. La validación de datos producirá un error al validar los registros si el orden de clasificación es diferente entre PostgreSQL y Oracle.
- La validación de datos genera consultas adicionales en las bases de datos de origen y de destino. Debe asegurarse de que ambas bases de datos tengan suficientes recursos para gestionar esta carga adicional.
- La validación de datos no se admite si una migración utiliza el filtrado personalizado o se consolidan varias bases de datos en una.
- Para un punto de enlace de Oracle de origen o destino, AWS DMS utiliza DBMS\_CRYPTO para la validación de BLOB. Si su punto de enlace de Oracle utiliza BLOB, entonces debe conceder el permiso de ejecución en dbms\_crypto a la cuenta de usuario que se utiliza para obtener acceso al punto de enlace de Oracle. Puede hacerlo ejecutando la siguiente instrucción:

```
grant execute on sys.dbms_crypto to <dms_endpoint_user>;
```

- Si la base de datos de destino se modifica fuera de AWS DMS durante la validación, las discrepancias podrían no notificarse con precisión. Esto puede ocurrir si una de sus aplicaciones escribe datos en la tabla de destino, mientras AWS DMS realiza la validación en esa misma tabla.

- Si se modifican continuamente una o más filas durante la validación, AWS DMS no puede validar esas filas. Sin embargo, puede validar esas filas manualmente, una vez finalizada la tarea.
- Si AWS DMS detecta más de 10 000 registros suspendidos o con error, detiene la validación. Antes de continuar, deberá resolver los problemas subyacentes con los datos.

# Etiquetado de recursos en AWS Database Migration Service

Puede utilizar etiquetas en AWS Database Migration Service (AWS DMS) para agregar metadatos a sus recursos. Además, puede usar estas etiquetas junto con políticas de AWS Identity and Access Management (IAM) para administrar el acceso a los recursos de AWS DMS y controlar qué acciones se pueden aplicar a los recursos de AWS DMS. Por último, estas etiquetas se pueden utilizar para hacer un seguimiento de costos, agrupando los gastos correspondientes a recursos con etiqueta similar.

Todos los recursos de AWS DMS pueden etiquetarse:

- Instancias de replicación
- Puntos de enlace
- Tareas de replicación
- Certificados

Las etiquetas de AWS DMS son pares nombre-valor que el usuario define y asocia a un recurso de AWS DMS. El nombre es la clave. Si lo desea puede proporcionar un valor para la clave o no. También puede usar etiquetas para asignar información arbitraria a un recurso de AWS DMS. Las claves de etiqueta podrían utilizarse, por ejemplo, para definir una categoría, y el valor de la etiqueta podría ser un elemento dentro de esa categoría. Por ejemplo, puede definir una clave de etiqueta "proyecto" y un valor de etiqueta "Salix" para indicar que el recurso de AWS DMS va asignado al proyecto Salix. También puede usar etiquetas para designar recursos de AWS DMS para pruebas o para producción a través de una clave como entorno=prueba o entorno=producción. Se recomienda utilizar un conjunto coherente de claves de etiqueta que facilite el seguimiento de los metadatos asociados a los recursos de AWS DMS.

Use etiquetas para organizar sus facturas de AWS de manera que reflejen su propia estructura de costos. Para ello, regístrese para obtener una factura de su cuenta de AWS que incluya valores de clave de etiquetas. A continuación, para ver los costos de los recursos combinados, organice la información de facturación de acuerdo con los recursos con los mismos valores de clave de etiquetas. Por ejemplo, puede etiquetar varios recursos con un nombre de aplicación específico y luego organizar su información de facturación para ver el costo total de la aplicación en distintos servicios. Para obtener más información, consulte [Cost Allocation and Tagging](#) en [About AWS Billing and Cost Management](#).

Cada recurso de AWS DMS tiene un conjunto de etiquetas con todas las etiquetas asignadas a ese recurso de AWS DMS. Un conjunto de etiquetas puede contener hasta diez etiquetas o puede estar vacío. Si agrega una etiqueta a un recurso de AWS DMS con la misma clave que una etiqueta existente en el recurso, el nuevo valor sobrescribirá al antiguo.

AWS no aplica ningún significado semántico a las etiquetas, que se interpretan estrictamente como cadenas de caracteres. AWS DMS podría definir etiquetas en un recurso de AWS DMS, con arreglo a la configuración utilizada al crear el recurso.

En la siguiente lista se describen las características de una etiqueta de AWS DMS.

- La clave de la etiqueta es el nombre obligatorio de la etiqueta. El valor de la cadena puede tener una longitud de entre 1 y 128 caracteres Unicode y no puede llevar los prefijos "aws:" ni "dms:". La cadena puede contener únicamente el conjunto de letras Unicode, dígitos, espacio en blanco, '\_', '!', '/', '=', '+', '-' (Java regex: " $^(\backslash\backslash\p{L}\backslash\p{Z})\backslash\p{N}_\cdot\cdot\cdot\backslash\p{N}=\backslash\backslash\p{N}^*$ ").
- El valor de etiqueta es un valor de cadena optativo en la etiqueta. El valor de cadena puede tener una longitud de entre 1 y 256 caracteres Unicode y no puede llevar los prefijos "aws:" ni "dms:". La cadena

puede contener únicamente el conjunto de letras Unicode, dígitos, espacio en blanco, '\_','.', '/', '=', '+', '-' (Java regex: "^(\\p{L}\\p{Z}\\p{N}\_.:/=-\\-)\*\$").

Los valores no deben ser únicos dentro de un conjunto de etiquetas y también pueden ser nulos. Por ejemplo, puede tener un par clave-valor en un conjunto de etiquetas en proyecto/Trinity y centro-de-costos/Trinity.

Puede utilizar la CLI de AWS o la API de AWS DMS para añadir, listar y eliminar etiquetas de los recursos de AWS DMS. Si utiliza la CLI de AWS o la API de AWS DMS, deberá proporcionar el nombre de recurso de Amazon (ARN) correspondiente al recurso de AWS DMS con el que desee trabajar. Para obtener más información sobre cómo crear un ARN, consulte [Creación de un nombre de recurso de Amazon \(ARN\) para usarlo con AWS DMS \(p. 14\)](#).

Tenga en cuenta que las etiquetas se almacenan en caché con fines de autorización. Por este motivo, cuando se actualizan o se agregan valores a las etiquetas de recursos de AWS DMS, pueden tardar varios minutos en estar disponibles.

## API

Puede agregar, listar o eliminar etiquetas para una recurso de AWS DMS utilizando la API de AWS DMS.

- Para agregar una etiqueta a un recurso de AWS DMS, utilice la operación [AddTagsToResource](#).
- Para listar etiquetas asignadas a un recurso de AWS DMS, utilice la operación [ListTagsForResource](#).
- Para eliminar etiquetas de un recurso de AWS DMS, utilice la operación [RemoveTagsFromResource](#).

Para obtener más información acerca de cómo crear el ARN requerido, consulte [Creación de un nombre de recurso de Amazon \(ARN\) para usarlo con AWS DMS \(p. 14\)](#).

Cuando se trabaja con XML mediante la API de AWS DMS, las etiquetas utilizan el esquema siguiente:

```
<Tagging>
<TagSet>
<Tag>
<Key>Project</Key>
<Value>Trinity</Value>
</Tag>
<Tag>
<Key>User</Key>
<Value>Jones</Value>
</Tag>
</TagSet>
</Tagging>
```

La tabla siguiente proporciona una lista de las etiquetas XML permitidas y sus características. Tenga en cuenta que los valores de clave y de valor distinguen entre mayúsculas y minúsculas. Por ejemplo, proyecto=Trinity y PROYECTO=Trinity son dos etiquetas diferentes.

Elemento de etiquetado	Descripción
TagSet	Los conjuntos de etiquetas contienen todas las etiquetas asignadas a un recurso de Amazon RDS. Solo puede haber un conjunto de etiquetas por recurso. Solo puede trabajar con conjuntos de etiquetas a través de la API de AWS DMS.

Elemento de etiquetado	Descripción
Etiqueta	Las etiquetas son pares clave-valor que define el usuario. En un conjunto de etiquetas puede haber entre 1 y 10 etiquetas.
Key	<p>La clave es el nombre obligatorio de la etiqueta. El valor de cadena puede tener una longitud entre 1 y 128 caracteres Unicode y no puede llevar los prefijos "aws:" ni "dms:". La cadena puede contener solo el conjunto de letras Unicode, dígitos, espacio en blanco, '_', '.', '/', '=' , '+' , '-' (Java regex: "<code>^([\\p{L}\\p{Z}\\p{N}_.:=+\\-]*\$</code>").</p> <p>Las claves deben ser únicas dentro de un conjunto de etiquetas. Por ejemplo, en un conjunto de etiquetas no puede haber claves iguales pero con valores diferentes, como proyecto/Trinity y proyecto/Xanadu.</p>
Value	<p>El valor es la parte opcional de la etiqueta. El valor de cadena puede tener una longitud entre 1 y 256 caracteres Unicode y no puede llevar los prefijos "aws:" ni "dms:". La cadena puede contener solo el conjunto de letras Unicode, dígitos, espacio en blanco, '_', '.', '/', '=' , '+' , '-' (Java regex: "<code>^([\\p{L}\\p{Z}\\p{N}_.:=+\\-]*\$</code>").</p> <p>Los valores no deben ser únicos dentro de un conjunto de etiquetas y también pueden ser nulos. Por ejemplo, puede tener un par clave-valor en un conjunto de etiquetas en proyecto/Trinity y centro-de-costos/Trinity.</p>

# Uso de eventos y notificaciones en AWS Database Migration Service

AWS Database Migration Service (AWS DMS) utiliza Amazon Simple Notification Service (Amazon SNS) para enviar notificaciones cuando se produce un evento de AWS DMS, por ejemplo, la creación o eliminación de una instancia de replicación. Puede trabajar con estas notificaciones de cualquier forma que admita Amazon SNS para una región de AWS como, por ejemplo, un mensaje de correo electrónico, un mensaje de texto o una llamada a un punto de enlace HTTP.

AWS DMS agrupa los eventos en categorías a las que puede suscribirse para recibir una notificación cada vez que se produzca un evento en esa categoría. Por ejemplo, si se suscribe a la categoría de creación de una instancia de replicación determinada, recibirá una notificación cada vez que se produzca un evento relacionado con la creación y que afecte a su instancia de replicación. Si se suscribe a una categoría de cambios de configuración de una instancia de replicación, recibirá una notificación cada vez que cambie la configuración de la instancia de replicación. También recibirá una notificación cuando cambie una suscripción de notificación de eventos. Para obtener una lista de las categorías de evento proporcionada por AWS DMS, consulte [Categorías y mensajes de eventos de AWS DMS \(p. 344\)](#) más adelante.

AWS DMS envía notificaciones de eventos a las direcciones que proporcione al crear un evento de suscripción. Es posible que le interese crear diferentes suscripciones, como una suscripción que reciba todas las notificaciones de eventos y otra suscripción que incluya únicamente los eventos esenciales para sus recursos DMS de producción. Puede desactivar la notificación fácilmente sin eliminar una suscripción configurando la opción Enabled (Habilitada) en No en la consola de AWS DMS o el parámetro `Enabled` en false mediante la API de AWS DMS.

## Note

Las notificaciones de eventos de AWS DMS que utilizan mensajes de texto SMS están disponibles actualmente para los recursos de AWS DMS en todas las regiones donde se admite AWS DMS. Para obtener más información acerca de cómo utilizar mensajes de texto con SNS, consulte [Envío y recepción de notificaciones de SMS mediante Amazon SNS](#).

AWS DMS utiliza un identificador de suscripción para identificar cada suscripción. Puede publicar varias suscripciones a eventos de AWS DMS en el mismo tema de Amazon SNS. Al utilizar la notificación de eventos, se aplican las tarifas de Amazon SNS. Para obtener más información sobre la facturación de Amazon SNS, consulte [Precios de Amazon SNS](#).

Para suscribirse a eventos AWS DMS, utilice el siguiente proceso:

1. Creación de un tema de Amazon SNS. En el tema, especifique el tipo de notificación que desea recibir y a qué dirección o número se envía la notificación.
2. Cree una suscripción a notificaciones de eventos de AWS DMS mediante la Consola de administración de AWS, AWS CLI o API de AWS DMS.
3. AWS DMS envía un mensaje de correo electrónico o SMS de aprobación a las direcciones que envió con la suscripción. Para confirmar la suscripción, haga clic en el enlace del correo electrónico o mensaje SMS de aprobación.
4. Cuando confirme la suscripción, el estado de la suscripción se actualizará en la sección Event Subscriptions (Suscripciones a eventos) de la consola de AWS DMS.
5. Seguidamente, empezará a recibir notificaciones de eventos.

Para conocer la lista de categorías y eventos que puede recibir en las notificaciones, consulte la siguiente sección. Para obtener más información sobre cómo suscribirse y trabajar con las suscripciones a eventos de AWS DMS, consulte [Suscripción a notificaciones de eventos de AWS DMS \(p. 346\)](#).

## Categorías y mensajes de eventos de AWS DMS

AWS DMS genera un número significativo de eventos en categorías a las que puede suscribirse a través de la consola de AWS DMS o la API de AWS DMS. Cada categoría se aplica a un tipo de origen. Actualmente, AWS DMS es compatible con los tipos de origen siguientes: instancia de replicación y tarea de replicación.

En la siguiente tabla se desglosan las posibles categorías y eventos del tipo de origen instancia de replicación.

Categoría	ID de evento de DMS	Descripción
Cambio de configuración	DMS-EVENT-0012	REP_INSTANCE_CLASS_CHANGING: se está modificando la clase de esta instancia de replicación.
Cambio de configuración	DMS-EVENT-0014	REP_INSTANCE_CLASS_CHANGE_COMPLETE: se ha modificado la clase de esta instancia de replicación.
Cambio de configuración	DMS-EVENT-0018	BEGIN_SCALE_STORAGE: se está incrementando el almacenamiento de la instancia de replicación.
Cambio de configuración	DMS-EVENT-0017	FINISH_SCALE_STORAGE: se ha incrementado el almacenamiento de la instancia de replicación.
Cambio de configuración	DMS-EVENT-0024	BEGIN_CONVERSION_TO_HIGH_AVAILABILITY: la instancia de replicación se está transformando en una configuración Multi-AZ.
Cambio de configuración	DMS-EVENT-0025	FINISH_CONVERSION_TO_HIGH_AVAILABILITY: la instancia de replicación se ha transformado en una configuración Multi-AZ.
Cambio de configuración	DMS-EVENT-0030	BEGIN_CONVERSION_TO_NON_HIGH_AVAILABILITY: la instancia de replicación está evolucionando hacia una configuración Single-AZ.
Cambio de configuración	DMS-EVENT-0029	FINISH_CONVERSION_TO_NON_HIGH_AVAILABILITY: la instancia de replicación se ha transformado en una configuración Single-AZ.
Creación	DMS-EVENT-0067	CREATING_REPLICATION_INSTANCE: se está creando una instancia de replicación.
Creación	DMS-EVENT-0005	CREATED_REPLICATION_INSTANCE: se ha creado una instancia de replicación.
Eliminación	DMS-EVENT-0066	DELETING_REPLICATION_INSTANCE: se está eliminando la instancia de replicación.
Eliminación	DMS-EVENT-0003	DELETED_REPLICATION_INSTANCE: se ha eliminado la instancia de replicación.

Categoría	ID de evento de DMS	Descripción
Mantenimiento	DMS-EVENT-0047	FINISH_PATCH_INSTANCE: se ha actualizado el software de mantenimiento en la instancia de replicación.
Mantenimiento	DMS-EVENT-0026	BEGIN_PATCH_OFFLINE: se está realizando el mantenimiento sin conexión de la instancia de replicación. La instancia de replicación no está disponible en este momento.
Mantenimiento	DMS-EVENT-0027	FINISH_PATCH_OFFLINE: se ha completado el mantenimiento sin conexión de la instancia de replicación. La instancia de replicación ya está disponible.
Mantenimiento	DMS-EVENT-0068	CANNOT_PATCH_INSTANCE: la instancia de replicación está en un estado que no se puede actualizar.
Poco almacenamiento disponible	DMS-EVENT-0007	LOW_STORAGE: queda poco almacenamiento gratuito disponible para la instancia de replicación.
Comutación por error	DMS-EVENT-0013	FAILOVER_STARTED: se ha iniciado la comutación por error para una instancia de replicación Multi-AZ.
Comutación por error	DMS-EVENT-0049	FAILOVER_COMPLETED: se ha completado la comutación por error para una instancia de replicación Multi-AZ.
Comutación por error	DMS-EVENT-0015	HM_SECONDARY_PROMOTION_COMPLETE: comutación por error Multi-AZ a espera completa.
Comutación por error	DMS-EVENT-0050	MAZ_INSTANCE_ACTIVATION_STARTED: ha comenzado la activación Multi-AZ.
Comutación por error	DMS-EVENT-0051	MAZ_INSTANCE_ACTIVATION_COMPLETED: se ha completado la activación Multi-AZ.
Comutación por error	DMS-EVENT-0034	FAILOVER_RECENTLY_OCCURRED: si solicita una comutación por error con demasiada frecuencia, se produce este evento en lugar de los eventos de comutación por error normales.
Error	DMS-EVENT-0031	REPLICATION_INSTANCE_FAILURE: la instancia de replicación ha sufrido un error de almacenamiento.
Error	DMS-EVENT-0036	INCOMPATIBLE_NETWORK: la instancia de replicación ha fallado debido a una incompatibilidad de red.
Error	DMS-EVENT-0037	INACCESSIBLE_ENCRYPTION_CREDS: cuando el servicio no puede acceder a la clave de KMS utilizada para cifrar el volumen de datos.

En la siguiente tabla se desglosan las posibles categorías y eventos del tipo de origen tarea de replicación.

Categoría	ID de evento de DMS	Descripción
Cambio de estado	DMS-EVENT-0069	REPLICATION_TASK_STARTED: se ha iniciado la tarea de replicación.
Cambio de estado	DMS-EVENT-0077	REPLICATION_TASK_CDC_STARTED: la tarea de replicación ha iniciado CDC.
Cambio de estado	DMS-EVENT-0081	RELOAD_TABLES_EVENT: se ha solicitado la recarga de los detalles de tabla.
Cambio de estado	DMS-EVENT-0079	REPLICATION_TASK_STOPPED: se ha detenido la tarea de replicación.
Error	DMS-EVENT-0078	REPLICATION_TASK_FAILED: se ha producido un error con una tarea de replicación.
Error	DMS-EVENT-0082	CLEAN_TASK_FAILED: una llamada a limpiar datos de tarea ha devuelto un error.
ConfigurationChange	DMS-EVENT-0080	REPLICATION_TASK_MODIFIED: se ha modificado una tarea de replicación.
Eliminación	DMS-EVENT-0073	REPLICATION_TASK_DELETED: se ha eliminado la tarea de replicación.
Creación	DMS-EVENT-0074	REPLICATION_TASK_CREATED: se ha creado la tarea de replicación.

## Suscripción a notificaciones de eventos de AWS DMS

Puede crear una suscripción de notificación de eventos de AWS DMS para recibir notificaciones cada vez que se produzca un evento de AWS DMS. La forma más sencilla de crear una suscripción es con la consola de AWS DMS. Si decide crear las suscripciones de notificación de eventos mediante la API de AWS DMS, deberá crear un tema de Amazon SNS y suscribirse a dicho tema con la consola o la API de Amazon SNS. En este caso, también deberá tener en cuenta el nombre de recurso de Amazon (ARN) del tema porque ese ARN se utiliza al enviar comandos de CLI o acciones de la API. Para obtener información acerca de cómo crear un tema de Amazon SNS y suscribirse a él, consulte [Introducción a Amazon SNS](#).

En una suscripción de notificación puede especificar el tipo de origen sobre el que desea recibir notificaciones y el origen de AWS DMS que desencadena el evento. Puede definir el tipo de origen de AWS DMS con un valor `SourceType`. Puede definir el origen que genera el evento mediante un valor `SourceIdentifier`. Si especifica `SourceType` y `SourceIdentifier`, por ejemplo, `SourceType = db-instance` y `SourceIdentifier = myDBInstance1`, recibirá todos los eventos DB\_Instance de un origen especificado. Si especifica `SourceType`, pero no especifica `SourceIdentifier`, recibirá una notificación de los eventos de ese tipo de origen para todos sus orígenes de AWS DMS. Si no especifica ni `SourceType` ni `SourceIdentifier`, se le notificarán los eventos generados en todos los orígenes de AWS DMS que pertenezcan a su cuenta de cliente.

## Consola de administración de AWS

Para suscribirse a la notificación de eventos de AWS DMS mediante la consola

1. Inicie sesión en la consola de administración de AWS y seleccione AWS DMS. Tenga en cuenta que si ha iniciado sesión como usuario de AWS Identity and Access Management (IAM), será preciso que disponga de los permisos adecuados para tener acceso a AWS DMS.
2. En el panel de navegación seleccione Event Subscriptions.
3. En la página Event Subscriptions seleccione Create Event Subscription.
4. En la página Create Event Subscription haga lo siguiente:
  - a. En Name escriba un nombre para la suscripción de notificación de evento.
  - b. Elija un tema Amazon SNS existente para Send notifications to (Enviar notificaciones a) o elija create topic (crear tema). Debe disponer de un tema de Amazon SNS al que enviar las notificaciones o si no crear el tema. Si elige create topic, puede especificar una dirección de correo electrónico a la que se enviarán las notificaciones.
  - c. En Source Type elija un tipo de origen. La única opción es replication instance.
  - d. Elija Yes para activar la suscripción. Si desea crear la suscripción, pero que todavía no envíen notificaciones, seleccione No.
  - e. En función del tipo de origen que haya seleccionado, seleccione las categorías y orígenes del evento de las que desea recibir notificaciones.

Create Event Subscription

The screenshot shows the 'Create Event Subscription' form. The fields filled in are:

- Name\*: DMS-subscription
- Topic name\*: DMS-event-topic
- Recipient type\*: Email
- With these recipients\*: sgrayson@amazon.com
- Source Type\*: Replication Instance
- Enabled\*: Yes (radio button selected)

Below these fields are two sections for 'Event Categories':

- Left section: Radio buttons for 'Select All' (selected) and 'Select Specific'. A large empty rectangular box follows.
- Right section: Radio buttons for 'Select All' (selected) and 'Select Specific'. A large empty rectangular box follows.

- f. Seleccione Create.

La consola de AWS DMS indica que se está creando la suscripción.

## API de AWS DMS

Para suscribirse a la notificación de eventos de AWS DMS mediante la API de AWS DMS

- Llame a [CreateEventSubscription](#).

# Migración de grandes almacenes de datos con AWS Database Migration Service y AWS Snowball Edge

Las migraciones de datos más grandes pueden incluir muchos terabytes de información. Este proceso puede ser engorroso debido a los límites de ancho de banda de red o a la gran cantidad de datos. AWS Database Migration Service (AWS DMS) puede utilizar [AWS Snowball Edge](#) y Amazon S3 para migrar grandes bases de datos más rápidamente que otros métodos.

AWS Snowball Edge es un servicio de AWS que proporciona un dispositivo Edge que puede usar para transferir datos a la nube a velocidades superiores a las de la red. Un dispositivo Edge es un dispositivo propiedad de AWS. Puede albergar hasta 100 TB de datos. Usa el cifrado de 256 bits y un Módulo de plataforma segura (TPM) estándar del sector diseñado para garantizar la seguridad y la cadena de custodia completa de los datos. AWS Snowball Edge ofrece muchas características adicionales; para obtener más información, consulte [¿Qué es un AWS Snowball Edge?](#) en la Guía para desarrolladores de AWS Snowball Edge.

Amazon S3 es un servicio de almacenamiento y de recuperación de AWS. Para almacenar un objeto en Amazon S3, debe cargar en un bucket el archivo que quiera almacenar. Al cargar un archivo, puede configurar permisos para el objeto y también para cualquier metadato. Para obtener más información, consulte la [documentación de S3](#).

Cuando se utiliza un dispositivo Edge, el proceso de migración de datos consta de las siguientes fases:

1. Puede utilizar el Herramienta de conversión de esquemas de AWS (AWS SCT) para extraer los datos localmente y transferirlos a un dispositivo Edge.
2. Envía el dispositivo o dispositivos Edge de vuelta a AWS.
3. Después de que AWS recibe su envío, el dispositivo Edge carga automáticamente sus datos en un bucket de Amazon S3.
4. AWS DMS toma los archivos y migra los datos al almacén de datos de destino. Si utiliza la captura de datos de cambio (CDC), esas actualizaciones se escriben en el bucket de Amazon S3 y, a continuación, se aplican al almacén de datos de destino.

En las secciones siguientes, puede obtener más información sobre cómo usar un dispositivo Edge para migrar bases de datos relacionales con AWS SCT y AWS DMS. También puede utilizar un dispositivo Edge y AWS SCT para migrar almacenes de datos locales a la nube de AWS. Para obtener más información acerca de las migraciones de almacén de datos, consulte [Migración de datos desde un almacén de datos local a Amazon Redshift](#) en la Guía del usuario de AWS Schema Conversion Tool.

## Temas

- [Información general sobre migración de grandes almacenes de datos con DMS y Snowball Edge \(p. 349\)](#)
- [Requisitos previos para la migración de grandes almacenes de datos con DMS y Snowball Edge \(p. 350\)](#)
- [Lista de comprobación de migración \(p. 350\)](#)
- [Procedimientos paso a paso para migrar datos utilizando AWS DMS con Snowball Edge \(p. 352\)](#)
- [Limitaciones cuando se trabaja con Snowball Edge y AWS DMS \(p. 368\)](#)

# Información general sobre migración de grandes almacenes de datos con DMS y Snowball Edge

El proceso de utilización de AWS DMS y AWS Snowball Edge incorpora tanto aplicaciones locales como servicios administrados por Amazon. Usamos los términos local y remoto para distinguir estos componentes.

Los componentes locales incluyen lo siguiente:

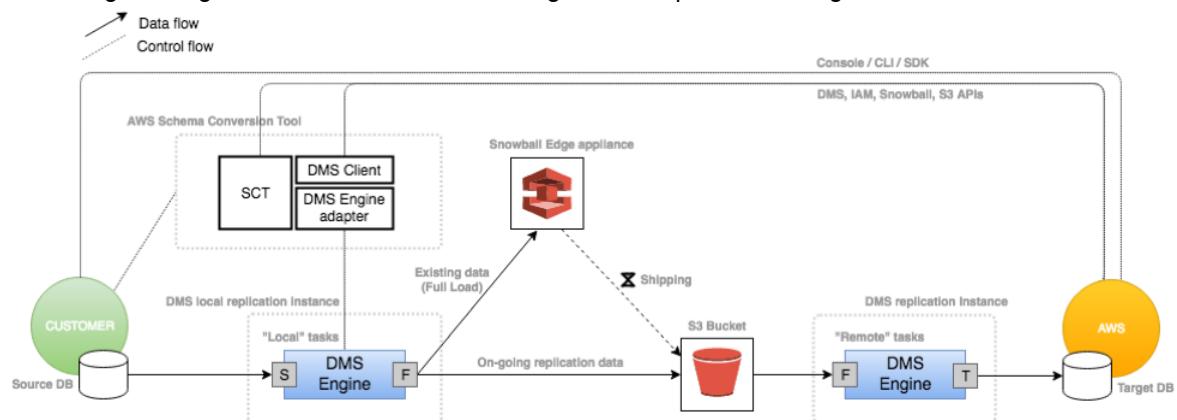
- AWS SCT
- Agente de AWS DMS (una versión local de AWS DMS que funciona localmente)
- dispositivos Snowball Edge

Los componentes remotos incluyen lo siguiente:

- Amazon S3
- AWS DMS

En las secciones siguientes, puede encontrar una guía paso a paso para configurar, instalar y administrar una migración de AWS DMS mediante un dispositivo o dispositivos Edge.

En el diagrama siguiente se muestra información general del proceso de migración.



La migración implica una tarea local, en la que traslada los datos a un dispositivo Edge mediante el agente de DMS. Una vez que se ha cargado un dispositivo Edge, lo devuelve a AWS. Si tiene varios dispositivos Edge, puede devolverlos al mismo tiempo o de forma secuencial. Cuando AWS recibe un dispositivo Edge, una tarea remota que utiliza AWS DMS carga los datos en el almacenamiento de datos de destino en AWS.

Para migrar desde un almacén de datos local a un almacén de datos de AWS, siga estos pasos:

1. Utilice la Consola de administración de AWS Snowball para crear un nuevo trabajo para importar datos a S3 con un dispositivo Snowball Edge Storage Optimized. El trabajo implica solicitar que se envíe a su dirección el dispositivo.
2. Configure AWS SCT en un equipo local que pueda acceder a AWS. Instale la herramienta cliente de Snowball Edge en un equipo local distinto.
3. Cuando llegue el dispositivo Edge, enciéndalo, cóncéctelo y, a continuación, desbloquéelo con la herramienta de cliente. Para obtener información paso a paso, consulte [Introducción a AWS Snowball Edge](#) en la Guía para desarrolladores de AWS Snowball Edge.

4. Instale los controladores de Open Database Connectivity (ODBC) para sus orígenes de datos. Póngalos en la máquina con la herramienta cliente de Edge.
5. Instale y configure el host de AWS DMS Agent en la máquina con la herramienta cliente de Edge.

El agente de DMS debe tener conectividad a la base de datos de origen, AWS SCT, AWS y el Snowball Edge. El agente de DMS se admite solo en las siguientes plataformas Linux:

- Red Hat Enterprise Linux versiones 6.2 a 6.8, 7.0 y 7.1 (64 bits)
- SUSE Linux versión 12 (64 bits)

Aunque el agente de DMS está disponible en el paquete de instalación de AWS SCT, es mejor que no se encuentre en el mismo lugar. Le recomendamos que instale el agente de DMS en otro equipo (no en la máquina donde ha instalado AWS SCT).

6. Cree un proyecto nuevo en AWS SCT.
7. Configure AWS SCT para utilizar el dispositivo Snowball Edge.
8. Registre el agente de DMS con AWS SCT.
9. Cree una tarea local y de DMS en SCT.
10. Ejecute y monitorice la tarea en SCT.

## Requisitos previos para la migración de grandes almacenes de datos con DMS y Snowball Edge

Antes de iniciar el proceso de migración, necesita los siguientes requisitos previos:

- Está familiarizado con el funcionamiento básico de AWS SCT.
- Ha creado o puede crear el bucket o buckets de S3 que va a utilizar en la migración.
- Tiene una instancia de replicación de AWS DMS en la misma región que el bucket de S3.
- Está familiarizado con el uso de la AWS Command Line Interface (AWS CLI).
- Está familiarizado con la [Guía para desarrolladores de Snowball Edge](#).

## Lista de comprobación de migración

Para facilitar las cosas durante la migración, puede utilizar la siguiente lista de comprobación para crear una lista de los elementos que necesita durante la migración.

DMS Migration Checklist

This checklist is for my schemas named:

The database engine that my schemas reside on is:

AWS Region for the migration:

Name of migration job that you created in the AWS Snowball Management Console:

S3 bucket (and folder) for this job:

IAM role that has access to the S3 Bucket and the target database on AWS:

-----  
Path to the installation directory of AWS SCT (needed for a future step):

Name/IP of Machine #1 (SCT):

Name/IP of Machine #2 (Connectivity):

  
-----

IP address of your Snowball Edge:

Port for the Snowball Edge:

Unlock code for the Snowball Edge device:

Path to the manifest file:

Output of the command snowballEdge get-secret-access-key:

    AWS access key ID:

    AWS secret access Key:

  
-----

Confirm ODBC drivers is installed on Machine #2 (Connectivity):

  
-----

Confirm DMS Agent is installed on Machine #2 (Connectivity):

Confirm DMS Agent is running two processes:

DMS Agent password:

DMS Agent port number:

Confirm that your firewall allows connectivity:

  
-----

Name of SCT project:

  
-----

Confirm that DMS Agent is registered with SCT:

New agent or service profile name that you provided:

  
-----

Confirm local and DMS task exists:

Task name that you provided:

  
-----

Confirm:

DMS Agent connects to the following:

- \_\_ The source database
- \_\_ The staging S3 bucket
- \_\_ The Edge device

DMS task connects to the following:

- \_\_ The staging S3 bucket

— The target database on AWS

Confirm the following:

- Stopped Edge client
- Powered off Edge device
- Returned Edge device to AWS

## Procedimientos paso a paso para migrar datos utilizando AWS DMS con Snowball Edge

En las secciones siguientes puede encontrar información detallada sobre los pasos para la migración.

### Paso 1: Crear un trabajo de Snowball Edge

Siga los pasos que se describen en la sección [Introducción a un dispositivo Snowball Edge](#) en la Guía para desarrolladores de AWS Snowball Edge. Abra la Consola de administración de AWS Snowball y cree un nuevo trabajo para Import into Amazon S3 (Importar en Amazon S3).

Asegúrese de solicitar un dispositivo Snowball Edge (Snowball Edge Storage Optimized), ya que los dispositivos Snowball normales no son compatibles para AWS DMS. Siga las indicaciones en pantalla para resto de la configuración. Tiene la oportunidad de revisar la configuración antes de crear el trabajo.

### Paso 2: Descargar e instalar la Herramienta de conversión de esquemas de AWS (AWS SCT)

Necesita dos equipos locales para ejecutar este proceso, además del dispositivo Edge.

Descargue la aplicación Herramienta de conversión de esquemas de AWS e instálela en un equipo local que puede obtener acceso a AWS. Para obtener instrucciones, que incluyen información sobre sistemas operativos compatibles, consulte [Instalación y actualización de AWS Schema Conversion Tool](#).

En otro equipo, donde desea instalar el agente de DMS, descargue e instale el cliente de Snowball Edge de [Recursos de AWS Snowball Edge](#).

Una vez que haya terminado este paso, debe tener dos máquinas:

- Máquina n.º 1 (SCT), con AWS SCT instalado
- Máquina n.º 2 (Conectividad), con el cliente de Edge, donde desea instalar el agente de DMS y los controladores ODBC para las bases de datos que va a migrar

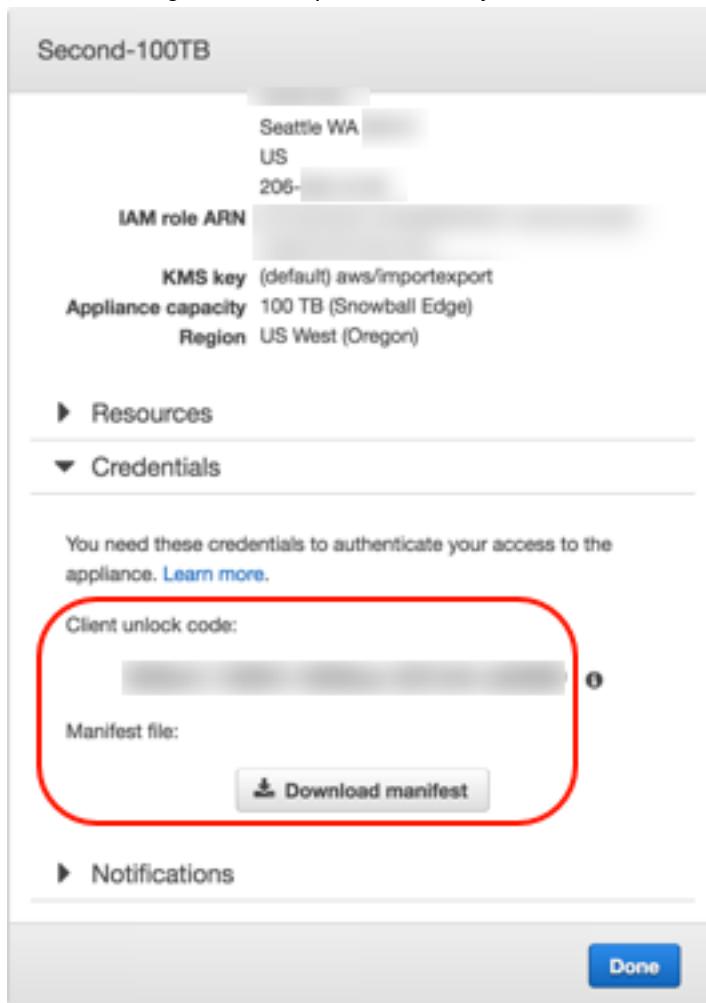
### Paso 3: Desbloquear el dispositivo Snowball Edge

Cuando llegue el dispositivo Edge, debe prepararlo para su uso.

Siga los pasos que se describen en la sección [Introducción a un dispositivo AWS Snowball Edge](#) en la Guía para desarrolladores de AWS Snowball Edge.

También puede consultar la visita guiada [¿Cómo desbloqueo un AWS Snowball Edge?](#) de AWS Support, o consultar la página de marketing [Introducción a AWS Snowball Edge](#) para obtener más recursos.

Encienda el dispositivo, [conéctelo a su red local](#), registre la dirección IP del dispositivo Edge y [obtenga el código de desbloqueo y el archivo de manifiesto](#) de la consola de Snowball Edge. En la consola, elija el trabajo, seleccione View job details (Ver detalles del trabajo) y, a continuación, Credentials (Credenciales). Guarde el código de desbloqueo de cliente y el archivo de manifiesto.



En la pantalla del dispositivo Edge, obtenga la IP del dispositivo Edge en la pestaña Connection (Conexión). A continuación, desbloquee el dispositivo mediante el comando `snowballEdge unlock` con la información de IP y de las credenciales. El ejemplo siguiente muestra la sintaxis de muestra para este comando.

```
snowballEdge unlock -i IP_Address -m Local_path_to_manifest_file -  
u 29_character_unlock_code
```

A continuación se incluye un comando de ejemplo.

```
snowballEdge unlock \  
-i 192.0.2.0 \  
-m /Downloads/JID2EXAMPLE-0c40-49a7-9f53-916aEXAMPLE81-manifest.bin \  
-u 12345-abcd-e12345-ABCDE-12345
```

Por último, [recupera la clave de acceso y la clave secreta de Snowball](#) del dispositivo utilizando el cliente de Edge. A continuación se muestra la entrada y salida de ejemplo del comando para obtener la clave de acceso.

Ejemplo de entrada

```
snowballEdge list-access-keys \
--endpoint https://192.0.2.0 \
--manifest-file Path_to_manifest_file \
--unlock-code 12345-abcde-12345-ABCDE-12345
```

Ejemplo de resultados

```
{ \
    "AccessKeyId" : [ "AKIAIOSFODNN7EXAMPLE" ] \
}
```

A continuación se muestra la entrada y salida de ejemplo del comando para obtener la clave secreta.

Ejemplo de entrada

```
snowballEdge get-secret-access-key \
--access-key-id AKIAIOSFODNN7EXAMPLE \
--endpoint https://192.0.2.0 \
--manifest-file /Downloads/JID2EXAMPLE-0c40-49a7-9f53-916aEXAMPLE81-manifest.bin \
--unlock-code 12345-abcde-12345-ABCDE-12345
```

Ejemplo de resultados

```
[snowballEdge]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

Cuando el Snowball Edge está listo para usar, puede interactuar con él directamente utilizando la AWS CLI o el SDK de S3 Adapter para Snowball. Este adaptador también funciona con el dispositivo Edge.

## Paso 4: Configurar el host de agente de DMS con los controladores ODBC

Utilizando la Máquina n.º 2 (Conectividad) del paso 2, donde el cliente de Edge ya está instalado, instale los controladores ODBC necesarios. Estos controladores son necesarios para conectarse a la base de datos de origen. El controlador necesario varía en función del motor de base de datos. En las secciones siguientes, puede encontrar información para cada motor de base de datos.

Temas

- [Oracle \(p. 354\)](#)
- [Microsoft SQL Server \(p. 355\)](#)
- [ASE SAP Sybase \(p. 355\)](#)
- [MySQL \(p. 355\)](#)
- [PostgreSQL \(p. 355\)](#)

### Oracle

Instale Oracle Instant Client para Linux (x86-64) versión 11.2.0.3.0 o posterior.

Además, si no se ha incluido todavía en el sistema, debe crear un enlace simbólico en el `$ORACLE_HOME\lib` directory. Este enlace debe denominarse `libclntsh.so` y debería apuntar a una versión específica de este archivo. Por ejemplo, en un cliente de Oracle 12c se utiliza lo siguiente.

```
lrwxrwxrwx 1 oracle oracle 63 Oct 2 14:16 libclntsh.so ->/u01/app/oracle/home/lib/
libclntsh.so.12.1
```

Además, la variable de entorno `LD_LIBRARY_PATH` debe anexarse con el directorio lib de Oracle y añadirse al script `site_arep_login.sh` bajo la carpeta lib de la instalación. Añada este script si no existe.

```
vi /opt/amazon/aws-schema-conversion-tool-dms-agent/bin/site_arep_login.sh

export ORACLE_HOME=/usr/lib/oracle/12.2/client64;
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib
```

## Microsoft SQL Server

Instalar el controlador ODBC para Microsoft.

Actualice el script de `site_arep_login.sh` con el siguiente código.

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql/lib64/
```

## ASE SAP Sybase

El cliente ODBC para SAP Sybase ASE de 64 bits debe estar instalado.

Si el directorio de instalación es `/opt/sap`, actualice el script `site_arep_login.sh` con lo siguiente.

```
export SYBASE_HOME=/opt/sap
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$SYBASE_HOME/DataAccess64/ODBC/lib:$SYBASE_HOME/
DataAccess/ODBC/lib:$SYBASE_HOME/OCS-16_0/lib:$SYBASE_HOME/OCS-16_0/lib3p64:$SYBASE_HOME/
OCS-16_0/lib3p
```

El archivo `/etc/odbcinst.ini` debe incluir las entradas siguientes.

```
[Sybase]
Driver=/opt/sap/DataAccess64/ODBC/lib/libsybdbrvodb.so
Description=Sybase ODBC driver
```

## MySQL

Instale MySQL Connector/ODBC para Linux, versión 5.2.6 o posterior.

Asegúrese de que el archivo `/etc/odbcinst.ini` contiene una entrada para MySQL, como en el siguiente ejemplo.

```
[MySQL ODBC 5.2.6 Unicode Driver]
Driver = /usr/lib64/libmyodbc5w.so
UsageCount = 1
```

## PostgreSQL

Install `postgresql94-9.4.4-1PGDG.<OS Version>.x86_64.rpm`. Este paquete contiene el ejecutable de `psql`. Por ejemplo, `postgresql94-9.4.4-1PGDG.rhel7.x86_64.rpm` es el paquete necesario para Red Hat 7.

Instale el controlador de ODBC `postgresql-09.03.0400-1PGDG.<OS version>.x86_64` o una versión superior para Linux, donde <versión SO> es el SO de la máquina del agente. Por ejemplo, `postgresql-09.03.0400-1PGDG.rhel7.x86_64` es el cliente necesario para Red Hat 7.

Asegúrese de que el archivo `/etc/odbcinst.ini` contiene una entrada para PostgreSQL, como en el siguiente ejemplo.

```
[PostgreSQL]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-9.4/lib/psqlodbc.so
Setup = /usr/pgsql-9.4/lib/psqlodbcw.so
Debug = 0
CommLog = 1
UsageCount = 2
```

## Paso 5: Instalar el agente de DMS

Utilizando la Máquina n.º 2 (Conectividad) del paso 2, donde el cliente de Edge y los controladores ODBC ya están instalados, instale y configure el agente de DMS. El agente de DMS se suministra como parte del paquete de instalación de AWS SCT, que se describe en la Guía del usuario de AWS Schema Conversion Tool.

Una vez que haya terminado este paso, debe tener dos máquinas locales preparadas:

- Máquina n.º 1 (SCT) con AWS SCT instalado
- Máquina n.º 2 (Conectividad) con el cliente de Edge, los controladores ODBC y el agente de DMS instalados

### Instalación del agente de DMS

1. En el directorio de instalación de AWS SCT, localice el archivo de RPM denominado `aws-schema-conversion-tool-dms-agent-2.4.1-R1.x86_64.rpm`.

Cópielo en Máquina n.º 2 (Conectividad), la máquina del agente de DMS. SCT y el agente de DMS se deben instalar en máquinas independientes. El agente de DMS debe estar ubicado en el mismo equipo que el cliente de Edge y los controladores ODBC.

2. En la Máquina n.º 2 (Conectividad), ejecute el siguiente comando para instalar el agente de DMS. Para simplificar los permisos, ejecute este comando como usuario `root`.

```
sudo rpm -i aws-schema-conversion-tool-dms-agent-2.4.0-R2.x86_64.rpm
```

Este comando utiliza la ubicación de instalación predeterminada de `/opt/amazon/aws-schema-conversion-tool-dms-agent`. Para instalar el agente de DMS en otra ubicación, utilice la siguiente opción.

```
sudo rpm --prefix <installation_directory> -i aws-schema-conversion-tool-dms-agent-2.4.0-R2.x86_64.rpm
```

3. Para verificar que el agente de DMS está funcionando, utilice el comando siguiente.

```
ps -ef | grep repctl
```

La salida de este comando debería mostrar dos procesos en ejecución.

Para configurar el agente de DMS, debe proporcionar una contraseña y número de puerto. La contraseña se utiliza más adelante para registrar el agente de DMS con AWS SCT, por tanto guárdela

a mano. Elija un número de puerto sin utilizar para que el agente de DMS escuche conexiones de AWS SCT. Es posible que tenga que configurar su firewall para permitir la conectividad.

Ahora configure el agente de DMS mediante el script `configure.sh`.

```
sudo /opt/amazon/aws-schema-conversion-tool-dms-agent/bin/configure.sh
```

Aparece la siguiente solicitud. Introduzca la contraseña. Cuando se le solicite, introduzca de nuevo la contraseña para confirmarla.

```
Configure the AWS Schema Conversion Tool DMS Agent server
Note: you will use these parameters when configuring agent in AWS Schema
      Conversion Tool

Please provide password for the server
Use minimum 8 and up to 20 alphanumeric characters with at least one digit and one
      capital case character

Password:
```

El resultado es el siguiente. Proporcione un número de puerto.

```
chown: missing operand after 'amazon:amazon'
Try 'chown --help' for more information.
/opt/amazon/aws-schema-conversion-tool-dms-agent/bin/repctl:
    /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libcom_err.so.3: no version
      information available (required by
        /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libgssapi_krb5.so.2)
/opt/amazon/aws-schema-conversion-tool-dms-agent/bin/repctl:
    /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libcom_err.so.3: no version
      information available (required by
        /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libkrb5.so.3)
[setserverpassword command] Succeeded

Please provide port number the server will listen on (default is 3554)
Note: you will have to configure your firewall rules accordingly
Port:
```

El resultado es el siguiente, confirme que el servicio se inicia.

```
Starting service...
/opt/amazon/aws-schema-conversion-tool-dms-agent/bin/repctl:
    /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libcom_err.so.3: no version
      information available (required by
        /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libgssapi_krb5.so.2)
/opt/amazon/aws-schema-conversion-tool-dms-agent/bin/repctl:
    /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libcom_err.so.3: no version
      information available (required by
        /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libkrb5.so.3)
AWS Schema Conversion Tool DMS Agent was sent a stop signal
AWS Schema Conversion Tool DMS Agent is no longer running
[service command] Succeeded
/opt/amazon/aws-schema-conversion-tool-dms-agent/bin/repctl:
    /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libcom_err.so.3: no version
      information available (required by
        /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libgssapi_krb5.so.2)
/opt/amazon/aws-schema-conversion-tool-dms-agent/bin/repctl:
    /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libcom_err.so.3: no version
      information available (required by
        /opt/amazon/aws-schema-conversion-tool-dms-agent/lib/libkrb5.so.3)
```

```
AWS Schema Conversion Tool DMS Agent was started as PID 1608
```

Recomendamos que la instale la [AWS Command Line Interface \(AWS CLI\)](#). Mediante la AWS CLI, puede interrogar a Snowball Edge para ver los archivos de datos escritos en el dispositivo. Utilice las credenciales de AWS recuperadas de Edge para obtener acceso al dispositivo Edge. Por ejemplo, podría ejecutar el siguiente comando.

```
aws s3 ls --profile SnowballEdge --endpoint https://192.0.2.0:8080 bucket-name --recursive
```

Este comando produce el resultado siguiente.

```
2018-08-20 10:55:31 53074692 streams/load00000001000573E166ACF4C0/00000001.fcd.gz
2018-08-20 11:14:37 53059667 streams/load00000001000573E166ACF4C0/00000002.fcd.gz
2018-08-20 11:31:42 53079181 streams/load00000001000573E166ACF4C0/00000003.fcd.gz
```

Para detener el agente de DMS, ejecute el comando siguiente en el directorio `/opt/amazon/aws-schema-conversion-tool-dms-agent/bin`.

```
./aws-schema-conversion-tool-dms-agent stop
```

Para iniciar el agente de DMS, ejecute el comando siguiente en el directorio `/opt/amazon/aws-schema-conversion-tool-dms-agent/bin`.

```
./aws-schema-conversion-tool-dms-agent start
```

## Paso 6: Crear un nuevo proyecto de AWS SCT

A continuación, crea un nuevo proyecto de AWS SCT que especifica las bases de datos de origen y de destino. Para obtener más información, consulte [Creación de un proyecto de AWS Schema Conversion Tool](#) en la Guía del usuario de AWS Schema Conversion Tool.

Para crear un proyecto nuevo en AWS SCT

1. Inicie AWS SCT y elija File (Archivo) y, a continuación, New Project (Nuevo proyecto). Aparece el cuadro de diálogo New Project (Nuevo proyecto).
2. Agregue la siguiente información del proyecto.

Para este parámetro	Haga lo siguiente
Project Name	Introduzca un nombre para su proyecto, que se almacenará localmente en su equipo.
Ubicación	Introduzca la ubicación del archivo local de su proyecto.
OLTP	Elija Transactional Database (OLTP) [Base de datos transaccional (OLTP)].
Source DB Engine	Elija el almacén de datos de origen.
Target DB Engine	Elija el almacén de datos de destino.

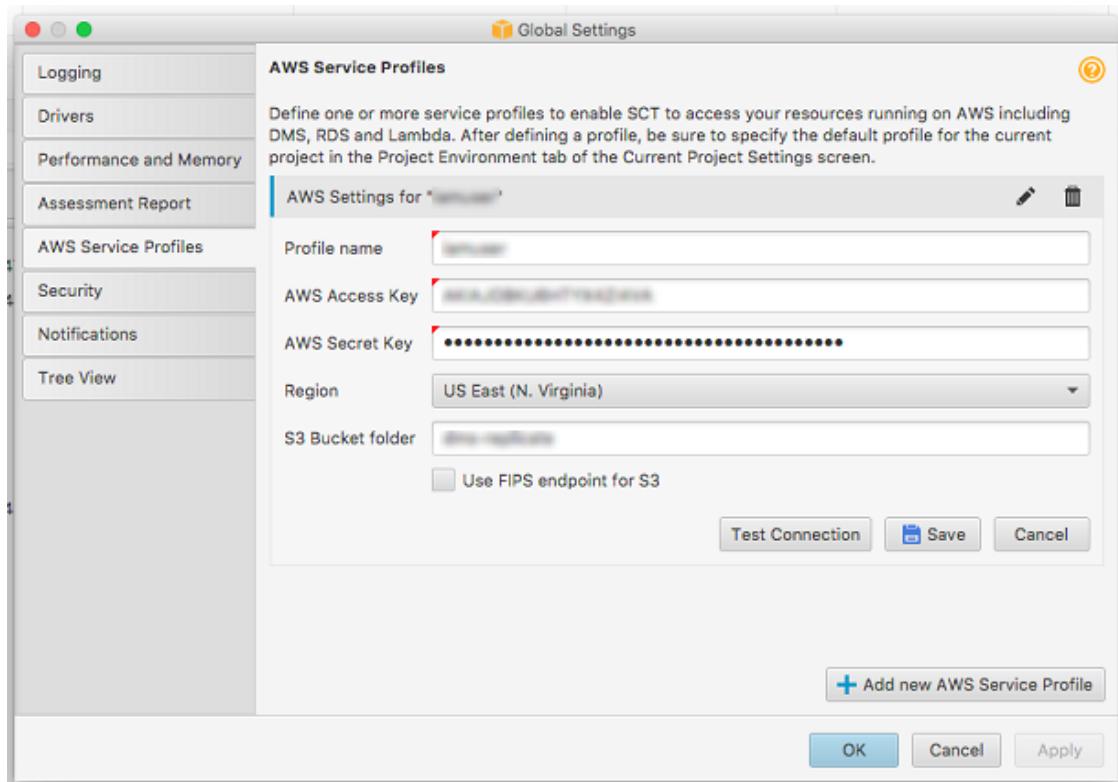
3. Elija OK (Aceptar) para crear su proyecto de AWS SCT.
4. Conéctese a las bases de datos de origen y destino.

## Paso 7: Configure AWS SCT para utilizar el Snowball Edge

El perfil de servicio AWS SCT se debe actualizar para utilizar el agente de DMS, que es un AWS DMS local que funciona en el entorno local.

Para actualizar el perfil de AWS SCT para trabajar con el agente de DMS

1. Start AWS SCT.
2. Seleccione Settings (Configuración), Global Settings (Configuración global), AWS Service Profiles (Perfiles de servicio de AWS).
3. Elija Add New AWS Service Profile (Añadir nuevo perfil de servicio de AWS).



4. Agregue la siguiente información del perfil.

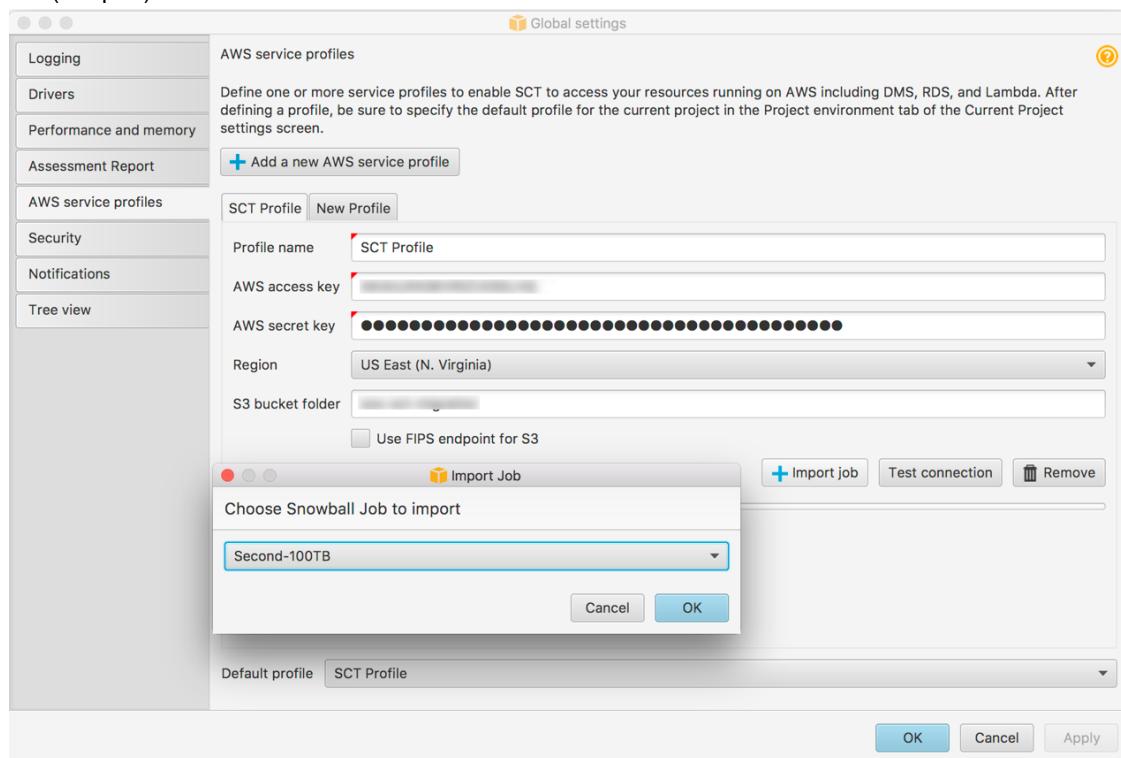
Para este parámetro	Haga lo siguiente
Profile Name	Introduzca un nombre para su proyecto, que se almacenará localmente en su equipo.
Clave de acceso de AWS	Introduzca la clave de acceso de AWS de la cuenta de AWS y región de AWS que pretende utilizar para la migración. Las credenciales suministradas deben tener permisos para obtener acceso al trabajo de Snowball Edge en AWS.
Clave secreta de AWS	Introduzca la clave secreta de AWS de la cuenta de AWS y región de AWS que pretende utilizar para la migración.

Para este parámetro	Haga lo siguiente
Región	Elija la región de AWS de la cuenta que está utilizando. La instancia de replicación de DMS, el bucket de S3 y el almacén de datos de destino deben estar en esta región de AWS.
Carpeta del bucket de S3	Introduzca un nombre para el bucket de S3 que se le asignó al crear el trabajo de Snowball Edge.

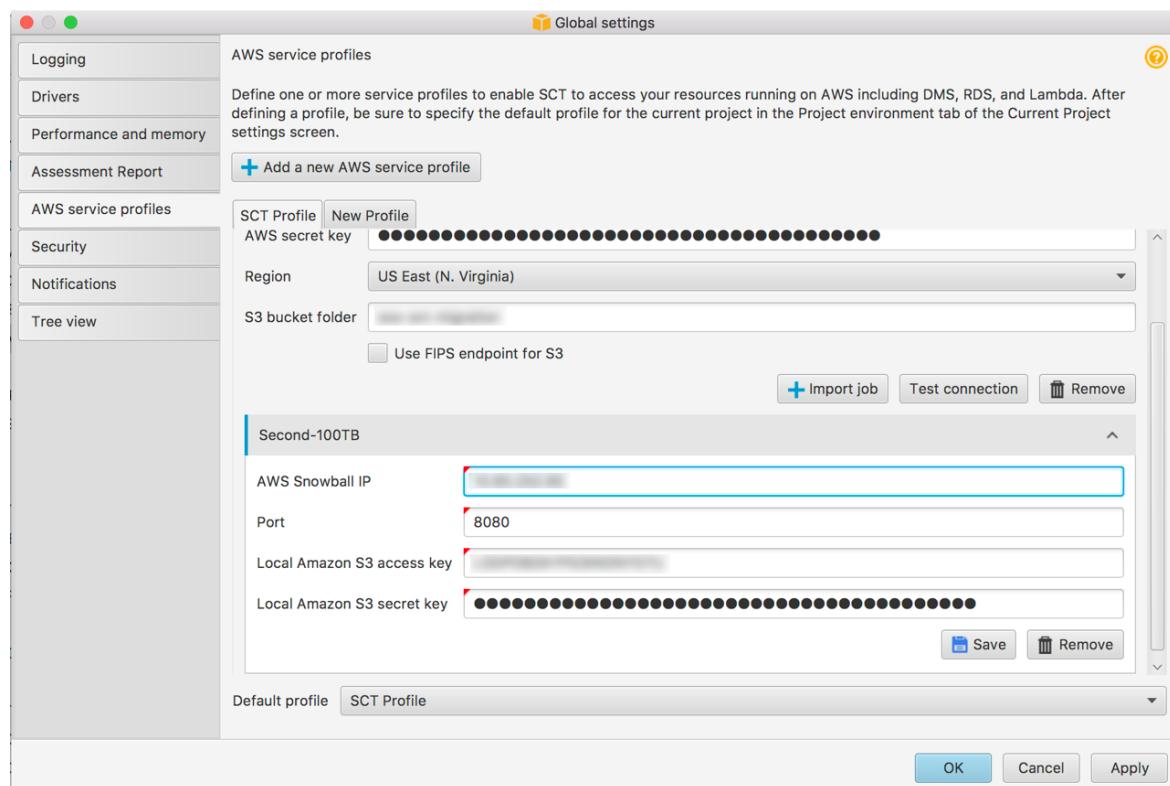
- Una vez que haya escrito la información, elija Test Connection (Conexión de prueba) para verificar que AWS SCT puede conectarse al bucket de Amazon S3.

La sección OLTP Local & DMS Data Migration (OLTP Local y migración de datos de DMS) en la ventana emergente debería mostrar todas las entradas con el estado Pass (Correcto). Si la prueba falla, el error se debe probablemente a que en la cuenta que está usando faltan privilegios para acceder al bucket de Amazon S3.

- Si se supera la prueba, elija OK (Aceptar) y, a continuación, OK (Aceptar) de nuevo para cerrar la ventana y el cuadro de diálogo.
- Elija Import job (Importar trabajo), elija el trabajo de Snowball Edge de la lista y, a continuación, elija OK (Aceptar).



Ahora configure AWS SCT para utilizar Snowball Edge. Introduzca la dirección IP de Snowball Edge, el puerto de escucha en el dispositivo (el valor predeterminado es 8080) y las claves de acceso y las claves secretas de Snowball que recuperó con anterioridad. Seleccione OK (Aceptar) para guardar los cambios.

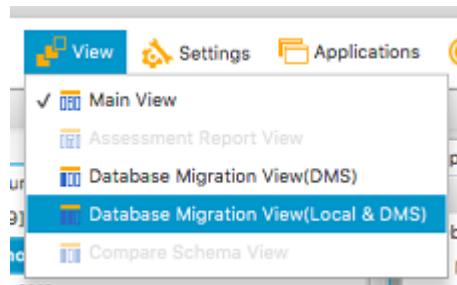


## Paso 8: Registrar el agente de DMS en AWS SCT

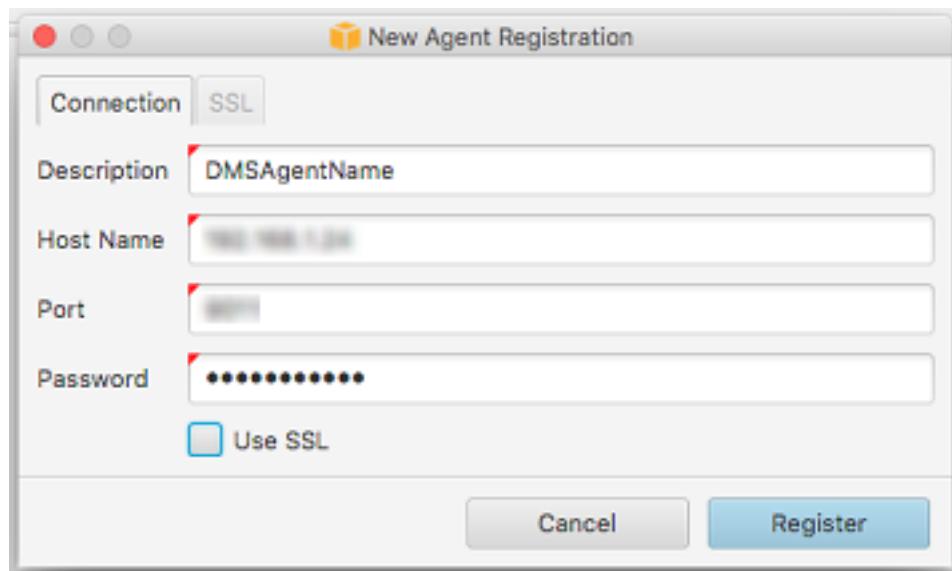
A continuación, registre el agente de DMS en AWS SCT. SCT intenta, a continuación, conectarse al agente, mostrando el estado. Cuando el agente está disponible, el estado cambia a activo.

### Para registrar el agente de DMS

1. Inicie AWS SCT, elija View (Ver) y, a continuación, elija Database Migration View (Local & DMS) [Vista de migración de base de datos (local y DMS)].



2. Seleccione la pestaña Agent (Agente) y, a continuación, elija Register (Registrar). Aparecerá el cuadro de diálogo New Agent Registration (Registro de nuevo agente).



3. Introduzca la información en el cuadro de diálogo New Agent Registration (Registro de nuevo agente).

Para este parámetro	Haga lo siguiente
Description (Descripción)	Escriba el nombre del agente.
Host Name	Introduzca la dirección IP de la máquina en la que instaló el agente de DMS.
Puerto	Introduzca el número de puerto que utilizó cuando configuró el agente de DMS.
Contraseña	Introduzca la contraseña que utilizó cuando configuró el agente de DMS.

4. Elija Register (Registrar) para registrar el agente con su proyecto de AWS SCT.

## Paso 9: Crear una tarea local y de DMS

A continuación, debe crear la tarea de migración integral. Esta tarea incluye dos tareas secundarias:

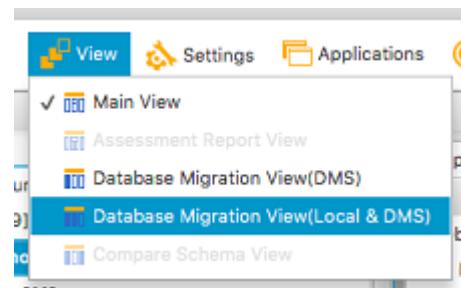
- La tarea secundaria local: esta tarea migra datos desde la base de datos de origen al dispositivo Snowball Edge.
- La tarea secundaria de DMS: esta tarea traslada los datos desde el dispositivo a un bucket de Amazon S3 y lo migra a la base de datos de destino.

### Note

Le recomendamos que pruebe su migración antes de utilizar el dispositivo Snowball Edge. Puede hacerlo configurando una tarea para enviar datos, como por ejemplo una sola tabla, a un bucket de Amazon S3 en lugar de al dispositivo Snowball Edge.

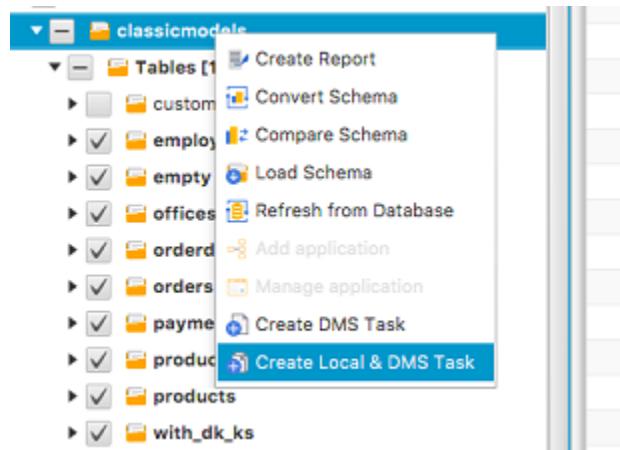
Para crear la tarea de migración integral

1. Inicie AWS SCT, elija View (Ver) y, a continuación, elija Database Migration View (Local & DMS) [Vista de migración de base de datos (local y DMS)].

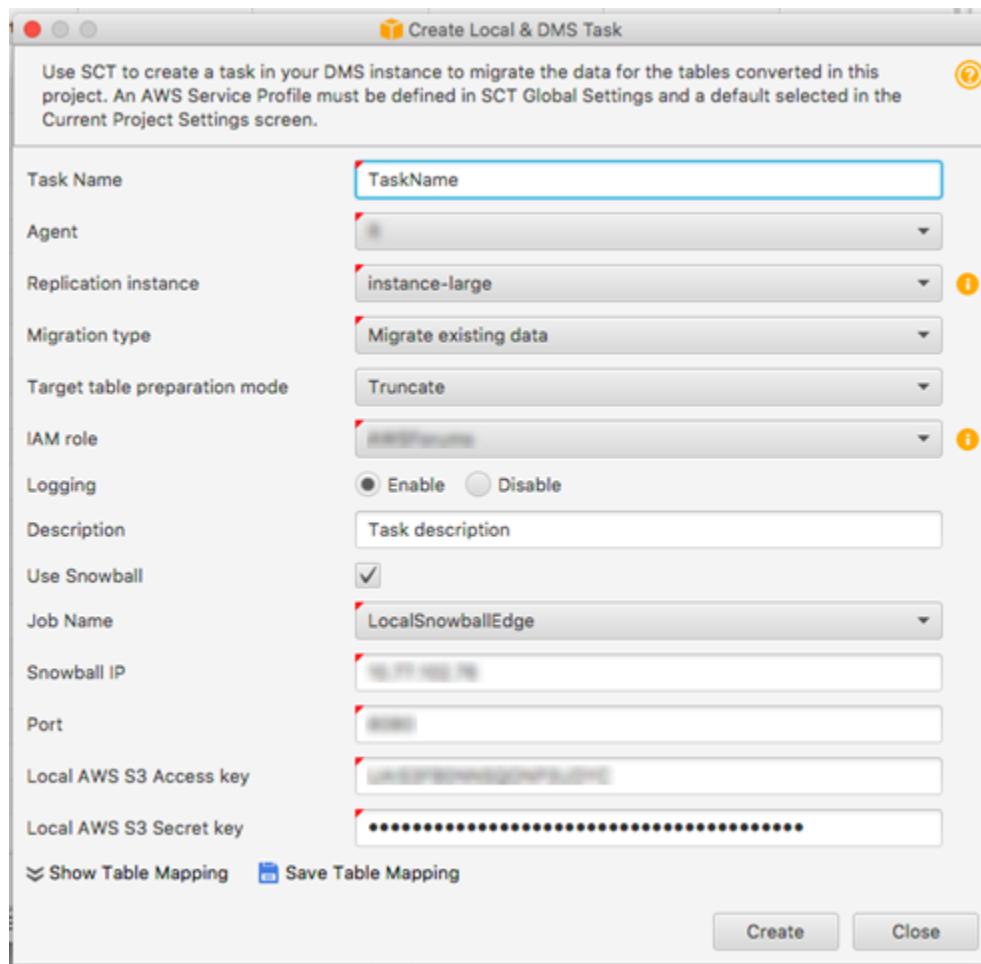


2. En el panel de la izquierda que muestra el esquema de la base de datos de origen, seleccione un esquema para su migración. Abra el menú contextual (haga clic con el botón derecho) para el esquema y seleccione Create Local & DMS Task (Crear tarea local y DMS).

No puede migrar tablas individuales mediante AWS DMS y Snowball Edge.



Aparece la siguiente pantalla.



3. Añada la información de su tarea.

Para este parámetro	Haga lo siguiente
Task Name	Introduzca un nombre para la tarea.
Agente	Elija DMS Agent (Agente de DMS).
Replication Instance (Instancia de replicación)	Elija la instancia de replicación de AWS DMS que desea usar.
Tipo de migración	<p>Elija el tipo de migración que desea:</p> <p>Elija Migrate existing data (Migrar datos existente) para migrar el contenido del esquema elegido. Este proceso se denomina carga completa en AWS DMS.</p> <p>Elija Migrate existing data and replicate ongoing changes (Migrar los datos existentes y replicar los cambios continuos) para migrar el contenido del esquema elegido y capturar todos los cambios continuos a la base de datos. Este proceso se denomina carga completa y CDC en AWS DMS.</p>

Para este parámetro	Haga lo siguiente
Target table preparation mode	Elija el modo de preparación que desea utilizar:  Truncate (Truncar): las tablas se truncan sin que los metadatos de la tabla se vean afectados.  Drop tables on target (Eliminar tablas en destino): las tablas existentes se eliminan y se crean otras para sustituirlas.  Do nothing (No hacer nada): los datos y los metadatos de las tablas de destino no cambiarán.
Rol de IAM	Elija el rol de IAM predefinido que tiene permisos para acceder al bucket de Amazon S3 y a la base de datos de destino. Para obtener más información acerca de los permisos necesarios para obtener acceso a un bucket de Amazon S3, consulte <a href="#">Requisitos previos para utilizar S3 como origen para AWS DMS</a> .
Compression format (Formato de compresión)	Elija si desea cargar los archivos comprimidos o no:  GZIP: los archivos se comprimen antes de la carga. Esta es la opción predeterminada.  No Compression (Sin compresión): las extracciones son más rápidas, pero requiere más espacio.
Logging (Registro)	Elija Enable (Habilitar) para que Amazon CloudWatch cree registros para la migración. Se le cobrarán cargos por este servicio. Para obtener más información acerca de CloudWatch, consulte <a href="#">Cómo funciona Amazon CloudWatch</a> .
Description (Descripción)	Introduzca una breve descripción de la tarea.
S3 Bucket (Bucket de S3)	Introduzca el nombre de un bucket de S3 configurado para este trabajo de Snowball Edge en la consola de Snowball Edge
Utilice Snowball Edge	Elija esta casilla de verificación para utilizar Snowball Edge. Si esta casilla no está marcada, los datos se cargan directamente en el bucket de S3.
Job Name	Elija el nombre de trabajo de Snowball Edge que ha creado.
Snowball Edge IP	Introduzca la dirección IP del dispositivo Snowball Edge.
Puerto	Introduzca el valor de puerto para el dispositivo Snowball Edge.
Local Amazon S3 Access key (Clave de acceso local de Amazon S3)	Introduzca la clave de acceso local de Snowball Edge que recuperó del dispositivo.
Local Amazon S3 Secret key (Clave secreta local de Amazon S3)	Introduzca la clave secreta local de Snowball Edge que recuperó del dispositivo.

4. Seleccione Create (Crear) para crear la tarea.

## Paso 10: Ejecutar y monitorizar la tarea en SCT

Puede iniciar la tarea de migración cuando las conexiones a todos los puntos de enlace se hayan realizado con éxito, incluidas las siguientes:

- Conexiones del agente de DMS a:
  - La base de datos de origen
  - El bucket de Amazon S3 transitorio
  - El dispositivo Edge
- Conexiones de tarea de DMS a:
  - El bucket de Amazon S3 transitorio
  - La base de datos de destino en AWS

Si todas las conexiones están funcionando correctamente, la consola de SCT tiene un aspecto similar a la siguiente captura de pantalla y está listo para comenzar.

The screenshot shows the AWS Database Migration Service (DMS) SCT console. In the top navigation bar, the 'Tasks' tab is selected. Below the navigation bar, a table displays a single task named 'abhinavmigration'. The task status is '0%' complete, with 0 tables loaded and 0 tables loading. Below the table, there are several buttons: Start, Stop, Delete, Test, Refresh, and Show log. The 'Task details' section provides more information about the task, including its version (0), round trip latency (0 ms), source endpoint (name: sqlserver, status: Successful, connection message: 'Connection successful'), and target endpoint (name: aurora-postgresql, status: Successful, connection message: 'Connection successful').

Utilice el procedimiento siguiente para iniciar la migración.

Para iniciar la tarea de migración

1. Elija la tarea de migración, a continuación elija Start (Comenzar).

Task Name	Status	Complete %	Elapsed Time	Tables Loaded	Tables Loading	Tables...
abhinavmigration		0%		0	0	0
abhinavmigration-Local-generated		0%		0	0	0
abhinavmigration-DMS-generated		0%		0	0	0
snowball-migration		0%		0	0	0
snowball-migration-Local-generated		0%		0	0	0
snowball-migration-DMS-generated		0%		0	0	0

Buttons at the bottom: Start, Stop, Delete, Test, Refresh, Show log.

Task details panel:

- ID: f26b735c2dbc462283d5a89f2d5a7961
- Task name: abhinavmigration

- Para monitorizar el agente de DMS, seleccione Show Log (Mostrar registro). Los detalles del registro incluyen los registros del servidor del agente (Agent Log [Registro del agente]) y de la tarea de ejecución local (Task Log [Registro de tareas]). La conectividad del punto de enlace se realiza por el servidor. Dado que la tarea local no se está ejecutando, no tiene registros de tareas. Los problemas de conexión se muestran en la pestaña Agent Log (Registro del agente).

Date	Component	Verbosity	Text
2017-12-01 09:52:00	INFRASTRUCTURE	Info	Allocated 20 ODBC connect...
2017-12-01 09:52:01	SOURCE_CAPTURE	Info	Source endpoint 'Mysql' is u...
2017-12-01 09:52:01	TARGET_LOAD	Info	Start loading table 'classicm...
2017-12-01 09:52:01	TARGET_LOAD	Info	Start loading table 'classicm...

- Verifique que el estado de la tarea de migración es del 50 por ciento. Puede utilizar la consola de Snowball Edge o AWS SCT para comprobar el estado del dispositivo.

Task Name	Status	Complete %	Elapsed Time	Tables
DemoTask		50%		0
DemoTask-Local		100%	0h:00m:11s	10
DemoTask-DMS		0%	0h:01m:00s	0

Una vez que las tablas de origen se han cargado en el dispositivo Snowball Edge, AWS SCT actualiza el estado de la tarea para mostrar que se ha completado al 50 por ciento. Esto se debe a que la otra mitad de la tarea implica que AWS DMS tome los datos de Amazon S3 al almacén de datos de destino.

4. Siga los pasos que se indican en la documentación de Snowball Edge, empezando por la sección denominada [Detención del cliente de Snowball y apagado del dispositivo Snowball Edge](#). Estos pasos incluyen lo siguiente:
  - Detener el cliente de Snowball Edge
  - Apagar el dispositivo Edge
  - Devolución del dispositivo Edge a AWS
5. Finalizar la migración después de que el dispositivo se devuelve a AWS implica esperar a que se complete la tarea remota.

Cuando el dispositivo Snowball Edge se recibe en AWS, se inicia la ejecución de la tarea (DMS) remota. Si el tipo de migración que ha elegido fue Migrate existing data (Migrar datos existentes), el estado de la tarea de DMS mostrará 100 % completado cuando se hayan transferido los datos desde Amazon S3 al almacen de datos de destino.

Si se establece el modo de una tarea para incluir la replicación continua, entonces, una vez que la carga completa se haya completado, la tarea continúa ejecutándose, mientras AWS DMS aplica cambios continuos.

## Limitaciones cuando se trabaja con Snowball Edge y AWS DMS

Existen algunas limitaciones que deben tenerse en cuenta a la hora de trabajar con AWS Snowball Edge:

- Cada tarea de AWS SCT crea dos conexiones de punto de enlace en AWS DMS. Si crea varias tareas, puede alcanzar el límite de un recurso para el número de puntos de enlace que puede crear.
- Un esquema es el ámbito de tarea mínimo cuando se utiliza Snowball Edge. No puede migrar tablas individuales o subconjuntos de tablas mediante Snowball Edge.
- El agente de DMS no admite HTTP/HTTPS o configuraciones de proxy SOCKS. Las conexiones al origen y al destino pueden generar un error si el host del agente de DMS utiliza proxies.
- El modo de LOB limita el tamaño de archivo de LOB a 32 K. Los LOB mayores de 32 K no se migran.
- En algunos casos, se puede producir un error al cargar desde la base de datos local al dispositivo Edge o al cargar datos desde Amazon S3 a la base de datos de destino. En algunos de estos casos, el error se puede recuperar y la tarea se puede reiniciar. Si AWS DMS no puede recuperarse del error, la migración se detiene. Si esto ocurre, póngase en contacto con AWS Support.

# Solución de problemas de tareas de migración en AWS Database Migration Service

Las secciones siguientes facilitan información sobre cómo solucionar problemas con AWS Database Migration Service (AWS DMS).

## Temas

- [Tareas de migración lentas \(p. 369\)](#)
- [La barra de estado de la tarea no se mueve \(p. 370\)](#)
- [Faltan claves externas e índices secundarios \(p. 370\)](#)
- [Problemas de conexión de Amazon RDS \(p. 370\)](#)
- [Problemas de red \(p. 371\)](#)
- [Atasco de CDC después de migrar una carga total \(p. 371\)](#)
- [Errores de violación de clave principal al reiniciar una tarea \(p. 371\)](#)
- [Error al cargar por primera vez los esquemas \(p. 372\)](#)
- [Fallan las tareas con un error desconocido \(p. 372\)](#)
- [Al reiniciar la tarea las tablas se cargan desde el principio \(p. 372\)](#)
- [Número de tablas por tarea \(p. 372\)](#)
- [Solución de problemas específicos de Oracle \(p. 372\)](#)
- [Solución de problemas específicos de MySQL \(p. 375\)](#)
- [Solución de problemas específicos de PostgreSQL \(p. 379\)](#)
- [Solución de problemas específicos de Microsoft SQL Server \(p. 381\)](#)
- [Solución de problemas específicos de Amazon Redshift \(p. 382\)](#)
- [Solución de problemas específicos de Amazon Aurora MySQL \(p. 384\)](#)

## Tareas de migración lentas

Existen varios problemas que pueden provocar lentitud en una tarea de migración o hacer que las tareas posteriores se ejecuten a menor velocidad que la tarea inicial. La razón más común para que una tarea de migración se ejecute con lentitud es que se hayan asignado los recursos inadecuados a la instancia de replicación de AWS DMS. Compruebe el uso que su instancia de replicación hace de la CPU, de la memoria, de los archivos de intercambio y de las IOPS para asegurarse de que tiene suficientes recursos para las tareas que se ejecutan en ella. Por ejemplo, si hay varias tareas con Amazon Redshift como punto de enlace, se generan muchas operaciones de E/S. Puede ampliar el número de IOPS para su instancia de replicación o dividir las tareas entre varias instancias de replicación para lograr una migración más eficaz.

Para obtener más información sobre cómo determinar el tamaño de su instancia de replicación, consulte [Selección del tamaño óptimo de una instancia de replicación \(p. 387\)](#)

Para aumentar la velocidad de una carga de migración inicial, haga lo siguiente:

- Si su objetivo es una instancia de base de datos de Amazon RDS, asegúrese de que Multi-AZ no esté habilitada para la instancia de la base de datos de destino.

- Durante la carga, desactive cualquier copia de seguridad automática o el inicio de sesión en la base de datos de destino y vuelva a activar estas funciones en cuanto finalice la migración.
- Si la función está disponible en el destino, utilice IOPS provisionadas.
- Si sus datos de migración contienen LOB, compruebe que la tarea esté optimizada para migrar listas LOB. Consulte [Configuración de las tareas de los metadatos de destino \(p. 263\)](#) para obtener más información sobre cómo optimizar la migración de listas LOB.

## La barra de estado de la tarea no se mueve

La barra de estado de la tarea proporciona una estimación del avance de la tarea. La calidad de esta estimación depende de la calidad de las estadísticas de la tabla de la base de datos de origen; cuanto mejores sean las estadísticas de la tabla, más precisa será la estimación. Si una tarea solo tiene una tabla sin estimación de estadísticas de fila, no podemos proporcionar ningún tipo de estimación sobre el porcentaje completado. En este caso, puede utilizar el estado de la tarea y la indicación de las filas cargadas para confirmar que la tarea está en ejecución y avanzando.

## Faltan claves externas e índices secundarios

AWS DMS crea tablas, claves primarias y, en algunos casos, índices únicos, pero no crea ningún otro objeto que no se necesite para migrar eficientemente los datos desde el origen. Por ejemplo, no crea índices secundarios, limitaciones de claves no primarias ni valores predeterminados de datos.

Para migrar objetos secundarios desde la base de datos, utilice las herramientas nativas de la base de datos si está migrando al mismo motor de base de datos que su base de datos de origen. Utilice Schema Conversion Tool si está migrando a otro motor de base de datos distinto del utilizado por su base de datos de origen para migrar objetos secundarios.

## Problemas de conexión de Amazon RDS

Puede haber varias razones por las que no pueda conectar con una instancia de base de datos Amazon RDS establecida como un punto de enlace. Entre ellas se incluyen:

- La combinación de nombre de usuario y contraseña es incorrecta.
- Compruebe que el valor del punto de enlace que aparece en la consola de Amazon RDS para la instancia es el mismo que el identificador del punto de enlace utilizado para crear el punto de enlace DMS de AWS.
- Compruebe que el valor del puerto mostrado en la consola de Amazon RDS para la instancia es el mismo que el puerto asignado al punto de enlace de DMS de AWS.
- Compruebe que el grupo de seguridad asignado a la instancia de base de datos de Amazon RDS permite conexiones desde la instancia de replicación de AWS DMS.
- Si la instancia de replicación de AWS DMS y la instancia de base de datos de Amazon RDS no están en la misma VPC, compruebe que se puede acceder públicamente a la instancia de la base de datos.

## Mensaje de error de cadena de conexión de hilo incorrecta y valor de hilo incorrecto 0

Este error puede producirse a menudo al probar el enlace a un punto de enlace. El error indica que hay un problema en la cadena de conexión, como un espacio después de la dirección IP de host o que se ha copiado un carácter incorrecto en la cadena de conexión.

## Problemas de red

El problema de red más habitual tiene que ver con el grupo de seguridad de VPC que utiliza la instancia de replicación de AWS DMS. De forma predeterminada, este grupo de seguridad tiene normas que permiten salidas a 0.0.0.0/0 en todos los puertos. Si modifica este grupo de seguridad o utiliza su propio grupo de seguridad, la salida debe estar permitida al menos a los puntos de enlace de origen y de destino en los puertos de la base de datos respectivos.

Otros problemas relacionados con la configuración son:

- La instancia de replicación y los puntos de enlace de origen y de destino están en la misma VPC: el grupo de seguridad que utilizan los puntos de enlace debe permitir recibir datos en el puerto desde la instancia de replicación. Asegúrese de que el grupo de seguridad utilizado por la instancia de replicación entra a los puntos de enlace. Otra opción es crear una regla en el grupo de seguridad que utilizan los puntos de enlace que otorgue acceso a la dirección IP privada de la instancia de replicación.
- El punto de enlace de origen está fuera de la VPC que utiliza la instancia de replicación (a través de la gateway a Internet): el grupo de seguridad de la VPC debe incluir normas de direccionamiento que envíen el tráfico no destinado a la VPC a la gateway a Internet. En esta configuración, la conexión con el punto de enlace parece provenir de la dirección IP pública de la instancia de replicación.
- El punto de enlace de origen está fuera de la VPC que utiliza la instancia de replicación (mediante gateway a NAT): puede configurar una gateway de traducción de direcciones de red (NAT) mediante una única dirección IP elástica asociada a una única interfaz de red elástica que reciba después un identificador NAT (nat- #####). Si la VPC incluye una ruta predeterminada a dicho NAT en lugar de la gateway de Internet, la instancia de replicación aparecerá para ponerse en contacto con el punto de enlace de la base de datos mediante la dirección IP pública de la gateway de Internet. En este caso, la entrada al punto de enlace de la base de datos fuera de la VPC debe permitir la entrada de la dirección NAT en lugar de la dirección IP pública de la instancia de replicación.

## Atasco de CDC después de migrar una carga total

Los cambios de la replicación pueden ser lentos o se pueden producir atascos después de haber realizado una migración de carga completa y si hay varios ajustes de AWS DMS en conflicto unos con otros. Por ejemplo, si el parámetro Target table preparation mode (Modo de preparación de tabla de destino) está definido como Do nothing (No hacer nada) o Truncate (Truncar), entonces ha indicado a AWS DMS que no configure las tablas de destino, incluido crear índices principales y únicos. Si no ha creado claves primarias ni únicas en las tablas de destino, AWS DMS deberá analizar por completo la tabla en cada actualización, lo cual puede afectar al rendimiento.

## Errores de violación de clave principal al reiniciar una tarea

Este error se puede producir cuando permanecen en la base de datos de destino datos procedentes de una tarea de migración anterior. Si el parámetro Target table preparation mode (Modo de preparación de tabla de destino) está definido en Do nothing (No hacer nada), AWS DMS no hace ninguna actividad de preparación en la tabla de destino, ni tampoco limpia datos insertados en una tarea anterior. A fin de reiniciar su tarea y evitar que se produzcan estos errores, debe eliminar las filas insertadas en las tablas de destino en la ejecución anterior de la tarea.

## Error al cargar por primer vez los esquemas

Si la carga inicial de los esquemas falla con un error `Operation:getSchemaListDetails:errType=, status=0, errMessage=, errDetails=`, la cuenta de usuario que utiliza AWS DMS para conectar con el punto de enlace de origen no tiene los permisos necesarios.

## Fallan las tareas con un error desconocido

La causa de estos tipos de error pueden ser varias, pero a menudo observamos que el problema se debe a que los recursos asignados a la instancia de replicación de AWS DMS no son suficientes. Compruebe el uso que la instancia de replicación hace de la CPU, de la memoria, de los archivos de intercambio y de las IOPS para confirmar que su instancia tiene suficientes recursos para realizar la migración. Para obtener más información acerca de la monitorización, consulte [Métricas del servicio de migración de datos \(p. 325\)](#).

## Al reiniciar la tarea las tablas se cargan desde el principio

AWS DMS reinicia la carga de tablas desde el principio cuando no ha terminado la carga inicial de una tabla. Cuando se reinicia una tarea, AWS DMS no vuelve a cargar las tablas que completaban la carga inicial, sino que vuelve a cargar las tablas desde el principio antes de finalizar la carga inicial.

## Número de tablas por tarea

Aunque no hay un límite fijo en el número de tablas por cada tarea de replicación, hemos visto que por lo general suele resultar acertado limitar el número de tablas de una tarea a menos de 60 000. El uso de recursos puede provocar atascos si una única tarea utiliza más de 60 000 tablas.

## Solución de problemas específicos de Oracle

Los siguientes problemas surgen específicamente al usar AWS DMS con bases de datos de Oracle.

### Temas

- [Obtención de datos de vistas \(p. 372\)](#)
- [Migración de LOB desde Oracle 12c \(p. 373\)](#)
- [Alternar entre Oracle LogMiner y Binary Reader \(p. 373\)](#)
- [Error de CDC de Oracle detenido 122301 y de tope de reintentos de CDC de Oracle superado. \(p. 373\)](#)
- [Adición automática de un registro complementario a un punto de enlace de origen de Oracle \(p. 374\)](#)
- [No se están capturando los cambios de LOB \(p. 374\)](#)
- [Error ORA-12899 de valor demasiado grande para <nombre-columna> \(p. 374\)](#)
- [Malinterpretación del tipo de datos NUMBER \(p. 374\)](#)

## Obtención de datos de vistas

Puede extraer los datos una vez desde una vista; no puede utilizarlos para la replicación continua. Para poder extraer los datos de las vistas, debe añadir el código siguiente a Extra connection attributes en la

sección Advanced del punto de enlace de origen de Oracle. Tenga en cuenta que al extraer los datos de una vista, la vista se muestra como una tabla en el esquema de destino.

```
exposeViews=true
```

## Migración de LOB desde Oracle 12c

AWS DMS puede utilizar dos métodos para capturar cambios en una base de datos de Oracle: Binary Reader y Oracle LogMiner. De forma predeterminada, AWS DMS utiliza Oracle LogMiner para capturar los cambios. Sin embargo, en Oracle 12c, Oracle LogMiner no admite columnas de LOB. Para capturar cambios en columnas LOB en Oracle 12c, utilice Binary Reader.

## Alternar entre Oracle LogMiner y Binary Reader

AWS DMS puede utilizar dos métodos para capturar cambios en una base de datos origen de Oracle: Binary Reader y Oracle LogMiner. Oracle LogMiner es la opción predeterminada. Si desea cambiar y usar Binary Reader para capturar cambios, haga lo siguiente:

Para utilizar Binary Reader para capturar cambios

1. Inicie sesión en la consola de administración de AWS y seleccione DMS.
2. Seleccione Endpoints.
3. Seleccione el punto de enlace de origen de Oracle donde utilizar Binary Reader.
4. Seleccione Modify.
5. Seleccione Advanced y agregue después el código siguiente en el cuadro de texto Extra connection attributes:

```
useLogminerReader=N
```

6. Utilice una herramienta de desarrollador de Oracle como SQL-Plus para conceder el privilegio adicional siguiente a la cuenta de usuario de AWS DMS empleada para conectar con el punto de enlace de origen de Oracle:

```
SELECT ON V_$TRANSPORTABLE_PLATFORM
```

## Error de CDC de Oracle detenido 122301 y de tope de reintentos de CDC de Oracle superado.

Este error se produce cuando los registros de archivos de Oracle necesarios se han eliminado de su servidor antes de que AWS DMS pudiera utilizarlos para capturar los cambios. Amplíe sus políticas de retención de registros en el servidor de base de datos. Para una base de datos de Amazon RDS, ejecute el procedimiento siguiente para ampliar la retención de registros. El código del ejemplo siguiente amplía la retención de registros en una instancia de base de datos de Amazon RDS a 24 horas.

```
exec rdsadmin.rdsadmin_util.set_configuration('archivelog retention hours',24);
```

## Adición automática de un registro complementario a un punto de enlace de origen de Oracle

De forma predeterminada, el registro complementario de AWS DMS está desactivado. Para activar automáticamente el registro complementario para un punto de enlace de origen de Oracle, haga lo siguiente:

Para agregar un registro complementario a un punto de enlace de origen de Oracle

1. Inicie sesión en la consola de administración de AWS y seleccione DMS.
2. Seleccione Endpoints.
3. Seleccione el punto de enlace de origen de Oracle al que desee agregar el registro complementario.
4. Seleccione Modify.
5. Seleccione Advanced y agregue después el código siguiente en el cuadro de texto Extra connection attributes:

```
addSupplementalLogging=Y
```

6. Elija Modify.

## No se están capturando los cambios de LOB

En la actualidad, una tabla debe tener una clave principal para que AWS DMS pueda capturar los cambios de LOB. Si una tabla que contiene LOB no tiene una clave principal, hay varias medidas que puede aplicar para capturar los cambios de los LOB:

- Añadir una clave principal a la tabla. Esto puede ser tan sencillo como añadir una columna de ID y rellenarla con una secuencia utilizando un activador.
- Crear una vista materializada de la tabla que incluya un ID generado por el sistema como clave principal y migrar la vista materializada en lugar de la tabla.
- Crear una espera lógica, agregar una clave principal a la tabla y migrar desde la espera lógica.

## Error ORA-12899 de valor demasiado grande para <nombre-columna>

El error ORA-12899 que indica que el valor es demasiado grande para <nombre-columna> suele ser debido a una discrepancia en el conjunto de caracteres que utilizan las bases de datos de origen y de destino o porque los ajustes de NLS difieren entre las dos bases de datos. Una causa habitual de este error es que el parámetro NLS\_LENGTH\_SEMANTICS de la base de datos de origen esté definido en CHAR y el parámetro NLS\_LENGTH\_SEMANTICS en BYTE.

## Malinterpretación del tipo de datos NUMBER

El tipo de datos NUMBER de Oracle se convierte a varios tipos de datos de AWS DMS dependiendo de la precisión y la escala de NUMBER. Estas conversiones pueden consultarse aquí [Uso de una base de datos de Oracle como origen para AWS DMS \(p. 91\)](#). El modo de conversión del tipo NUMBER también puede verse afectado por el uso de atributos de conexión adicionales para el punto de enlace de origen de

Oracle. Estos atributos de conexión adicionales se pueden consultar en [Atributos de conexión adicionales al usar Oracle como origen para AWS DMS \(p. 100\)](#).

## Solución de problemas específicos de MySQL

Los siguientes problemas surgen específicamente al usar AWS DMS con bases de datos de MySQL.

### Temas

- [Error de la tarea CDC para el punto de enlace de la instancia de base de datos de Amazon RDS al haberse deshabilitado el registro binario \(p. 375\)](#)
- [Las conexiones a una instancia de MySQL de destino se desconectan durante una tarea \(p. 375\)](#)
- [Adición de Autocommit a un punto de enlace compatible con MySQL \(p. 376\)](#)
- [Desactivación de claves externas en un punto de enlace de destino compatible con MySQL \(p. 376\)](#)
- [Caracteres sustituidos por signos de interrogación \(p. 377\)](#)
- [Entradas de registro "Bad event" \(evento incorrecto\) \(p. 377\)](#)
- [Captura de datos de cambios con MySQL 5.5 \(p. 377\)](#)
- [Aumento de la retención de registros binarios para instancias de base de datos de Amazon RDS \(p. 377\)](#)
- [Mensaje de registro que indica que algunos cambios desde la base de datos de origen no han surtido efecto al aplicarlos a la base de datos de destino. \(p. 377\)](#)
- [Error de identificador demasiado largo \(p. 377\)](#)
- [Error por el que un juego de caracteres incompatible generar error en la conversión de datos del campo \(p. 378\)](#)
- [Error: página de códigos 1252 a UTF8 \[120112\] Se ha producido un error en la conversión de datos del campo \(p. 378\)](#)

## Error de la tarea CDC para el punto de enlace de la instancia de base de datos de Amazon RDS al haberse deshabilitado el registro binario

Este problema se produce con las instancias de base de datos de Amazon RDS, porque las copias de seguridad automatizadas están deshabilitadas. Habilite las copias de seguridad automáticas fijando el período de retención de copia de seguridad en un valor diferente de cero.

## Las conexiones a una instancia de MySQL de destino se desconectan durante una tarea

Si una tarea con LOB se está desconectando de un destino MySQL con el siguiente tipo de errores en el registro de tareas, es posible que tenga que modificar algunos de los ajustes de esa tarea.

```
[TARGET_LOAD ]E: RetCode: SQL_ERROR SqlState: 08S01 NativeError:  
2013 Message: [MySQL][ODBC 5.3(w) Driver][mysqld-5.7.16-log]Lost connection  
to MySQL server during query [122502] ODBC general error.
```

```
[TARGET_LOAD ]E: RetCode: SQL_ERROR SqlState: HY000 NativeError:  
2006 Message: [MySQL][ODBC 5.3(w) Driver]MySQL server has gone away
```

[122502] ODBC general error.

Para resolver el problema de una tarea que se esté desconectado de un destino MySQL, haga lo siguiente:

- Compruebe que ha definido la variable `max_allowed_packet` de su base de datos en un valor suficientemente alto como para almacenar sus LOB más grandes.
- Compruebe que ha configurado las variables siguientes para disponer de un valor de tiempo de espera amplio. Le sugerimos que utilice un valor mínimo de 5 minutos para cada una de estas variables.
  - `net_read_timeout`
  - `net_write_timeout`
  - `wait_timeout`
  - `interactive_timeout`

## Adición de Autocommit a un punto de enlace compatible con MySQL

Para añadir autocommit a un punto de enlace de destino compatible con MySQL

1. Inicie sesión en la consola de administración de AWS y seleccione DMS.
2. Seleccione Endpoints.
3. Seleccione el punto de enlace de destino compatible con MySQL al que desee añadir autocommit.
4. Seleccione Modify.
5. Seleccione Advanced y agregue después el código siguiente en el cuadro de texto Extra connection attributes:

```
Initstmt= SET AUTOCOMMIT=1
```

6. Elija Modify.

## Desactivación de claves externas en un punto de enlace de destino compatible con MySQL

Puede desactivar las comprobaciones de claves externas en MySQL añadiendo lo siguiente a Extra Connection Attributes (Atributos de conexión adicionales), en la sección Advanced (Avanzado) del punto de enlace de destino MySQL, Compatibilidad de Amazon Aurora con MySQL o MariaDB.

Para desactivar claves externas en un punto de enlace de destino compatible con MySQL

1. Inicie sesión en la consola de administración de AWS y seleccione DMS.
2. Seleccione Endpoints.
3. Seleccione el punto de enlace de destino de MySQL, Aurora MySQL o MariaDB cuyas claves externas desea deshabilitar.
4. Seleccione Modify.
5. Seleccione Advanced y agregue después el código siguiente en el cuadro de texto Extra connection attributes:

```
Initstmt=SET FOREIGN_KEY_CHECKS=0
```

6. Elija Modify.

## Caracteres sustituidos por signos de interrogación

La situación que más habitualmente origina este problema es que los caracteres del punto de enlace de origen se hayan codificado mediante un juego de caracteres incompatibles con AWS DMS. Por ejemplo, las versiones de motor de AWS DMS anteriores a la versión 3.1.1 no admiten el conjunto de caracteres UTF8MB4.

## Entradas de registro "Bad event" (evento incorrecto)

Las entradas de evento incorrecto en los registros de migración suelen indicar que se ha intentado realizar una operación DDL no admitida en el punto de enlace de la base de datos de origen. Las operaciones DDL incompatibles generan un evento que la instancia de replicación no puede omitir, por lo que se registra un evento incorrecto. Para solucionar este problema, reinicie la tarea desde el principio, para que vuelvan a cargarse las tablas y se empiecen a capturar cambios en un punto posterior a la operación DDL incompatible.

## Captura de datos de cambios con MySQL 5.5

La captura de datos de cambios (CDC) de AWS DMS para bases de datos compatibles con MySQL de Amazon RDS requiere un registro binario de imagen completa basado en filas, incompatible con la versión de MySQL 5.5 o anteriores. Para utilizar la función CDC de AWS DMS debe actualizar su instancia de base de datos de Amazon RDS a MySQL versión 5.6.

## Aumento de la retención de registros binarios para instancias de base de datos de Amazon RDS

AWS DMS precisa que se retengan archivos de registros binarios para capturar datos de cambios. Para retener registros durante más tiempo en una instancia de base de datos de Amazon RDS, siga este procedimiento. El ejemplo que sigue amplía el tiempo de retención de registros binarios hasta 24 horas.

```
call mysql.rds_set_configuration('binlog retention hours', 24);
```

## Mensaje de registro que indica que algunos cambios desde la base de datos de origen no han surtido efecto al aplicarlos a la base de datos de destino.

Cuando AWS DMS actualiza el valor de una columna de la base de datos MySQL a su valor existente, MySQL devuelve un mensaje `zero rows affected`. Este comportamiento es distinto de otros motores de base de datos, como Oracle y SQL Server, los cuales actualizan las filas aunque el valor de sustitución sea el mismo que el existente.

## Error de identificador demasiado largo

El siguiente error se produce cuando un identificador es demasiado largo:

```
TARGET_LOAD E: RetCode: SQL_ERROR SqlState: HY000 NativeError:  
1059 Message: MySQLhttp://ODBC 5.3(w) Driverhttp://mysqld-5.6.10Identifier  
name '<name>' is too long 122502 ODBC general error. (ar_odbc_stmt.c:4054)
```

Cuando se define para crear tablas y claves primarias en la base de datos de destino, AWS DMS no utiliza los mismos nombres para las claves primarias que los usados en la base de datos de origen. En vez de eso, AWS DMS crea el nombre de la clave principal basándose en el nombre de las tablas. Cuando el nombre de la tabla es largo, el identificador generado automáticamente puede superar el límite permitido para MySQL. Para resolver este problema, cree antes las tablas y claves primarias en la base de datos de destino y utilice una tarea con el ajuste Target table preparation mode definido en Do nothing o Truncate para llenar las tablas de destino.

## Error por el que un juego de caracteres incompatible generar error en la conversión de datos del campo

El siguiente error se produce cuando un juego de caracteres no compatible genera error en la conversión de datos del campo:

```
"[SOURCE_CAPTURE ]E: Column '<column name>' uses an unsupported character set [120112]  
A field data conversion failed. (mysql_endpoint_capture.c:2154)
```

Este error a menudo se produce por tablas o bases de datos que utilizan codificación UTF8MB4. Las versiones de motor de AWS DMS anteriores a la 3.1.1. no admiten el conjunto de caracteres UTF8MB4. Compruebe también los parámetros de su base de datos relacionados con las conexiones. Para ver estos parámetros puede utilizarse el siguiente comando:

```
SHOW VARIABLES LIKE '%char%';
```

## Error: página de códigos 1252 a UTF8 [120112] Se ha producido un error en la conversión de datos del campo

El siguiente error puede producirse durante una migración si existen caracteres que no pertenecen a la página de códigos 1252 en la base de datos MySQL de origen.

```
[SOURCE_CAPTURE ]E: Error converting column 'column_xyz' in table  
'table_xyz' with codepage 1252 to UTF8 [120112] A field data conversion failed.  
(mysql_endpoint_capture.c:2248)
```

Como solución provisional, puede utilizar el atributo de conexión adicional CharsetMapping con el punto de enlace de MySQL de origen para especificar el mapeo del conjunto de caracteres. Es posible que tenga que reiniciar la tarea de migración de AWS DMS desde el principio si añade este atributo de conexión adicional.

Por ejemplo, el siguiente atributo de conexión adicional puede utilizarse para un punto de enlace de origen de MySQL donde el conjunto de caracteres de origen es `utf8` o `latin1`. `65001` es el identificador de la página de códigos UTF8.

```
CharsetMapping=utf8,65001
CharsetMapping=latin1,65001
```

## Solución de problemas específicos de PostgreSQL

Los siguientes problemas surgen específicamente al usar AWS DMS con bases de datos de PostgreSQL.

### Temas

- [Tipos de datos JSON truncados \(p. 379\)](#)
- [Las columnas de un tipo de datos definido por el usuario no se migran correctamente \(p. 380\)](#)
- [Error que indica que no se ha seleccionado ningún esquema de creación \(p. 380\)](#)
- [No se están replicando las eliminaciones y las actualizaciones en una tabla mediante CDC \(p. 380\)](#)
- [Las instrucciones TRUNCATE no se están propagando \(p. 380\)](#)
- [Impedir que PostgreSQL capture instrucciones DDL \(p. 380\)](#)
- [Selección del esquema donde crear los objetos de base de datos para capturar instrucciones DDL \(p. 380\)](#)
- [Ausencia de tablas de Oracle después de migrar a PostgreSQL \(p. 381\)](#)
- [La tarea de uso de vista como origen no tiene filas copiadas \(p. 381\)](#)

## Tipos de datos JSON truncados

AWS DMS trata el tipo de datos JSON en PostgreSQL como columna del tipo de datos LOB. Esto significa que el límite de tamaño de LOB cuando utilice el modo Limited LOB se aplica a datos JSON. Por ejemplo, si se establece el modo Limited LOB en 4096 KB, cualquier dato JSON de más de 4096 KB se trunca en el límite de 4096 KB y fallará la prueba de validación en PostgreSQL.

Por ejemplo, la siguiente información de registro muestra JSON truncado debido a la configuración de modo Limited LOB y error de validación.

```
03:00:49
2017-09-19T03:00:49 [TARGET_APPLY ]E: Failed to execute statement:
'UPDATE "public"."delivery_options_quotes" SET "id"=? , "enabled"=? ,
"new_cart_id"=? , "order_id"=? , "user_id"=? , "zone_id"=? , "quotes"=? ,
"start_at"=? , "end_at"=? , "last_quoted_at"=? , "created_at"=? ,
"updated_at"=? WHERE "id"=? ' [1022502] (ar_odbc_stmt
2017-09-19T03:00:49 [TARGET_APPLY ]E: Failed to execute statement:
'UPDATE "public"."delivery_options_quotes" SET "id"=? , "enabled"=? ,
"new_cart_id"=? , "order_id"=? , "user_id"=? , "zone_id"=? , "quotes"=? ,
"start_at"=? , "end_at"=? , "last_quoted_at"=? , "created_at"=? ,
"updated_at"=? WHERE "id"=? ' [1022502] (ar_odbc_stmt.c:2415)
#
03:00:49
2017-09-19T03:00:49 [TARGET_APPLY ]E: RetCode: SQL_ERROR SqlState:
22P02 NativeError: 1 Message: ERROR: invalid input syntax for type json;;
Error while executing the query [1022502] (ar_odbc_stmt.c:2421)
2017-09-19T03:00:49 [TARGET_APPLY ]E: RetCode: SQL_ERROR SqlState:
22P02 NativeError: 1 Message: ERROR: invalid input syntax for type json;;
```

Error while executing the query [1022502] (ar\_odbc\_stmt.c:2421)

## Las columnas de un tipo de datos definido por el usuario no se migran correctamente

Cuando se replica desde un origen de PostgreSQL, AWS DMS crea la tabla de destino con los mismos tipos de datos para todas las columnas, además de las columnas con los tipos de datos definidos por el usuario. En estos casos, el tipo de datos se crea como de "caracteres variables" en el destino.

## Error que indica que no se ha seleccionado ningún esquema de creación

El error "SQL\_ERROR SqlState: 3F000 NativeError: 7 Message: ERROR: no schema has been selected to create in" puede aparecer cuando la asignación de su tabla JSON contiene un valor comodín para el esquema pero la base de datos de origen no admite dicho valor.

## No se están replicando las eliminaciones y las actualizaciones en una tabla mediante CDC

Las operaciones de eliminación y actualización durante la captura de datos de cambios (CDC) se omiten si la tabla de origen no tiene una clave primaria. AWS DMS admite la captura de datos de cambios (CDC) con las tablas de PostgreSQL con claves primarias; si una tabla no tiene un clave primaria, los registros WAL no incluirán una imagen anterior de la fila de la base de datos y AWS DMS no podrá actualizar la tabla. Cree una clave principal en la tabla de origen si desea que se repliquen las operaciones de eliminación.

## Las instrucciones TRUNCATE no se están propagando

Cuando se usa la captura de datos de cambios (CDC), AWS DMS no admite operaciones TRUNCATE.

## Impedir que PostgreSQL capture instrucciones DDL

Puede impedir que un punto de enlace de destino de PostgreSQL capture instrucciones DDL añadiendo la siguiente instrucción Extra Connection Attribute. El parámetro Extra Connection Attribute está disponible en la pestaña Advanced del punto de enlace de origen.

captureDDLS=N

## Selección del esquema donde crear los objetos de base de datos para capturar instrucciones DDL

Puede controlar en qué esquema se crean los objetos de la base de datos relacionados con la captura de instrucciones DDL. Añada la siguiente instrucción Extra Connection Attribute. El parámetro Extra Connection Attribute está disponible en la pestaña Advanced del punto de enlace de destino.

```
ddlArtifactsSchema=xyzddlschema
```

## Ausencia de tablas de Oracle después de migrar a PostgreSQL

Oracle utiliza nombres de tabla en mayúsculas, mientras que PostgreSQL utiliza nombres de tabla en minúsculas. Al migrar de Oracle a PostgreSQL es muy probable que necesite proporcionar reglas de transformación bajo la sección de asignación de tablas de su tarea para convertir las mayúsculas y minúsculas en los nombres de las tablas.

Sus tablas y sus datos siguen siendo accesibles; si migró sus tablas sin utilizar reglas de transformación para convertir las mayúsculas y minúsculas en los nombres de las tablas, deberá encerrar los nombres de sus tablas entre comillas cuando haga referencia a ellas.

## La tarea de uso de vista como origen no tiene filas copiadas

AWS DMS no admite vistas como punto de enlace de origen de PostgreSQL.

## Solución de problemas específicos de Microsoft SQL Server

Los siguientes problemas surgen específicamente al usar AWS DMS con bases de datos de Microsoft SQL Server.

### Temas

- [Permisos especiales para que la cuenta de usuario AWS DMS utilice CDC \(p. 381\)](#)
- [Errores al capturar cambios para una base de datos de SQL Server \(p. 381\)](#)
- [Faltan columnas de identidad \(p. 382\)](#)
- [Error debido a que SQL Server no admite publicaciones \(p. 382\)](#)
- [Cambios que no aparecen en el destino \(p. 382\)](#)

## Permisos especiales para que la cuenta de usuario AWS DMS utilice CDC

La cuenta de usuario utilizada con AWS DMS requiere la función SysAdmin de SQL Server para funcionar correctamente cuando se utiliza la captura de datos de cambios (CDC). CDC para SQL Server solo puede usarse con bases de datos locales o con bases de datos en instancias EC2.

## Errores al capturar cambios para una base de datos de SQL Server

Los errores durante la captura de datos de cambios (CDC) pueden indicar con frecuencia que no se cumple alguno de los requisitos previos. Por ejemplo, el requisito que más comúnmente no se tiene en

cuenta es el requisito previo de hacer una copia de seguridad completa de la base de datos. El registro de tareas refleja esta omisión con el siguiente error:

```
SOURCE_CAPTURE E: No FULL database backup found (under the 'FULL' recovery model).  
To enable all changes to be captured, you must perform a full database backup.  
120438 Changes may be missed. (sqlserver_log_queries.c:2623)
```

Revise los requisitos previos indicados para usar SQL Server como origen [Uso de una base de datos de Microsoft SQL Server como origen para AWS DMS \(p. 108\)](#).

## Faltan columnas de identidad

AWS DMS no es compatible con columnas de identidad al crear un esquema de destino. Debe añadirlas después de la primera vez que se haya completado la carga.

## Error debido a que SQL Server no admite publicaciones

El error siguiente se genera cuando se utiliza SQL Server Express como un punto de enlace de origen:

```
RetCode: SQL_ERROR SqlState: HY000 NativeError: 21106  
Message: This edition of SQL Server does not support publications.
```

AWS DMS no es compatible actualmente con SQL Server Express como origen o destino.

## Cambios que no aparecen en el destino

Para capturar los cambios de forma coherente, AWS DMS precisa que una base de datos de SQL Server de origen esté en modo de recuperación de datos "FULL" o "BULK LOGGED". No se admite el modelo "SIMPLE".

El modelo de recuperación de SIMPLE registra la información mínima necesaria para permitir a los usuarios recuperar su base de datos. Todas las entradas de registro inactivas se truncan automáticamente cuando se genera un punto de control. Si bien se siguen registrando todas las operaciones, en cuanto se genera un punto de comprobación, el registro se trunca automáticamente, lo que significa que queda disponible para reutilizarse y pueden sobrescribirse entradas de registro más antiguas. Al sobrescribirse entradas del registro, los cambios no pueden capturarse y por eso AWS DMS no admite el modelo de recuperación de datos SIMPLE. Para obtener información sobre otros requisitos previos necesarios para usar SQL Server como origen, consulte [Uso de una base de datos de Microsoft SQL Server como origen para AWS DMS \(p. 108\)](#).

# Solución de problemas específicos de Amazon Redshift

Los siguientes problemas surgen específicamente al usar AWS DMS con bases de datos de Amazon Redshift.

Temas

- Carga en un clúster de Amazon Redshift en una región diferente de la de la instancia de replicación de AWS DMS (p. 383)
- Error por existir ya la relación "awsdms\_apply\_exceptions" (p. 383)
- Errores con tablas cuyo nombre comienza con "awsdms\_changes" (p. 383)
- Visualización de tablas en clúster con nombres como dms.awsdms\_changes00000000XXXX (p. 383)
- Permisos necesarios para trabajar con Amazon Redshift (p. 383)

## Carga en un clúster de Amazon Redshift en una región diferente de la de la instancia de replicación de AWS DMS

Esto no se puede hacer. AWS DMS exige que la instancia de replicación de AWS DMS y el clúster de Redshift estén en la misma región.

### Error por existir ya la relación "awsdms\_apply\_exceptions"

El error que indica que la relación "awsdms\_apply\_exceptions" ya existe a menudo se produce cuando un punto de enlace de Redshift se especifica como punto de enlace de PostgreSQL. Para solucionar este problema, modifique el punto de enlace y cambie Target engine a "redshift".

### Errores con tablas cuyo nombre comienza con "awsdms\_changes"

Los mensajes de error relacionados con tablas cuyos nombres comienzan con "awsdms\_changes" suelen producirse cuando dos tareas que intentan cargar datos en el mismo clúster de Amazon Redshift se ejecutan al mismo tiempo. Debido a la forma en que se nombran las tablas temporales, las tareas simultáneas pueden entrar en conflicto al actualizar la misma tabla.

### Visualización de tablas en clúster con nombres como dms.awsdms\_changes00000000XXXX

AWS DMS crea tablas temporales cuando los datos se cargan a partir de archivos almacenados en S3. El nombre de esas tablas temporales lleva el prefijo "dms.awsdms\_changes". Estas tablas son necesarias para que AWS DMS pueda almacenar los datos la primera vez que se cargan y antes de colocarlos en la tabla de destino final.

### Permisos necesarios para trabajar con Amazon Redshift

Para utilizar AWS DMS con Amazon Redshift, la cuenta de usuario que utilice para acceder a Amazon Redshift debe tener los permisos siguientes:

- CRUD (seleccionar, insertar, actualizar y eliminar)
- Cargar masiva

- Crear, modificar y eliminar (si lo requiere la definición de la tarea)

Para ver todos los requisitos previos necesarios para utilizar Amazon Redshift como destino, consulte [Uso de una base de datos de Amazon Redshift como destino para AWS Database Migration Service \(p. 175\)](#).

## Solución de problemas específicos de Amazon Aurora MySQL

Los siguientes problemas surgen específicamente al usar AWS DMS con bases de datos de Amazon Aurora MySQL.

### Temas

- [Error por campos CHARACTER SET UTF8 terminados por ',' entre líneas "" terminadas por '\n' \(p. 384\)](#)

## Error por campos CHARACTER SET UTF8 terminados por ',' entre líneas "" terminadas por '\n'

Si utiliza Amazon Aurora MySQL como destino y aparece un error como el siguiente en los registros, esto suele significar que ANSI\_QUOTES forma parte del parámetro SQL\_MODE. Si ANSI\_QUOTES forma parte del parámetro SQL\_MODE, las comillas dobles se gestionan como comillas sencillas y dan problemas al ejecutar una tarea. Para solucionar este error, elimine ANSI\_QUOTES del parámetro SQL\_MODE.

```
2016-11-02T14:23:48 [TARGET_LOAD ]E: Load data sql statement. load data local infile
"/rdsdbdata/data/tasks/7X04FJHCVON7TYTLQ6RX3CQHDU/data_files/4/LOAD000001DF.csv" into
table
`VOSPUSER`.`SANDBOX_SRC_FILE` CHARACTER SET UTF8 fields terminated by ','
enclosed by '\"' lines terminated by '\n'(`SANDBOX_SRC_FILE_ID`, `SANDBOX_ID`,
`FILENAME`, `LOCAL_PATH`, `LINES_OF_CODE`, `INSERT_TS`, `MODIFIED_TS`, `MODIFIED_BY`,
`RECORD_VER`, `REF_GUID`, `PLATFORM_GENERATED`, `ANALYSIS_TYPE`, `SANITIZED`, `DYN_TYPE`,
`CRAWL_STATUS`, `ORIG_EXEC_UNIT_VER_ID`); (provider_syntax_manager.c:2561)
```

# Prácticas recomendadas para AWS Database Migration Service

Para sacar el máximo partido de AWS Database Migration Service (AWS DMS), consulte las recomendaciones de esta sección sobre la manera más provechosa de migrar sus datos.

## Temas

- Mejora del rendimiento de una migración con AWS DMS (p. 385)
- Selección del tamaño óptimo de una instancia de replicación (p. 387)
- Reducción de la carga de trabajo en su base de datos de origen (p. 388)
- Uso del registro de tareas para solucionar problemas de migración (p. 388)
- Conversión del esquema (p. 388)
- Migración de objetos binarios grandes (LOB) (p. 389)
- Replicación continua (p. 390)
- Cambio del usuario y esquema para un destino Oracle (p. 390)
- Cambio de espacios de tabla de tabla e índice para un destino Oracle (p. 391)
- Mejora del desempeño al migrar tablas de gran tamaño (p. 392)

## Mejora del rendimiento de una migración con AWS DMS

Hay una serie de factores que afectan al desempeño del proceso de migración con AWS DMS:

- Disponibilidad de recursos en el origen
- Desempeño de la red disponible
- Capacidad de los recursos del servidor de replicación
- Capacidad del sistema de destino para incorporar cambios
- El tipo y la distribución de los datos de origen
- El número de objetos que se van a migrar

En nuestras pruebas, hemos tardado entre 12 y 13 horas en migrar un terabyte de datos con una sola tarea de AWS DMS y en las condiciones ideales. Estas condiciones ideales incluían el uso de bases de datos de origen que se ejecutaban en Amazon EC2 y en Amazon RDS con bases de datos de destino en Amazon RDS, todas ellas en la misma zona de disponibilidad. Las bases de datos de origen incluían una cantidad representativa de datos distribuidos de una forma relativamente uniforme y unas cuantas tablas de gran tamaño que contenían hasta 250 GB de datos. Los datos de origen no contenían tipos de datos complejos, como BLOB.

Puede mejorar el desempeño si se siguen algunas o todas las prácticas recomendadas que se mencionan a continuación. Si se puede utilizar o no una de estas prácticas depende en gran parte en su caso de uso específico. Comentaremos las limitaciones, según proceda.

### Carga de varias tablas en paralelo

De forma predeterminada, AWS DMS carga ocho tablas a la vez. Podrá observar cierta mejora en el desempeño si aumenta ligeramente este valor cuando utilice un servidor de replicación muy grande, como una instancia dms.c4.xlarge o más grande. No obstante, llegará un momento en que

si sigue aumentando este paralelismo, el desempeño será inferior. Si el servidor de replicación es relativamente pequeño, como dms.t2.medium, le recomendamos que reduzca el número de tablas cargadas en paralelo.

Para cambiar este número en la consola de administración de AWS, abra la consola, elija Tasks (Tareas), elija crear o modificar una tarea y, a continuación, elija Advanced Settings (Configuración avanzada). En Tuning Settings (Configuración de ajuste), cambie la opción Maximum number of tables to load in parallel (Número máximo de tablas que se pueden cargar en paralelo).

Para cambiar este número a través de la interfaz de línea de comandos (CLI) de AWS, cambie el parámetro `MaxFullLoadSubTasks` en `TaskSettings`.

#### Trabajo con índices, disparadores y límites de integridad referencial

Los índices, los disparadores y los límites de integridad referencial pueden afectar al desempeño de la migración y hacer que la migración provoque un error. La forma en que esto afecta a la migración depende de si la tarea de replicación es una tarea de carga completa o una tarea de replicación continua (CDC).

Para una tarea de carga completa, le recomendamos que elimine los índices de clave primaria, los índices secundarios, los límites de integridad referencial y los disparadores del lenguaje de manipulación de datos (DML). Si lo prefiere, puede retrasar su creación hasta que las tareas de carga completa se hayan completado. No necesita índices durante una tarea de carga completa y los índices generarán una sobrecarga de mantenimiento si están presentes. Dado que la tarea de carga completa carga grupos de tablas de una vez, se infringen los límites de integridad referencial. Del mismo modo, los disparadores `INSERT`, `UPDATE` y `DELETE` pueden producir errores, por ejemplo, si se activa una inserción de fila para una tabla que se haya cargado de forma masiva previamente. Otros tipos de disparadores también afectan al desempeño debido al procesamiento añadido.

Puede crear índices de clave principal y secundarios antes de una tarea de carga completa si los volúmenes de datos son relativamente pequeños y el tiempo de migración adicional no es un problema. Los límites de integridad referencial y los disparadores deben deshabilitarse siempre.

Para una tarea de carga completa + CDC, le recomendamos que añada índices secundarios antes de la fase de CDC. Dado que AWS DMS utiliza la replicación lógica, deben implementarse índices secundarios que admiten operaciones de DML para evitar los análisis de tablas completas. Puede detener la tarea de replicación antes de la fase de CDC para crear índices, disparadores y límites de integridad referencial antes de reiniciar la tarea.

#### Inhabilitar los backups y el registro de transacciones

Cuando se migra a una base de datos de Amazon RDS, es conveniente desactivar los backup y el despliegue Multi-AZ en el sistema de destino hasta que todo esté preparado para realizar el traspaso. Del mismo modo, cuando se migra a un sistema que no es Amazon RDS, es aconsejable inhabilitar los registros en el destino hasta después del momento del traspaso.

#### Usar varias tareas

En ocasiones, el uso de varias tareas para una sola migración puede mejorar el desempeño. Si tiene conjuntos de tablas que no participan en transacciones comunes, seguramente podrá dividir el proceso de migración en varias tareas. La coherencia transaccional se mantiene dentro de una tarea, por lo que es importante que las tablas de tareas aparte no participen en transacciones comunes. Además, cada tarea leerá de manera independiente la secuencia de transacciones, por lo que habrá que tener la precaución de no exigir demasiado a la base de datos de origen.

Puede utilizar varias tareas para crear secuencias de replicación independientes, lo que le permite paralelizar las lecturas en el origen, los procesos en la instancia de replicación y las escrituras en la base de datos de destino.

#### Cómo optimizar el procesamiento de cambios

De forma predeterminada, AWS DMS procesa los cambios en un modo transaccional, para garantizar la integridad transaccional. Si puede permitirse interrupciones temporales en la integridad de las

transacciones, active la opción de aplicación optimizada por lotes. Para resultar más eficaz, esta opción agrupa las transacciones y las aplica en lotes. La opción de aplicación optimizada por lotes infringe casi siempre los límites de integridad referencial, por lo tanto, debe desactivarlos durante el proceso de migración y activarlos de nuevo en el proceso de traslado.

## Selección del tamaño óptimo de una instancia de replicación

La selección de la instancia de replicación adecuada depende de varios factores del caso de uso. Para ayudar a entender cómo se utilizan los recursos de instancias de replicación, consulte la siguiente explicación. Trata la situación habitual de una tarea de carga completa + CDC.

Durante una tarea de carga completa, AWS DMS carga las tablas de manera individual. De forma predeterminada, se cargan ocho tablas cada vez. AWS DMS captura los cambios en curso al origen durante una tarea de carga completa de modo que los cambios se puedan aplicar más adelante en el punto de enlace de destino. Los cambios se almacenan en caché en la memoria y, en caso de agotarse la memoria disponible, se almacenan en la memoria caché del disco. Al completarse una tarea de carga completa para una tabla, AWS DMS aplica inmediatamente los cambios almacenados en caché a la tabla de destino.

Después de que se hayan aplicado todos los cambios en la memoria caché pendientes para una tabla, el punto de enlace de destino se encuentra en un estado coherente desde el punto de vista transaccional. En este momento, el destino está sincronizado con el punto de enlace de origen con respecto a los últimos cambios almacenados en caché. AWS DMS a continuación, comienza la replicación continua entre el origen y el destino. Para ello, AWS DMS toma las operaciones de cambio de los registros de la transacción de origen y las aplica al destino de manera coherente desde el punto de vista transaccional (suponiendo que la aplicación optimizada por lotes no está seleccionada). AWS DMS transmite los cambios en curso a través de la memoria a la instancia de replicación, si es posible. De lo contrario, AWS DMS escribe los cambios en el disco en la instancia de replicación hasta que pueden aplicarse al destino.

El usuario tiene cierto control sobre la forma en que la instancia de replicación gestiona el procesamiento de los cambios y sobre cómo se utiliza la memoria en dicho proceso. Para obtener más información acerca de cómo ajustar el procesamiento de cambios, consulte [Configuración de los ajustes del procesamiento de cambios \(p. 269\)](#).

Por la explicación anterior puede verse que la cantidad total de memoria disponible es una consideración fundamental. Si la instancia de replicación tiene memoria suficiente para que AWS DMS pueda transmitir los cambios en caché y en curso sin escribirlos en el disco, el rendimiento de la migración aumenta considerablemente. Del mismo modo, al configurar la instancia de replicación con suficiente espacio en disco para posibilitar el almacenamiento en caché de los cambios y el almacenamiento de logs también aumenta el desempeño. El número máximo de IOPS posible depende del tamaño de disco seleccionado.

Tenga en cuenta los siguientes son factores a la hora de elegir una clase de instancia de replicación y el almacenamiento en disco disponible:

- Tamaño de la tabla: las tablas grandes tardan más en cargarse y por eso las transacciones en estas tablas deben almacenarse en la caché hasta que se cargue la tabla. Después de cargada una tabla, estas transacciones en la caché se aplican y dejan de conservarse en el disco.
- Actividad de lenguaje de manipulación de datos (DML): una base de datos ocupada genera más transacciones. Estas transacciones deben almacenarse en la caché hasta que se cargue la tabla. Las transacciones para una única tabla se aplican lo antes posible tras cargarse la tabla, hasta que todas las tablas estén cargadas.
- Tamaño de transacción: las transacciones de ejecución prolongada pueden generar muchos cambios. Para conseguir el mejor rendimiento, si AWS DMS aplica los cambios en modo transaccional, debe haber suficiente memoria disponible para transmitir todos los cambios de la transacción.

- Tamaño total de la migración: las migraciones grandes tardan más tiempo y generan un número proporcionalmente mayor de archivos de log.
- Número de tareas: cuantas más tareas haya, más memoria caché será necesaria y más archivos de log se generarán.
- Objetos grandes: las tablas con LOB tardan más en cargarse.

Algunas pruebas demuestran que los archivos de registro consumen la mayor parte del espacio que necesita AWS DMS. Los ajustes de almacenamiento predeterminados suelen ser suficientes.

Sin embargo, las instancias de replicación que ejecutan varias tareas pueden requerir más espacio en el disco. Asimismo, si la base de datos incluye tablas grandes y activas, es posible que tenga que incrementar el espacio en disco para las transacciones que se almacenan en la caché del disco durante una tarea de carga completa. Por ejemplo, si su carga tarda 24 horas y produce 2 GB de transacciones por hora, debería confirmar que dispone de 48 GB de espacio para las transacciones almacenadas en la caché. Además, cuanto más espacio de almacenamiento asigne a la instancia de replicación, mayor será el número de IOPS que se obtiene.

Las directrices anteriores no cubren todas las situaciones posibles. Es sumamente importante tener en cuenta los detalles específicos de su caso de uso particular al determinar el tamaño de la instancia de replicación. Cuando la migración se esté ejecutando, monitorice la CPU, la memoria que se puede liberar, la cantidad de almacenamiento libre y las IOPS de su instancia de replicación. En función de los datos que recopile, puede ampliar o reducir las dimensiones de la instancia de replicación según sea necesario.

## Reducción de la carga de trabajo en su base de datos de origen

AWS DMS utiliza algunos recursos en la base de datos de origen. Durante una tarea de carga completa, AWS DMS analiza por completo la tabla de origen para cada una de las tablas procesadas en paralelo. Asimismo, cada tarea que se crea como parte de una migración comprueba si existen cambios en el origen como parte del proceso de CDC. Para que AWS DMS realice la CDC para algunos orígenes, como Oracle, puede que tenga que aumentar la cantidad de datos escritos en el registro de cambios de su base de datos.

Si descubre que está sobrecargando la base de datos de origen, puede reducir el número de tareas o de tablas por cada tarea de la migración. Cada tarea obtiene los cambios del origen de forma independiente, por lo que la consolidación de las tareas puede reducir la carga de trabajo de la captura de cambios.

## Uso del registro de tareas para solucionar problemas de migración

En algunos casos, AWS DMS puede encontrar problemas para los que los mensajes de advertencia o de error aparecen solo en el registro de tareas. En concreto, los problemas de truncamiento de datos o de rechazo de filas por infracciones en la clave externa solo están escritos en el log de tareas. Por lo tanto, asegúrese de revisar este registro al migrar una base de datos. Para habilitar la visualización del log de tareas, configure Amazon CloudWatch como parte de la creación de tareas.

## Conversión del esquema

AWS DMS no convierte ni los esquemas ni el código. Si desea convertir un esquema existente en otro motor de base de datos, puede utilizar la herramienta Schema Conversion Tool (AWS SCT) de AWS.

AWS SCT convierte los objetos, tablas, índices, vistas, disparadores y otros objetos del sistema de origen en el formato de lenguaje de definición de datos (DDL) de destino. También puede utilizar AWS SCT para convertir la mayor parte del código de aplicación, como PL/SQL o TSQL, en el lenguaje de destino equivalente.

Puede obtener AWS SCT como descarga gratuita de AWS. Para obtener más información sobre la herramienta AWS SCT, consulte la [Guía de usuario de AWS Schema Conversion Tool](#).

Si los puntos de enlace de origen y de destino se corresponden con el mismo motor de base de datos, puede utilizar herramientas como Oracle SQL Developer, MySQL Workbench o PgAdmin4 para trasladar su esquema.

## Migración de objetos binarios grandes (LOB)

En general, AWS DMS migra los datos LOB en dos fases.

1. AWS DMS crea una nueva fila de la tabla de destino y la rellena con todos los datos, excepto el valor de LOB asociado.
2. AWS DMS actualiza la fila en la tabla de destino con los datos de LOB.

Este proceso de migración para los LOB requiere que, durante el proceso, todas las columnas de LOB de la tabla de destino sean NULLABLE. Esto es así aunque las columnas de LOB no sean NULLABLE en la tabla de origen. Si AWS DMS crea las tablas de destino, define las columnas de LOB como NULLABLE de forma predeterminada. Si crea las tablas de destino utilizando otro mecanismo, como la importación o exportación, debe asegurarse de que las columnas de LOB sean NULLABLE antes de iniciar la tarea de migración.

Este requisito tiene una excepción. Supongamos que realiza una migración homogénea desde un origen de Oracle a un destino de Oracle y elige Limited Lob mode (Modo de LOB limitado). En este caso, la totalidad de la fila se rellena a la vez, incluido cualquier valor de LOB. En tal caso, AWS DMS puede crear las columnas de LOB de la tabla de destino con restricciones NOT NULL, en caso necesario.

## Uso del modo LOB limitado

AWS DMS utiliza dos métodos que equilibran el rendimiento y la comodidad cuando la migración contiene valores LOB.

- Limited LOB mode (Modo LOB limitado) migra todos los valores LOB hasta un límite de tamaño especificado por el usuario (el valor predeterminado es 32 KB). Los valores LOB que superen el límite de tamaño deben migrarse manualmente. Normalmente, el valor predeterminado de Limited LOB mode (Modo de LOB limitado) para todas las tareas de migración proporciona el mejor desempeño. Sin embargo, debe asegurarse de que la configuración del parámetro Max LOB size (Tamaño máximo de LOB) sea correcta. Este parámetro debe establecerse en el tamaño de LOB más grande para todas las tablas.
- Full LOB mode (Modo de LOB completo) migra todos los datos LOB de las tablas, independientemente de su tamaño. Full LOB mode (Modo de LOB completo) resulta conveniente porque traslada todos los datos LOB de las tablas, si bien el proceso puede tener un impacto significativo en el desempeño.

Para algunos motores de base de datos, como PostgreSQL, AWS DMS trata los tipos de datos JSON como LOB. Asegúrese de que si ha elegido Limited LOB mode (Modo de LOB limitado), la opción Max LOB size (Tamaño máximo de LOB) esté definida en un valor que no haga que los datos JSON se truncuen.

AWS DMS es totalmente compatible en cuanto al uso de tipos de datos de objetos grandes (BLOB, CLOB y NCLOB). Los siguientes son puntos de enlace de origen totalmente compatibles con objetos LOB:

- Oracle
- Microsoft SQL Server
- ODBC

Los siguientes son puntos de enlace de destino totalmente compatibles con objetos LOB:

- Oracle
- Microsoft SQL Server

El siguiente punto de enlace de destino tiene compatibilidad limitada con objetos LOB. No puede utilizar un tamaño LOB ilimitado para este punto de enlace de destino.

- Amazon Redshift

Si los puntos de enlace son totalmente compatibles con objetos LOB, también puede fijar un límite de tamaño para los tipos de datos LOB.

## Replicación continua

AWS DMS proporciona la replicación continua de los datos, a la vez que mantiene las bases de datos de origen y de destino sincronizadas. Solo se replica una cantidad limitada de lenguaje de definición de datos (DDL). AWS DMS no propaga elementos tales como índices, usuarios, privilegios, procedimientos almacenados y otros cambios de la base de datos no relacionados directamente con los datos de tabla.

Si tiene previsto utilizar la replicación continua, debe habilitar la opción Multi-AZ (Multi-AZ) al crear la instancia de replicación. Al elegir la opción Multi-AZ (Multi-AZ) consigue alta disponibilidad y soporte de conmutación por error para la instancia de replicación. Sin embargo, esta opción puede afectar al desempeño.

## Cambio del usuario y esquema para un destino Oracle

Al utilizar Oracle como destino, AWS DMS migra los datos al esquema propiedad del usuario del punto de enlace de destino.

Por ejemplo, supongamos que va a migrar un esquema denominado **PERFDATA** a un punto de enlace de destino de Oracle y que el nombre de usuario del punto de enlace de destino es **MASTER**. AWS DMS se conectará al destino de Oracle como **MASTER** y rellena el esquema **MASTER** con objetos de base de datos de **PERFDATA**.

Para anular este comportamiento, debe proporcionar una transformación de esquema. Por ejemplo, si desea migrar los objetos de esquema **PERFDATA** a un esquema **PERFDATA** en el punto de enlace de destino, puede utilizar la siguiente transformación:

```
{  
    "rule-type": "transformation",  
    "rule-id": "2",  
    "rule-name": "2",  
    "object-locator": {
```

```
        "schema-name": "PERFDATA"
    },
    "rule-target": "schema",
    "rule-action": "rename",
    "value": "PERFDATA"
}
```

Para obtener más información sobre transformaciones, consulte [Especificación de selección de tabla y transformaciones mediante correspondencia de tabla con JSON \(p. 288\)](#).

## Cambio de espacios de tabla de tabla e índice para un destino Oracle

Al utilizar Oracle como destino, AWS DMS migra los espacios de tabla, tal y como se define en el origen si el origen es también de Oracle. Sin embargo, si el origen no es de Oracle, AWS DMS migra todas las tablas e índices a los espacios de tabla de tabla e índice predeterminados en el destino.

Por ejemplo, suponga que su origen es un motor de base de datos distinto de Oracle. Todas las tablas de destino e índices se migran a los mismos espacios de tabla predeterminados.

Para anular este comportamiento, tiene que proporcionar las transformaciones de espacio de tabla correspondientes. Por ejemplo, suponga que desea migrar tablas e índices a espacios de tabla de tabla e índice en el destino de Oracle que se nombran según el esquema del origen. En este caso, puede utilizar transformaciones similares a las siguientes. Aquí, el esquema en el origen se denomina `INVENTORY` y los espacios de tabla de tabla e índice correspondiente en el destino se llaman `INVENTORYTBL` e `INVENTORYIDX`, respectivamente.

```
{
    "rule-type": "transformation",
    "rule-id": "3",
    "rule-name": "3",
    "rule-action": "rename",
    "rule-target": "table-tablespace",
    "object-locator": {
        "schema-name": "INVENTORY",
        "table-name": "%",
        "table-tablespace-name": "%"
    },
    "value": "INVENTORYTBL"
},
{
    "rule-type": "transformation",
    "rule-id": "4",
    "rule-name": "4",
    "rule-action": "rename",
    "rule-target": "index-tablespace",
    "object-locator": {
        "schema-name": "INVENTORY",
        "table-name": "%",
        "index-tablespace-name": "%"
    },
    "value": "INVENTORYIDX"
}
```

Para obtener más información sobre transformaciones, consulte [Especificación de selección de tabla y transformaciones mediante correspondencia de tabla con JSON \(p. 288\)](#).

# Mejora del desempeño al migrar tablas de gran tamaño

Si desea mejorar el desempeño al migrar una tabla de gran tamaño, puede dividir la migración en más de una tarea. Para dividir la migración en varias tareas usando el filtrado de filas, utilice una clave o una clave de partición. Por ejemplo, si tiene un ID entero de clave principal de 1 a 8 000 000, puede crear ocho tareas y usar el filtrado de filas para migrar un millón de registros por cada una.

Para aplicar el filtrado de filas en la consola de administración de AWS, abra la consola, elija Tasks (Tareas) y cree una nueva tarea. En la sección Table mappings (Mapeos de tabla), añada un valor para Selection Rule (Regla de selección). A continuación, puede añadir un filtro de columna con una condición inferior o igual a, superior o igual a, igual a o de rango (entre dos valores). Para obtener más información acerca del filtrado de columnas, consulte [Especificación de selección de tabla y transformaciones mediante correspondencia de tabla desde la consola \(p. 283\)](#).

También puede migrar los datos en función de la fecha, si tiene una tabla grande particionada por fecha. Por ejemplo, supongamos que tiene una tabla particionada por mes y solo se actualizan los datos correspondientes al mes actual. En este caso, puede crear una tarea de carga completa para cada partición mensual estática y crear una tarea de carga completa + CDC para la partición actualizada actualmente.

# Referencia de AWS DMS

En esta sección de referencia encontrará información adicional que puede necesitar cuando utilice AWS Database Migration Service (AWS DMS), incluida información sobre la conversión de tipos de datos.

AWS DMS mantiene los tipos de datos cuando realiza una migración de base de datos homogénea en la que el origen y el destino utilizan el mismo tipo de motor. Cuando realice una migración heterogénea, en la que migra desde un tipo de motor de base de datos a otro, los tipos de datos se convierten a un tipo de datos intermedio. Para ver cómo aparecen los tipos de datos en la base de datos de destino, consulte las tablas de tipo de datos para los motores de base de datos de origen y de destino.

Cuando se migra una base de datos, hay ciertas cosas importantes sobre los tipos de datos que se deben tener en cuenta:

- El tipo de datos FLOAT es intrínsecamente una aproximación. Cuando se inserta un valor específico en FLOAT, es posible que se represente de forma distinta en la base de datos. Esta diferencia se debe a que FLOAT no es un tipo de datos exacto, como los tipos de datos decimales NUMBER o NUMBER (p,s). Como resultado, el valor interno de FLOAT que se almacena en la base de datos puede ser diferente del valor que ha insertado. Por tanto, el valor migrado de un valor FLOAT puede no coincidir exactamente con el valor de la base de datos de origen.

Para obtener más información sobre este problema, consulte los siguientes artículos:

- [IEEE floating point](#) en Wikipedia
- [Representación IEEE de punto flotante](#) en MSDN
- [Por qué los números de punto flotante pierden precisión](#) en MSDN

## Temas

- [Tipos de datos para AWS Database Migration Service \(p. 393\)](#)

## Tipos de datos para AWS Database Migration Service

AWS Database Migration Service utiliza tipos de datos integrados para migrar datos desde un tipo de motor de base de datos a un tipo de motor de base de datos diferente. En la siguiente tabla se muestran los tipos de datos integrados y sus descripciones.

Tipos de datos de AWS DMS	Descripción
STRING	Una cadena de caracteres.
WSTRING	Una cadena de caracteres de dos bytes.
BOOLEANO	Un valor booleano.
BYTES	Un valor de datos binarios.
FECHA	Un valor de fecha: año, mes, día.
TIME	Un valor de tiempo: hora, minutos, segundos.

Tipos de datos de AWS DMS	Descripción
DATETIME	Un valor de marca de tiempo: año, mes, día, hora, minuto, segundo, fracciones de segundo. Las fracciones de segundo tienen una escala máxima de 9 dígitos.
INT1	Un entero firmado de un solo byte.
INT2	Un entero firmado de dos bytes.
INT4	Un entero firmado de cuatro bytes.
INT8	Un entero firmado de ocho bytes.
NUMERIC	Un valor numérico exacto con una precisión y escala fijas.
REAL4	Un valor de punto flotante de precisión única.
REAL8	Un valor de punto flotante de doble precisión.
UINT1	Un número entero sin firmar de un byte.
UINT2	Un número entero sin firmar de dos bytes.
UINT4	Un número entero sin firmar de cuatro bytes.
UINT8	Un número entero sin firmar de ocho bytes.
BLOB	Objeto binario grande. Este tipo de datos solo se puede utilizar con los puntos de enlace de Oracle.
CLOB	Objeto de caracteres de gran tamaño.
NCLOB	Objeto de caracteres nativos de gran tamaño.

# Notas de la versión de AWS DMS

A continuación, puede encontrar las notas de distintas versiones de AWS Database Migration Service (AWS DMS).

## Note

Ahora puede actualizar las instancias de replicación directamente desde la versión 2.4.x a la más reciente.

## Notas de la versión 3.1.3 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 3.1.3 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Soporte de formato de almacenamiento Apache Parquet para los destinos Amazon S3	Ahora puede seleccionar y configurar archivos .parquet como formato de almacenamiento para objetos de destino de S3.
Clave de cifrado de AWS Key Management Service (AWS KMS) para datos de destino de Amazon Redshift y Amazon S3	Ahora puede utilizar las claves de KMS para cifrado del lado del servidor de datos para destinos de Amazon Redshift y S3.
Selección mejorada y transformación de datos con mapeo de tablas	Ahora puede seleccionar de forma explícita una tabla y su esquema. También puede transformar tablas y espacios de tabla de índice para un destino de Oracle y actualizar la clave principal de una tabla en cualquier destino.
Compatibilidad con el etiquetado de objetos de destino de Amazon S3	Ahora puede asociar etiquetas con cada objeto escrito en su bucket de destino de S3.
Tipos de datos TIMESTAMP migrados desde Microsoft SQL Server a destinos Amazon Elasticsearch Service o Amazon Kinesis ahora se convierten de una cadena binaria a una cadena hexadecimal (p. ej. 0x654738). Anteriormente, estos valores TIMESTAMP se perdían cuando se mapeaban como datos binarios.	Los tipos de datos TIMESTAMP migrados desde Microsoft SQL Server a destinos Amazon Elasticsearch Service o Amazon Kinesis ahora se convierten de una cadena binaria a una cadena hexadecimal (p. ej. 0x654738). Anteriormente, estos valores TIMESTAMP se perdían cuando se mapeaban como datos binarios.
Transiciones en el nivel de registro DEFAULT o INFO	AWS DMS ahora registra cuando los niveles de registro se vuelven a cambiar a nivel DEFAULT o INFO. Anteriormente, estas transiciones de nivel de registro no se registraban.
Validación de datos y configuración de filtro	La validación de datos ahora respeta la configuración de filtro de filas al validar datos entre el origen y el destino.
Opción de gestión de cadenas para objetos de origen de Amazon S3	Un nuevo atributo de conexión adicional <code>rfc4180</code> está disponible para controlar la forma en que se gestionan las cadenas de origen con comillas dobles ("").

Nueva característica o mejora	Descripción
Instrucción CREATE TABLE AS admitida para los puntos de enlace de origen de Oracle	AWS DMS ahora puede realizar cambios en los datos como resultado de la ejecución de la instrucción CREATE TABLE AS para un punto de enlace de origen de Oracle.

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
17 de diciembre de 2018	Se ha corregido un problema donde la habilitación de SSL provocaba un error de tarea con un origen de Amazon S3 y un destino de PostgreSQL.
11 de diciembre de 2018	Se ha corregido un problema en el que se notificaba de forma incorrecta el `porcentaje completo para MySQL.
11 de noviembre de 2018	Se ha corregido un problema con un origen de Amazon S3 donde las comillas dobles ("") en una cadena provocaban un error de tarea.
6 de noviembre de 2018	Se ha corregido un problema donde un valor delimitador .csv predeterminado en una columna de encabezado para un destino de Amazon S3 no se aplicaba correctamente.
16 de agosto de 2018	Se ha corregido un problema que provocaba un error de restricción duplicado cuando estaba habilitada la recuperación de punto de comprobación.

## Notas de la versión 3.1.2 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 3.1.2 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Compatibilidad con una versión actualizada de IBM Db2 para Windows (Db2 LUW) como origen	A partir de ahora, puede utilizar Db2 LUW Version 11.1 como origen, con todos los Fix Packs.
Compatibilidad con el cálculo de latencia durante el cambio al horario de verano	Ahora, puede recalcular el desfase horario durante el horario de verano cuando usa Oracle o PostgreSQL como origen.

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
19 de julio de 2018	Se ha corregido un problema que se producía cuando PostgreSQL como origen enviaba valores Null como Empty a Oracle durante la captura de datos en modo Full LOB.

Fecha de notificación	Descripción
19 de septiembre de 2018	Se ha corregido un problema por el que los valores <code>Null</code> de las columnas <code>varchar</code> de SQL Server se migraban de forma diferente a todos los destinos.
7 de octubre de 2018	Se ha corregido un problema por el que la configuración <code>LOB</code> no funcionaba cuando existían reglas de transformación.
12 de octubre de 2018	Se ha corregido un problema por el que, en determinadas ocasiones, las tareas de replicación continua con Oracle como origen no podían reanudarse.
12 de octubre de 2018	Se ha corregido un problema por el que, en determinadas ocasiones, las tareas de replicación continua con SQL Server como origen no podían reanudarse.
Varias fechas	Se han solucionado varios problemas con PostgreSQL como origen que estaban presentes en la versión 3.1.1.

## Notas de la versión 3.1.1 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 3.1.1 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Migración de caracteres UTF8 de 4 bytes	AWS DMS ahora es compatible con todos los conjuntos de caracteres de 4 bytes, como UTF8MB4, y así sucesivamente. Esta característica funciona sin cambios de configuración.
Soporte de Microsoft SQL Server 2017 como origen	Soporte añadido para SQL Server 2017 como origen. Para obtener más información, consulte <a href="#">Uso de una base de datos de Microsoft SQL Server como origen para AWS DMS (p. 108)</a> .
Soporte para carga completa de tablas en paralelo	Se ha añadido soporte para carga completa en paralelo de grandes tablas basadas en particiones y subparticiones. Esta característica utiliza un subprocesso de descarga independiente para cada partición o subpartición de tabla para acelerar el proceso de carga masiva. También puede especificar los rangos específicos o las particiones para migrar un subconjunto de datos de la tabla. Fuentes compatibles son Oracle, SQL Server, Sybase, MySQL e IBM Db2 para Linux, UNIX, PostgreSQL y Windows (Db2 LUW). Para obtener más información, consulte <a href="#">Reglas y operaciones de table-settings (p. 300)</a> .
Control de ajustes de objeto grande (LOB) por tabla	Ahora puede controlar los ajustes de LOB por tabla con la configuración de mapeo de tablas adicional. Para obtener más información, consulte <a href="#">Configuración de las tareas de los metadatos de destino (p. 263)</a> . Los orígenes compatibles son Oracle, SQL Server, MySQL y PostgreSQL.
Control de orden para cargar tablas en una sola tarea de migración	A partir de ahora, puede controlar el orden para cargar las tablas con mapeos de tablas en una tarea de migración. Puede especificar el orden etiquetando la tabla con un número entero sin firmar de orden de carga

Nueva característica o mejora	Descripción
	en los mapeos de tablas. Las tablas con los valores de orden de carga más altos se migran en primer lugar. Para obtener más información, consulte <a href="#">Uso de la correspondencia de tablas para especificar la configuración de tareas (p. 283)</a> .
Soporte de actualizaciones de valores de clave primaria cuando se utiliza PostgreSQL como origen	Las actualizaciones de valores de clave primaria ahora se replican cuando se utiliza PostgreSQL como origen para replicación continua.

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
24 de abril de 2018	Se ha solucionado un problema en el que los usuarios no podían crear SQL Azure como punto de enlace de origen para SQL Server 2016.
5 de mayo de 2018	Se ha corregido un problema en el que CHR (0) en un origen de Oracle se migró como CHR(32) en un objetivo de Compatibilidad de Aurora con MySQL.
10 de mayo de 2018	Se ha corregido un problema en el que la replicación continua de Oracle como origen no funcionaba según lo previsto al utilizar Oracle LogMiner para migrar cambios desde standby físico de Oracle.
27 de mayo de 2018	Se ha corregido un problema en el que caracteres de diversos tipos de datos en PostgreSQL se triplicaban durante la migración a PostgreSQL.
12 de junio de 2018	Se ha corregido un problema en el que los datos se cambiaban durante una migración desde TEXT para NCLOB (PostgreSQL a Oracle) debido a diferencias en la forma en que estos motores gestionan valores nulos dentro de una cadena.
17 de junio de 2018	Se ha corregido un problema en el que la tarea de replicación no podía crear claves primarias en una instancia de MySQL de destino versión 5.5 al migrar desde una base de datos MySQL de origen versión 5.5.
23 de junio de 2018	Se ha corregido un problema en el que las columnas JSON se truncaban en un modo LOB completo al migrar desde una instancia de PostgreSQL a Compatibilidad de Aurora con PostgreSQL.
27 de junio de 2018	Se ha corregido un problema en el que la aplicación por lotes de cambios a PostgreSQL como destino producía un error debido a un problema al crear la tabla de cambios netos intermedia en el destino.
30 de junio de 2018	Se ha corregido un problema por el que la marca temporal de MySQL '0000-00-00 00:00:00' no se migraba según lo previsto al realizar una carga completa.
2 de julio de 2018	Se ha corregido un problema en el que una tarea de replicación de DMS no continuaba según lo previsto después de producirse la comutación por error de Aurora MySQL de origen.
9 de julio de 2018	Se ha solucionado un problema con una migración de MySQL a Amazon Redshift donde la tarea producía un error con un error de tipo de columna y datos desconocido.

Fecha de notificación	Descripción
21 de julio de 2018	Se ha corregido un problema en el que caracteres null en una cadena migraban de manera diferente desde SQL Server a PostgreSQL en modos LOB limitado y LOB completo.
23 de julio de 2018	Se ha solucionado un problema en las transacciones de protección en SQL Server como origen rellenaban el registro de transacciones en SQL Server.
26 de julio de 2018	Se ha corregido un problema en el que los valores nulos se migraban como valores vacíos en una migración de avance por volteo de PostgreSQL a Oracle.
Varias fechas	Se han solucionado varios problemas de registro para mantener a los usuarios más informados acerca de la migración mediante registros de Amazon CloudWatch.
Varias fechas	Corregidos varios errores de validación de datos.

## Notas de la versión 2.4.5 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 2.4.5 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Clave de cifrado de KMS para datos de destino de Amazon Redshift y Amazon S3	Ahora puede utilizar las claves de KMS para cifrado del lado del servidor de datos para destinos de Amazon Redshift y S3.
Transiciones en el nivel de registro DEFAULT o INFO	AWS DMS ahora registra cuando los niveles de registro se vuelven a cambiar a nivel DEFAULT o INFO. Anteriormente, estas transiciones de nivel de registro no se registraban.
Validación de datos y configuración de filtro	La validación de datos ahora respeta la configuración de filtro de filas al validar datos entre el origen y el destino.
Opción de gestión de cadenas para objetos de origen de Amazon S3	Un nuevo atributo de conexión adicional <code>rfc4180</code> está disponible para controlar la forma en que se gestionan las cadenas de origen con comillas dobles ("").

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
19 de enero de 2019	Se ha solucionado un problema para puntos de enlace de Oracle donde <code>securityDbEncryptionName</code> no se había establecido correctamente al especificar <code>password</code> y <code>securityDbEncryptionName</code> sin una <code>asm_password</code> .
11 de diciembre de 2018	Se ha corregido un problema en el que se notificaba de forma incorrecta el Porcentaje completo para MySQL.

Fecha de notificación	Descripción
11 de noviembre de 2018	Se ha solucionado un problema con un origen de Amazon S3 donde las comillas dobles ("") en una cadena provocaban un error de tarea.
6 de noviembre de 2018	Se ha corregido un problema donde un valor delimitador .csv predeterminado en una columna de encabezado para un destino de Amazon S3 no se aplicaba correctamente.
16 de agosto de 2018	Se ha corregido un problema que provocaba un error de restricción duplicado cuando la recuperación de punto de comprobación estaba habilitada.

## Notas de la versión 2.4.4 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 2.4.4 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Validación de migraciones con cláusulas de filtro	A partir de ahora, puede validar los datos al migrar un subconjunto de una tabla utilizando filtros de tabla.
actualizaciones de controladores de Open Database Connectivity (ODBC)	El controlador ODBC subyacente para MySQL se ha actualizado a 5.3.11-1 y el controlador ODBC subyacente para Amazon Redshift se ha actualizado a 1.4.2-1010.
Recálculo de latencia en caso de cambios de horario de verano	Ahora, puede recalcular el desfase horario durante el horario de verano para Oracle y PostgreSQL como origen. Los cálculos de latencia de origen y destino son correctos después del cambio de horario de verano.
Conversión del tipo de datos de UUID (SQL Server a MySQL)	Ahora, puede convertir un tipo de datos <b>UNIQUEIDENTIFIER</b> (es decir, un identificador único universal o UUID) a bytes al migrar entre SQL Server como origen y MySQL como destino.
Capacidad de cambiar los modos de cifrado para Amazon S3 como origen y Amazon Redshift como destino	Ahora, puede cambiar el modo de cifrado al migrar entre S3 como origen y Amazon Redshift como destino. Especifique el modo de cifrado con un atributo de conexión. El cifrado en el lado del servidor y AWS KMS son compatibles.

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
17 de julio de 2018	Se ha corregido un problema en el que PostgreSQL como origen enviaba valores nulos como valores vacíos a las bases de datos de Oracle de destino durante la captura de datos de cambios (CDC) en modo de objeto binario grande (LOB) completo.

Fecha de notificación	Descripción
29 de julio de 2018	Se ha corregido un problema en el que tareas de migración de entrada y salida de Amazon S3 no se podían reanudar después de actualizar desde la DMS versión 1.9.0.
5 de agosto de 2018	Se ha corregido un problema en el que el atributo de conexión adicional <code>ResumeFetchForXRows</code> no funcionaba correctamente con una clave primaria <code>VARCHAR</code> para un origen de MySQL.
12 de septiembre de 2018	Se ha corregido un problema en el que DMS con transacciones protegidas de SQL Server bloqueaban el registro de transacciones para reutilizarse.
21 de septiembre de 2018	Se ha solucionado un problema con carga masivas con error desde PostgreSQL como origen a Amazon Redshift como destino. Las tareas con error no registraban un error cuando se interrumpía la carga completa.
3 de octubre de 2018	Se ha corregido un problema en el que una tarea de migración de DMS no fallaba cuando los requisitos previos para la replicación continua no se configuraron correctamente para SQL Server como origen.
Varias fechas	Se han corregido múltiples problemas relacionados con la validación de datos y se ha añadido soporte mejorado para validar caracteres UTF-8 multibyte.

## Notas de la versión 2.4.3 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 2.4.3 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Recreación de metadatos de tabla con discordancia	<p>Se ha añadido un nuevo atributo de conexión adicional para puntos de enlace de MySQL: <code>CleanSrcMetadataOnMismatch</code>.</p> <p>Se trata de un atributo booleano que limpia y vuelve a crear la información de metadatos de la tabla en la instancia de replicación cuando se produce una discordancia. Un ejemplo es una situación en la que se ejecuta una instrucción <code>ALTER</code> en el lenguaje de definición de datos (DDL) en una tabla que podría dar como resultado información distinta sobre la tabla en caché en la instancia de replicación. De forma predeterminada, este atributo se establece en <code>false</code>.</p>
Mejoras de rendimiento para validación de datos	<p>Las mejoras incluyen lo siguiente:</p> <ul style="list-style-type: none"> <li>La validación de datos ahora realiza particiones de datos antes de que comience por tanto puede comparar particiones similares y validarlas. Esta versión contiene mejoras en los cambios para obtener lotes de particiones, lo que acelera el tiempo de partición y agiliza la validación de datos.</li> <li>Mejoras para gestionar las diferencias de intercalación de forma automática en función de la configuración de tareas.</li> </ul>

Nueva característica o mejora	Descripción
	<ul style="list-style-type: none"><li>Mejoras para identificar falsos positivos durante la validación, lo que también reduce los falsos positivos durante la fase de cambios en la memoria caché.</li><li>Mejoras de registro general en la validación de datos.</li><li>Las mejoras en las consultas usadas por la validación de datos cuando las tablas tienen claves primarias compuestas.</li></ul>

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
12 de febrero de 2018	Se ha corregido un problema en la replicación continua utilizando la aplicación por lotes donde AWS DMS perdía algunas inserciones como una restricción única en la tabla que se estaba actualizando.
16 de marzo de 2018	Se ha corregido un problema por el que la tarea de migración de Oracle a PostgreSQL se bloqueaba durante la fase de replicación continua, debido a la conmutación por error de Multi-AZ en la instancia de Amazon RDS para instancia de Oracle.

## Notas de la versión 2.4.2 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 2.4.2 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Binary Reader admite Amazon RDS para Oracle durante la captura de datos de cambios	Se ha añadido compatibilidad para utilizar Binary Reader en escenarios de captura de datos de cambios (CDC) desde un origen de Amazon RDS para Oracle durante la replicación continua.
Parámetros adicionales para el comando COPY para Amazon Redshift como destino	Se admiten los siguientes <a href="#">parámetros adicionales del comando copy de Amazon Redshift</a> usando atributos de conexión adicionales. Para obtener más información, consulte <a href="#">Atributos de conexión adicionales al usar Amazon Redshift como destino para AWS DMS (p. 183)</a> . <ul style="list-style-type: none"><li>TRUNCATECOLUMNS</li><li>REMOVEQUOTES</li><li>TRIMBLANKS</li></ul>
Opción de invalidar una tarea de migración cuando se trunca una tabla en un origen de PostgreSQL	Se ha introducido compatibilidad para invalidar una tarea en caso de encontrar un truncamiento en un origen de PostgreSQL al usar la configuración de una nueva tarea. Para obtener más información, consulte la configuración de <code>ApplyErrorFailOnTruncationDdl</code> en la sección <a href="#">Configuración de las tareas de administración de errores (p. 272)</a> .

Nueva característica o mejora	Descripción
Compatibilidad de validación para JSON/JSONB/HSTORE en los puntos de enlace de PostgreSQL	Se ha introducido compatibilidad de validación de datos de las columnas JSON, JSONB y HSTORE para PostgreSQL como origen y destino.
Se ha mejorado el registro de los orígenes MySQL	Se ha mejorado la visibilidad de los problemas en los registros binarios (binlogs) de MySQL durante la captura de datos de cambios (CDC). Los registros ahora muestran claramente un error o una advertencia si hay problemas al acceder a los binlogs de origen de MySQL durante la CDC.
Estadísticas de validación de datos adicionales	Se han añadido más estadísticas de tablas de replicación. Para obtener más información, consulte <a href="#">Estadísticas de la tarea de replicación (p. 335)</a> .

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
14 de enero de 2018	Corregidos todos los problemas relacionados con el tratamiento de las fechas cero (0000-00-00) en destinos de MySQL durante la carga completa y CDC. MySQL no acepta 0000-00-00 (no válido para MySQL) aunque algunos motores sí. Todas estas fechas se convierten en 0101-01-01 para un destino MySQL.
21 de enero de 2018	Corregido un error por el que no se podía migrar una tabla si el nombre de esta contenía el signo \$.
3 de febrero de 2018	Corregido un error por el que una columna JSON de un origen PostgreSQL se truncaba al migrarse a cualquier destino admitido.
12 de febrero de 2018	Corregido un error por el que la tarea de migración no se podía realizar después de una comutación por error en un destino Aurora MySQL.
21 de febrero de 2018	Corregido un error por el que una tarea de migración no podía iniciar su fase de replicación continua después de un problema de conectividad de red.
23 de febrero de 2018	Corregido un error por el que ciertas reglas de transformación en mapeos de tablas causaban el bloqueo de la tarea de migración durante la replicación continua en destinos de Amazon Redshift.
Notificado en varias fechas	Corregidos varios errores de validación de datos: <ul style="list-style-type: none"> <li>• Corregidos errores de tratamiento de cadenas anchas en la validación de orígenes y destinos de Oracle.</li> <li>• Tratamiento de la validación cuando una columna se elimina de una tabla en los mapeos de tablas.</li> <li>• Desempeño de la validación mejorado para orígenes con una alta tasa de cambios.</li> </ul>

# Notas de la versión 2.4.1 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 2.4.1 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Compatibilidad con JSONB para orígenes PostgreSQL	Se ha añadido la posibilidad de realizar la migración de JSONB desde PostgreSQL como origen. JSONB se considera un tipo de datos LOB y requiere que se utilicen los ajustes de LOB adecuados.
Compatibilidad con HSTORE para orígenes PostgreSQL	Se ha añadido la posibilidad de realizar la migración del tipo de datos HSTORE desde PostgreSQL como origen. HSTORE se considera un tipo de datos LOB y requiere que se utilicen los ajustes de LOB adecuados.
Parámetros adicionales para el comando COPY para Amazon Redshift como destino	Se ha introducido compatibilidad con los siguientes <a href="#">parámetros de copy adicionales para Redshift</a> utilizando estos atributos de conexión adicionales. <ul style="list-style-type: none"><li>• ACCEPTANYDATE</li><li>• DATEFORMAT</li><li>• TIMEFORMAT</li><li>• EMPTYASNULL</li></ul>

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
12 de julio de 2017	Se ha corregido un problema por el que la tarea de migración se bloquea antes de que se inicie la fase de carga completa cuando se lee una tabla de Oracle con el cifrado de columnas de TDE habilitado.
3 de octubre de 2017	Se ha corregido un problema por el que una columna JSON de un origen PostgreSQL no migraba según lo previsto.
5 de octubre de 2017	Se ha corregido un problema por el que una tarea de migración de DMS muestra una latencia de origen igual a 0 cuando el registro REDO de archivo no se encuentra en la instancia de origen de Oracle. Esta solución aumenta de forma lineal la latencia de origen en estas condiciones.
20 de noviembre de 2017	Se ha corregido un problema con la migración de LOB por el que una columna TEXT de PostgreSQL se migraba a una columna CLOB de Oracle con espacios adicionales después de cada carácter de la entrada de LOB.
20 de noviembre de 2017	Se ha corregido un problema por el que una tarea de migración no se interrumpía según lo previsto después de la actualización de una instancia de replicación subyacente desde la versión 1.9.0 a la 2.4.0.

Fecha de notificación	Descripción
30 de noviembre de 2017	Se ha corregido un problema por el que una tarea de migración de DMS no captura correctamente los cambios realizados por un comando copy que se ejecuta en una instancia de PostgreSQL de origen.
11 de diciembre de 2017	Se ha corregido un problema por el que una tarea de migración generaba un error al leer datos de cambios de un registro de contenedor inexistente en un origen de MySQL.
11 de diciembre de 2017	Se ha corregido un problema por el que DMS lee datos de cambios de una tabla inexistente desde un origen de MySQL.
20 de diciembre de 2017	Incluye varias correcciones y mejoras de la característica de validación de datos.
22 de diciembre de 2017	Se ha corregido un problema con el parámetro <code>maxFileSize</code> para destinos de Amazon Redshift. Este parámetro se interpretaba erróneamente como bytes en lugar de kilobytes.
4 de enero de 2018	Se ha corregido un error de asignación de memoria para tareas de migración con Amazon DynamoDB como destino. En determinadas condiciones, AWS DMS no asignaba suficiente memoria si el mapeo de objetos utilizado contenía una clave de ordenación.
10 de enero de 2018	Se ha corregido un problema con Oracle 12.2 como instrucciones de lenguaje de manipulación de datos (DML) no se capturaban según lo previsto cuando se utilizaban <code>ROWDEPENDENCIES</code> .

## Notas de la versión 2.4.0 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 2.4.0 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
Replicación de espacios de tabla de índice de Oracle	Aggrega una funcionalidad compatible con la replicación de espacios de tabla de índice de Oracle. Para consultar más información acerca de los espacios de tabla de índice, haga clic <a href="#">aquí</a> .
Soporte para acceso Amazon S3 entre cuentas	Aggrega una funcionalidad para respaldar que las ACL predefinidas (concesiones predefinidas) sean compatibles con el acceso entre cuentas con puntos de enlace de S3. Para obtener más información acerca de las ACL predefinidas, consulte <a href="#">ACL predefinidas</a> en Guía para desarrolladores de Amazon Simple Storage Service.  Uso: establezca un atributo de conexión adicional en el punto de enlace de S3, que es <code>CannedAclForObjects=value</code> . Los valores posibles son los siguientes: <ul style="list-style-type: none"><li>• <code>NONE</code></li><li>• <code>PRIVATE</code></li><li>• <code>PUBLIC_READ</code></li></ul>

Nueva característica o mejora	Descripción
	<ul style="list-style-type: none"> <li>• PUBLIC_READ_WRITE</li> <li>• AUTHENTICATED_READ</li> <li>• AWS_EXEC_READ</li> <li>• BUCKET_OWNER_READ</li> <li>• BUCKET_OWNER_FULL_CONTROL</li> </ul>

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
19 de julio de 2017	Se ha solucionado un problema en el que la tarea de replicación se ejecutaba en un bucle de reintentos infinito cuando la instancia de PostgreSQL de origen se ejecutaba fuera de las ranuras de replicación. Con esta solución la tarea genera un error en el que se indica que DMS no puede crear una ranura de replicación lógica.
27 de julio de 2017	Se ha solucionado un problema en el motor de replicación en el que el tipo de datos de MySQL enum generaba un error de tarea con un error de asignación de memoria.
7 de agosto de 2017	Se ha solucionado un problema que producía un comportamiento inesperado en las tareas de migración que tenían Oracle como origen cuando Oracle se desactivaba durante más de cinco minutos. Este problema hacía que la fase de replicación en curso se colgara incluso después de que el origen volviera a estar disponible.
24 de agosto de 2017	Se ha solucionado un problema en el PostgreSQL de destino en el que el fragmento correspondiente al tipo de datos TIME se gestionaba de forma incorrecta.
14 de septiembre de 2017	Se ha solucionado un problema en el que se escribían valores incorrectos en los campos de TOAST, en destinos basados en PostgreSQL durante las actualizaciones de la fase de CDC.
8 de octubre de 2017	Se ha solucionado un problema de la versión 2.3.0 en el que la replicación en curso con orígenes de MySQL 5.5 no funcionaba según lo previsto.
12 de octubre de 2017	Se ha solucionado un problema en los cambios de lectura de un origen de SQL Server 2016 durante la fase de replicación en curso. Esta solución debe utilizarse con el atributo de conexión adicional siguiente en el punto de enlace de SQL Server: <code>IgnoreTxnCtxValidityCheck=true</code>

## Notas de la versión 2.3.0 de AWS Database Migration Service (AWS DMS)

En las siguientes tablas se muestran las características y las correcciones de errores de la versión 2.3.0 de AWS Database Migration Service (AWS DMS).

Nueva característica o mejora	Descripción
S3 como origen	<a href="#">Uso de Amazon S3 como origen para AWS DMS (p. 148)</a>
SQL Azure como origen	<a href="#">Uso de Microsoft Azure SQL Database como origen para AWS DMS (p. 119)</a>
Plataforma: actualizaciones de SDK de AWS	Actualización del SDK de AWS en la instancia de replicación para 1.0.113. El SDK de AWS se utiliza para determinados puntos de enlace (como Amazon Redshift y S3) para cargar datos en nombre del cliente en dichos puntos de enlace. Uso sin restricciones.
Origen de Oracle: compatibilidad con la replicación de espacios de tabla en Oracle	Capacidad para migrar espacios de tabla desde un origen de Oracle, lo que elimina la necesidad de crear previamente espacios de tabla en el destino antes de la migración.  Uso: utilice la opción <code>ReadTableSpaceName</code> de los atributos de conexión adicionales del punto de enlace de origen de Oracle y establezcala en true para que admita la replicación de espacios de tabla. De forma predeterminada, esta opción se establece en false.
Origen de Oracle: compatibilidad de la CDC con bases de datos standby Oracle Active Data Guard como origen	Capacidad para utilizar una instancia de standby para Oracle Active Data Guard como origen para replicar los cambios continuos a un destino soportado. Este cambio elimina la necesidad de establecer conexión con una base de datos activa que podría estar en la fase de producción.  Uso: utilice la opción <code>StandbyDelayTime</code> de los atributos de conexión adicionales del punto de enlace del origen de Oracle y especifique la hora en minutos para indicar el retraso en la sincronización de la base de datos standby.
Origen de PostgreSQL: permite añadir un latido WAL	Se ha añadido un registro de escritura previa (WAL) (es decir, la ejecución de consultas ficticias) para la replicación desde un origen de PostgreSQL. Esta característica se ha añadido de modo que las ranuras de replicación lógica inactivas no se mantienen en registros WAL antiguos, lo que puede dar lugar a situaciones de almacenamiento completo en el origen. Este latido mantiene en movimiento a <code>restart_lsn</code> y evita que el almacenamiento se llene.  El uso, cuando proceda, es como se indica a continuación: <ul style="list-style-type: none"> <li>• <code>HeartbeatEnable</code> se establece en true (el valor predeterminado es false).</li> <li>• <code>HeartbeatSchema</code> es el esquema para artefactos de latido (el valor predeterminado es public).</li> <li>• <code>HeartbeatFrequency</code> es la frecuencia de latido en minutos (el valor predeterminado es 5 y el valor mínimo es 1)</li> </ul>
Todos los puntos de enlace: se mantiene una replicación homogénea con la transformación	La capacidad para operar como las migraciones de las tareas de migración homogéneas (desde la perspectiva de una estructura de tabla/tipo de datos) se introdujo con la versión 2.2.0. Sin embargo, DMS seguía convirtiendo los tipos de datos internamente cuando se lanzaba una tarea con transformaciones de la tabla. Esta característica mantiene los tipos de datos del origen en el destino para migraciones de tipo lift and shift, incluso cuando se usan transformaciones.  Uso sin restricciones para todas las migraciones homogéneas.

Nueva característica o mejora	Descripción
Todos los puntos de enlace: error de tarea cuando no se encuentran tablas	Capacidad para forzar un error de tarea de replicación cuando las reglas de inclusión de transformación no encuentran coincidencias.  Uso: cambie la opción de tarea <code>FailOnNoTablesCaptured</code> en true.
Oracle como origen: se detiene la tarea cuando falta el registro REDO de archivo.	Capacidad para salir de un bucle de reintentos y detener una tarea cuando falta el registro REDO de archivo en el origen.  Uso: utilice el atributo de conexión adicional <code>RetryTimeoutInMinutes</code> para especificar en minutos el tiempo de espera necesario para que se produzca la detención.

Los problemas resueltos son los siguientes.

Fecha de notificación	Descripción
5 de enero de 2017	Se ha solucionado un problema de colisión de ID del servidor cuando se lanzan varias tareas de DMS en la misma instancia de MySQL (versión 5.6 o versiones posteriores)
21 de febrero de 2017	Se ha solucionado un problema en el que la creación de una tabla genera un error en <code>nestingLevel=ONE</code> cuando <code>_id</code> en MongoDB es una cadena en el documento. Antes de esta solución <code>_id</code> (cuando es una cadena) se creaba como LONGTEXT (MySQL) o CLOB (Oracle), lo que generaba un error cuando se intentaba convertirlo en una clave principal.
5 de mayo de 2017	Se ha solucionado un problema en el que los LOB NULL se migraban vacíos cuando se usaba el modo LOB completo con un origen de Oracle.
5 de mayo de 2017	Se ha solucionado un error en el que una tarea con MySQL como origen genera un error de demasiadas conexiones cuando ya han pasado más de 24 horas desde la hora de inicio de la CDC personalizada.
24 de mayo de 2017	Se ha solucionado un problema en el que la tarea estaba en estado de inicio demasiado tiempo iniciándose cuando se lanzaban varias tareas en la instancia de replicación a la vez.
7 de julio de 2017	Se ha solucionado un problema que generó un mensaje de error de PostgreSQL sobre la utilización de todas las ranuras de conexión disponibles. Ahora se registra un error en el nivel de registro predeterminado cuando todas las ranuras de conexión disponibles con PostgreSQL se han usado y DMS no puede obtener más ranuras para continuar con el proceso de replicación.
19 de julio de 2017	Se ha solucionado un problema en el que las actualizaciones y eliminaciones desde Oracle a DynamoDB no se migraban correctamente.
8 de agosto de 2017	Se ha solucionado un problema que producía un comportamiento inesperado durante la CDC, cuando una instancia de base de datos de origen de Oracle se desactivaba durante más de cinco minutos durante una migración.

Fecha de notificación	Descripción
12 de agosto de 2017	Se ha solucionado un error en el que los valores nulos de cualquier origen se migraban como <code>amazon_null</code> , lo que provocaba problemas cuando se insertaban en tipos de datos otros distintos a varchar en Amazon Redshift.
27 de agosto de 2017	Para MongoDB, se ha solucionado un problema en el que la tarea de carga completa se bloquea cuando <code>nestingLevel=NONE</code> e <code>_id</code> no es <code>ObjectId</code> .

# Historial de revisión

En la siguiente tabla se describen los cambios importantes en la guía del usuario de AWS Database Migration Service después de enero de 2018.

Puede suscribirse a una fuente RSS para recibir notificaciones de las actualizaciones de esta documentación.

update-history-change	update-history-description	update-history-date
<a href="#">Migración con AWS Snowball Edge</a>	Se ha actualizado la documentación para utilizar AWS Snowball Edge para migrar bases de datos grandes.	January 24, 2019
<a href="#">Compatibilidad con Amazon Elasticsearch Service y Amazon Kinesis Data Streams como destinos</a>	Se ha añadido compatibilidad con Amazon ES y Kinesis Data Streams como destinos para la migración de datos.	November 15, 2018
<a href="#">Compatibilidad con el inicio nativo de CDC</a>	Se ha añadido compatibilidad con puntos de inicio nativos cuando se utiliza la captura de datos de cambios (CDC).	June 28, 2018
<a href="#">Compatibilidad con R4</a>	Se ha agregado compatibilidad con las clases de instancia de replicación R4.	May 10, 2018
<a href="#">Compatibilidad con Db2 LUW</a>	Se ha añadido compatibilidad con IBM Db2 LUW como origen para la migración de datos.	April 26, 2018
<a href="#">Compatibilidad con el log de tareas</a>	Se ha añadido compatibilidad para ver el uso de los logs de tareas y purgarlos.	February 8, 2018
<a href="#">Compatibilidad con SQL Server como destino</a>	Se ha añadido compatibilidad con Amazon RDS for Microsoft SQL Server como origen.	February 6, 2018

## Actualizaciones anteriores

En la siguiente tabla se describen los cambios importantes en la guía del usuario de AWS Database Migration Service antes de enero de 2018.

Cambio	Descripción	Fecha de modificación
Nueva característica	Se ha agregado compatibilidad para utilizar AWS DMS con AWS Snowball para migrar bases de datos de gran tamaño. Para obtener más información, consulte <a href="#">Migración de grandes almacenes de datos con AWS Database Migration Service y AWS Snowball Edge (p. 348)</a> .	17 de noviembre de 2017

Cambio	Descripción	Fecha de modificación
Nueva característica	Se ha agregado compatibilidad para el informe de evaluación de tareas y la validación de datos. Para obtener más información sobre el informe de evaluación de tareas, consulte <a href="#">Creación de un informe de evaluación de tarea (p. 252)</a> . Para obtener más información sobre la validación de datos, consulte <a href="#">Configuración de tareas de validación de datos (p. 270)</a> .	17 de noviembre de 2017
Nueva característica	Se ha agregado compatibilidad con plantillas de AWS CloudFormation. Para obtener más información, consulte <a href="#">Compatibilidad de AWS DMS con AWS CloudFormation (p. 13)</a> .	11 de julio de 2017
Nueva característica	Se ha añadido compatibilidad para utilizar Amazon Dynamo como destino. Para obtener más información, consulte <a href="#">Uso de una base de datos de Amazon DynamoDB como destino para AWS Database Migration Service (p. 210)</a> .	10 de abril de 2017
Nueva característica	Se ha añadido compatibilidad para utilizar MongoDB como destino. Para obtener más información, consulte <a href="#">Uso de MongoDB como origen para AWS DMS (p. 143)</a> .	10 de abril de 2017
Nueva característica	Se ha añadido compatibilidad para utilizar Amazon S3 como destino. Para obtener más información, consulte <a href="#">Uso de Amazon S3 como destino para AWS Database Migration Service (p. 189)</a> .	27 de marzo de 2017
Nueva característica	Se ha añadido compatibilidad para recargar las tablas de la base de datos durante una tarea de migración. Para obtener más información, consulte <a href="#">Volver a cargar tablas durante una tarea (p. 280)</a> .	7 de marzo de 2017
Nueva característica	Se ha añadido compatibilidad con los eventos y las suscripciones a eventos. Para obtener más información, consulte <a href="#">Uso de eventos y notificaciones en AWS Database Migration Service (p. 343)</a> .	26 de enero de 2017
Nueva característica	Se ha añadido compatibilidad con los puntos de enlace SSL para Oracle. Para obtener más información, consulte <a href="#">Compatibilidad con SSL para un punto de enlace de Oracle (p. 55)</a> .	5 de diciembre de 2016
Nueva característica	Se ha añadido compatibilidad para utilizar la captura de datos de cambios (CDC) con una instancia de base de datos de PostgreSQL en Amazon RDS. Para obtener más información, consulte <a href="#">Configuración de una instancia de base de datos de Amazon RDS para PostgreSQL como un origen (p. 124)</a> .	14 de septiembre de 2016
Compatibilidad con nuevas regiones	Se ha añadido compatibilidad con las regiones de Asia Pacífico (Mumbai), Asia Pacífico (Seúl) y América del Sur (São Paulo). Para obtener una lista de las regiones compatibles, consulte <a href="#">¿Qué es AWS Database Migration Service? (p. 1)</a> .	3 de agosto de 2016

Cambio	Descripción	Fecha de modificación
Nueva característica	Se ha añadido compatibilidad con la replicación continua. Para obtener más información, consulte <a href="#">Replicación continua (p. 390)</a> .	13 de julio de 2016
Nueva característica	Se ha añadido compatibilidad con las conexiones protegidas con SSL. Para obtener más información, consulte <a href="#">Uso de SSL con AWS Database Migration Service (p. 51)</a> .	13 de julio de 2016
Nueva característica	Se ha añadido compatibilidad con SAP Adaptive Server Enterprise (ASE) como punto de enlace de origen o de destino. Para obtener más información, consulte <a href="#">Uso de una base de datos SAP ASE como origen para AWS DMS (p. 140)</a> y <a href="#">Uso de una base de datos de SAP ASE como destino para AWS Database Migration Service (p. 187)</a> .	13 de julio de 2016
Nueva característica	Se ha añadido compatibilidad con filtros para mover un subconjunto de filas de la base de datos de origen a la base de datos de destino. Para obtener más información, consulte <a href="#">Usar filtros de origen (p. 317)</a> .	2 de mayo de 2016
Nueva característica	Se ha añadido compatibilidad con Amazon Redshift como un punto de enlace de destino. Para obtener más información, consulte <a href="#">Uso de una base de datos de Amazon Redshift como destino para AWS Database Migration Service (p. 175)</a> .	2 de mayo de 2016
Disponibilidad general	Versión inicial de AWS Database Migration Service.	14 de marzo de 2016
Versión de vista previa pública	Lanzamiento de la documentación preliminar de AWS Database Migration Service.	21 de enero de 2016

# AWS Glossary

For the latest AWS terminology, see the [AWS Glossary](#) in the AWS General Reference.