



Implementación de PostgreSQL gestionado para aplicaciones SaaS multiusuario en AWS

AWS Guía prescriptiva



AWS Guía prescriptiva: Implementación de PostgreSQL gestionado para aplicaciones SaaS multiusuario en AWS

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

Introducción	1
Resultados empresariales específicos	1
Selección de una base de datos para una aplicación SaaS	3
Elegir entre Amazon RDS y Aurora	6
Modelos de particionamiento SaaS multiusuario para PostgreSQL	8
modelo de silo de PostgreSQL	9
modelo de grupo de PostgreSQL	10
modelo de puente de PostgreSQL	12
Matriz de decisiones	14
Recomendaciones de seguridad de nivel de fila	30
Disponibilidad de PostgreSQL para el modelo de pool	32
Prácticas recomendadas	34
CompareAWS las opciones de PostgreSQL administrado	34
Seleccione un modelo de particionamiento SaaS para varios usuarios	34
Utilice la seguridad de nivel de fila para los modelos de particionamiento SaaS de grupos	34
Preguntas frecuentes	36
¿Qué opciones administradas de PostgreSQLAWS ofrece?	36
¿Qué servicio es óptimo para las aplicaciones SaaS?	36
¿Qué requisitos únicos debo tener en cuenta si decido utilizar una base de datos PostgreSQL con una aplicación SaaS multiusuario?	36
¿Qué modelos puedo usar para mantener el aislamiento de los datos de los inquilinos con PostgreSQL?	36
¿Cómo mantengo el aislamiento de los datos de los inquilinos con una única base de datos PostgreSQL que se comparte entre varios inquilinos?	37
Pasos siguientes	38
Recursos	39
Referencias	39
Socios	39
Historial de documentos	40
Glosario	41
#	41
A	42
B	45
C	47

D	50
E	55
F	57
G	58
H	59
I	60
L	63
M	64
O	68
P	71
Q	74
R	74
S	77
T	81
U	82
V	83
W	83
Z	85
.....	lxxxvi

Implementación de PostgreSQL gestionado para aplicaciones SaaS multiusuario en AWS

Tabby Ward y Thomas Davis, Amazon Web Services (AWS)

Abril de 2024 ([historial del documento](#))

Al seleccionar una base de datos para almacenar los datos operativos, es fundamental tener en cuenta cómo deben estructurarse los datos, a qué consultas responderán, con qué rapidez proporcionarán respuestas y la resiliencia de la propia plataforma de datos. Además de estas consideraciones generales, están las implicaciones del software como servicio (SaaS) para los datos operativos, como el aislamiento del rendimiento, la seguridad de los inquilinos y las características y patrones de diseño únicos que son típicos de los datos para aplicaciones SaaS de múltiples inquilinos. Esta guía explica cómo se aplican estos factores al uso de una base de datos PostgreSQL en Amazon Web Services AWS() como almacén de datos operativos principal para una aplicación SaaS multiusuario. En concreto, la guía se centra en dos opciones de PostgreSQL AWS gestionado: Amazon Aurora PostgreSQL Compatible Edition y Amazon Relational Database Service (Amazon RDS) para PostgreSQL.

Resultados empresariales específicos

Esta guía proporciona un análisis detallado de las mejores prácticas para aplicaciones SaaS multiusuario que utilizan Aurora compatible con PostgreSQL y Amazon RDS for PostgreSQL. Le recomendamos que utilice los patrones y conceptos de diseño que se proporcionan en esta guía para fundamentar y estandarizar la implementación de Amazon RDS for PostgreSQL o Aurora RDS for PostgreSQL para sus aplicaciones SaaS multiusuario.

Esta guía prescriptiva ayuda a lograr los siguientes resultados empresariales:

- Elegir la opción de PostgreSQL AWS gestionado más óptima para su caso de uso: en esta guía se comparan las opciones relacionales y no relacionales para el uso de bases de datos con aplicaciones SaaS. También analiza qué casos de uso son los más óptimos para Aurora, compatible con PostgreSQL, y Amazon RDS para PostgreSQL. Esta información le ayudará a seleccionar la mejor opción para su aplicación de SaaS.
- Aplicación de las mejores prácticas de SaaS mediante la adopción de un modelo de particionamiento de SaaS: en esta guía se analizan y comparan tres modelos generales de

particionamiento de SaaS que son aplicables a un sistema de administración de bases de datos (DBMS) PostgreSQL: modelos agrupados, puenteados y silos, y sus variaciones. Estos enfoques reflejan las mejores prácticas de SaaS y proporcionan flexibilidad a la hora de diseñar una aplicación SaaS. La aplicación de un modelo de particionamiento de SaaS es una parte crucial para preservar las mejores prácticas.

- Uso efectivo del RLS en los modelos de particionamiento SaaS de grupos: la seguridad a nivel de fila (RLS) permite aplicar el aislamiento de los datos de los inquilinos en una sola tabla de PostgreSQL al restringir las filas que se pueden ver en función del usuario o de una variable de contexto. Cuando se utiliza el modelo de particionamiento de grupos, se requiere el RLS para evitar el acceso entre inquilinos.

Selección de una base de datos para una aplicación SaaS

Para muchas aplicaciones SaaS multiusuario, la selección de una base de datos operativa se puede resumir en una elección entre bases de datos relacionales y no relacionales, o una combinación de ambas. Para tomar una decisión, tenga en cuenta estas características y requisitos de datos de aplicaciones de alto nivel:

- Modelo de datos de la aplicación
- Patrones de acceso a los datos
- Requisitos de latencia de base de datos
- Requisitos de integridad de los datos e integridad transaccional (atomicidad, consistencia, aislamiento y durabilidad, o ACID)
- Requisitos de disponibilidad y recuperación entre regiones

En la siguiente tabla se enumeran los requisitos y las características de los datos de las aplicaciones y se analizan en el contexto de las ofertas de AWS bases de datos: Aurora compatible con PostgreSQL y Amazon RDS for PostgreSQL (relacional) y Amazon DynamoDB (no relacional). Puede hacer referencia a esta matriz cuando intente decidir entre ofertas de bases de datos operativas relacionales y no relacionales.

Bases de datos	Requisitos y características de los datos de las aplicaciones SaaS				
	Modelo de datos	Patrones de acceso	Requisitos de latencia	Integridad transaccional y de datos	Disponibilidad y recuperación entre regiones
Relacional (Compatible con Aurora PostgreSQL y Amazon RDS para PostgreSQL)	Relacional o altamente normalizado.	No tiene que planificarse minuciosamente de antemano.	Preferible, una mayor tolerancia a la latencia; puede lograr latencias más bajas	De forma predeterminada, se mantiene una alta integridad transaccional y de datos.	En Amazon RDS, puede crear una réplica de lectura para el escalado entre regiones y

de forma
predeterm
inada con
Aurora e
implement
ando réplicas
de lectura,
almacenam
iento en
caché y
funciones
similares.

la conmutaci
ón por error.
[Aurora](#)
[automatiz](#)
[a principal](#)
[mente este](#)
[proceso](#). Para
configura
ciones activo-
activas
en varias
Regiones
de AWS,
puede utilizar
el [reenvío](#)
[de escritura](#)
junto con
las bases
de datos
globales de
[Aurora](#).

No relacional (Amazon DynamoDB)	Suele estar desnormalizado. Estas bases de datos aprovechan los patrones para modelar many-to-many relaciones , elementos grandes y datos de series temporales .	Todos los patrones de acceso (consultas) a los datos deben comprenderse a fondo antes de crear un modelo de datos.	Latencia muy baja con opciones como Amazon DynamoDB Accelerator (DAX) que permiten mejorar aún más el rendimiento.	Integridad transaccional opcional a costa del rendimiento. Los problemas de integridad de los datos se trasladan a la aplicación.	Fácil recuperación entre regiones y configuración activo-activa con tablas globales. (El cumplimiento de ACID solo se puede lograr en una sola región). AWS
---	--	--	--	---	--

Algunas aplicaciones SaaS multiusuario pueden tener modelos de datos únicos o circunstancias especiales que se gestionan mejor con bases de datos no incluidas en la tabla anterior. Por ejemplo, los conjuntos de datos de series temporales, los conjuntos de datos altamente conectados o el mantenimiento de un registro de transacciones centralizado pueden requerir el uso de un tipo de base de datos diferente. El análisis de todas las posibilidades va más allá del alcance de esta guía. Para obtener una lista completa de las ofertas de AWS bases de datos y cómo pueden satisfacer diferentes casos de uso a un alto nivel, consulte la sección de [bases](#) de datos del documento técnico Overview of Amazon Web Services.

El resto de esta guía se centra en los servicios de bases de datos AWS relacionales compatibles con PostgreSQL: Amazon RDS y Aurora PostgreSQL. DynamoDB requiere un enfoque diferente para optimizar las aplicaciones de SaaS, lo que va más allá del alcance de esta guía. Para obtener más información sobre DynamoDB, consulte la entrada AWS del [blog Cómo particionar datos SaaS agrupados de múltiples inquilinos con Amazon DynamoDB](#).

Elegir entre Amazon RDS y Aurora

En la mayoría de los casos, recomendamos utilizar Aurora, compatible con PostgreSQL, en lugar de Amazon RDS for PostgreSQL. En la siguiente tabla se muestran los factores que debe tener en cuenta al decidir entre estas dos opciones.

Componente DBMS	Amazon RDS para PostgreSQL	Compatible con Aurora PostgreSQL
Escalabilidad	Retraso de replicación de minutos, máximo de 5 réplicas de lectura	Retraso de replicación inferior a un minuto (normalmente menos de 1 segundo con bases de datos globales), máximo de 15 réplicas de lectura
Recuperación tras un bloqueo	Los puntos de control separados por 5 minutos (de forma predeterminada) pueden ralentizar el rendimiento de la base de datos	Recuperación asíncrona con subprocesos paralelos para una recuperación rápida
Conmutación por error	60 a 120 segundos, además del tiempo de recuperación en caso de accidente	Por lo general, unos 30 segundos (incluida la recuperación en caso de accidente)
Almacenamiento	Un máximo de 256 000 IOPS	Las IOPS están limitadas únicamente por el tamaño y la capacidad de la instancia Aurora
Alta disponibilidad y recuperación ante desastres	Dos zonas de disponibilidad con una instancia en espera y conmutación por error entre regiones para leer copias de seguridad copiadas o réplicas	Tres zonas de disponibilidad de forma predeterminada, conmutación por error entre regiones con bases de datos globales de Aurora

Componente DBMS	Amazon RDS para PostgreSQL	Compatible con Aurora PostgreSQL
	L	y reenvío de escritura para configuraciones activo-activas Regiones de AWS
Copia de seguridad	Durante el período de backup, puede afectar al rendimiento	Respaldos incrementales automáticos, sin impacto en el rendimiento
Clases de instancias de bases de datos	Consulte la lista de clases de instancias de Amazon RDS	Consulte la lista de clases de instancias de Aurora

En todas las categorías descritas en la tabla anterior, la compatibilidad con Aurora PostgreSQL suele ser la mejor opción. Sin embargo, Amazon RDS for PostgreSQL podría seguir teniendo sentido para cargas de trabajo pequeñas y medianas, ya que cuenta con una mayor selección de clases de instancias que podrían ofrecer una opción más rentable a expensas del conjunto de funciones más sólido de Aurora.

Modelos de particionamiento SaaS multiusuario para PostgreSQL

El mejor método para lograr la tenencia múltiple depende de los requisitos de su aplicación SaaS. En las siguientes secciones se muestran los modelos de particionamiento para implementar correctamente la multitenencia en PostgreSQL.

Note

Los modelos que se analizan en esta sección son aplicables tanto a Amazon RDS for PostgreSQL como a los compatibles con Aurora PostgreSQL. Las referencias a PostgreSQL en esta sección se aplican a ambos servicios.

Hay tres modelos de alto nivel que puede usar en PostgreSQL para la partición de SaaS: silo, puente y pool. En la siguiente imagen se resumen las ventajas y desventajas entre los modelos de silo y piscina. El modelo de puente es un híbrido de los modelos de silo y piscina.

Modelo de particionamiento	Ventajas	Desventajas
Silo	<ul style="list-style-type: none"> • alineación de la conformidad • Sin impacto entre inquilinos • Ajuste a nivel de inquilino • Disponibilidad a nivel de inquilino 	<ul style="list-style-type: none"> • Agilidad comprometida • Sin administración centralizada • Complejidad de implementación • Costo
piscina	<ul style="list-style-type: none"> • Agilidad • Optimización de costos • Administración centralizada • Despliegue simplificado 	<ul style="list-style-type: none"> • Impacto entre inquilinos • Desafíos de conformidad • Disponibilidad de todo o nada
puente	<ul style="list-style-type: none"> • Alguna alineación de la conformidad • Agilidad 	<ul style="list-style-type: none"> • Algunos desafíos de conformidad

Modelo de particionamiento	Ventajas	Desventajas
	<ul style="list-style-type: none">• Optimización de costos• Administración centralizada	<ul style="list-style-type: none">• Disponibilidad de todo o nada (en su mayoría)• Impacto entre inquilinos• Complejidad de implementación

En las siguientes secciones, se describen cada modelo de manera más detallada.

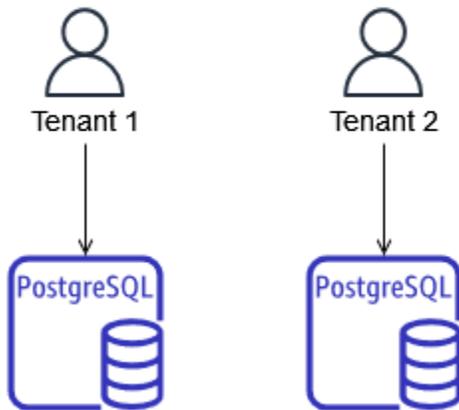
Modelos de particionamiento:

- [modelo de silo de PostgreSQL](#)
- [modelo de grupo de PostgreSQL](#)
- [modelo de puente de PostgreSQL](#)
- [Matriz de decisiones](#)

modelo de silo de PostgreSQL

El modelo de silo se implementa mediante el aprovisionamiento de una instancia de PostgreSQL para cada inquilino de una aplicación. El modelo de silos destaca por el desempeño de los inquilinos y el aislamiento de seguridad, y elimina por completo el fenómeno de los vecinos ruidosos. El fenómeno del vecino ruidoso se produce cuando el uso de un sistema por parte de un inquilino afecta al desempeño de otro inquilino. El modelo de silo permite adaptar el rendimiento específicamente a cada inquilino y, potencialmente, limitar las interrupciones al silo de un inquilino específico. Sin embargo, lo que generalmente impulsa la adopción de un modelo de silo son las estrictas restricciones regulatorias y de seguridad. Estas restricciones pueden estar motivadas por los clientes de SaaS. Por ejemplo, los clientes de SaaS pueden exigir que se aislen sus datos debido a restricciones internas, y los proveedores de SaaS pueden ofrecer dicho servicio por una tarifa adicional.

Silo model (separate PostgreSQL instances or clusters for each tenant)

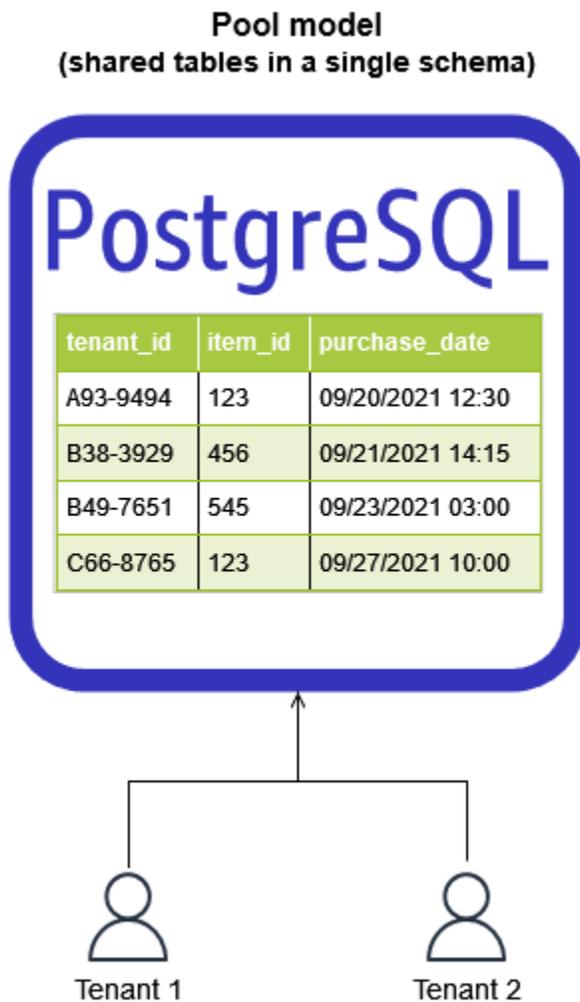


Si bien el modelo de silo puede ser necesario en ciertos casos, tiene muchos inconvenientes. A menudo es difícil utilizar el modelo de silo de manera rentable, ya que gestionar el consumo de recursos en varias instancias de PostgreSQL puede resultar complicado. Además, la naturaleza distribuida de las cargas de trabajo de las bases de datos en este modelo hace que sea más difícil mantener una visión centralizada de la actividad de los inquilinos. La gestión de tantas cargas de trabajo operadas de forma independiente aumenta la sobrecarga operativa y administrativa. El modelo de silos también hace que la incorporación de inquilinos sea más complicada y lleve más tiempo, ya que hay que aprovisionar recursos específicos para cada inquilino. Además, puede resultar más difícil escalar todo el sistema SaaS, ya que el número cada vez mayor de instancias de PostgreSQL específicas para cada inquilino exigirá más tiempo operativo de administración. Una última consideración es que una aplicación o una capa de acceso a datos deberán mantener un mapeo de los inquilinos con sus instancias de PostgreSQL asociadas, lo que aumenta la complejidad de la implementación de este modelo.

modelo de grupo de PostgreSQL

El modelo de grupo se implementa mediante el aprovisionamiento de una única instancia de PostgreSQL (Amazon RDS o Aurora) y el uso de [seguridad a nivel de fila \(RLS\)](#) para mantener el aislamiento de los datos de los inquilinos. Las políticas de RLS restringen qué filas de una tabla devuelven `SELECT` las consultas o qué filas se ven afectadas por `INSERT`, `UPDATE`, y `DELETE` los comandos. El modelo de grupo centraliza todos los datos de los inquilinos en un único esquema de PostgreSQL, por lo que es significativamente más rentable y requiere menos gastos operativos para su mantenimiento. La supervisión de esta solución también es significativamente más sencilla

debido a su centralización. Sin embargo, el monitoreo de los impactos específicos de los inquilinos en el modelo de piscina generalmente requiere algunos instrumentos adicionales en la aplicación. Esto se debe a que PostgreSQL, de forma predeterminada, no sabe qué inquilino consume recursos. La incorporación de inquilinos se simplifica porque no se requiere una nueva infraestructura. Esta agilidad facilita la realización de flujos de trabajo de incorporación de inquilinos rápidos y automatizados.



Si bien el modelo de piscina es generalmente más rentable y más sencillo de administrar, tiene algunas desventajas. El fenómeno del vecino ruidoso no se puede eliminar por completo en un modelo de piscina. Sin embargo, se puede mitigar asegurándose de que haya los recursos adecuados disponibles en la instancia de PostgreSQL y utilizando estrategias para reducir la carga en PostgreSQL, como transferir consultas para leer réplicas o a Amazon ElastiCache. El monitoreo efectivo también desempeña un papel a la hora de responder a los problemas de aislamiento

del desempeño de los inquilinos, ya que la instrumentación de la aplicación puede registrar y monitorear la actividad específica de los inquilinos. Por último, es posible que algunos clientes de SaaS no consideren suficiente la separación lógica proporcionada por el RLS y soliciten medidas de aislamiento adicionales.

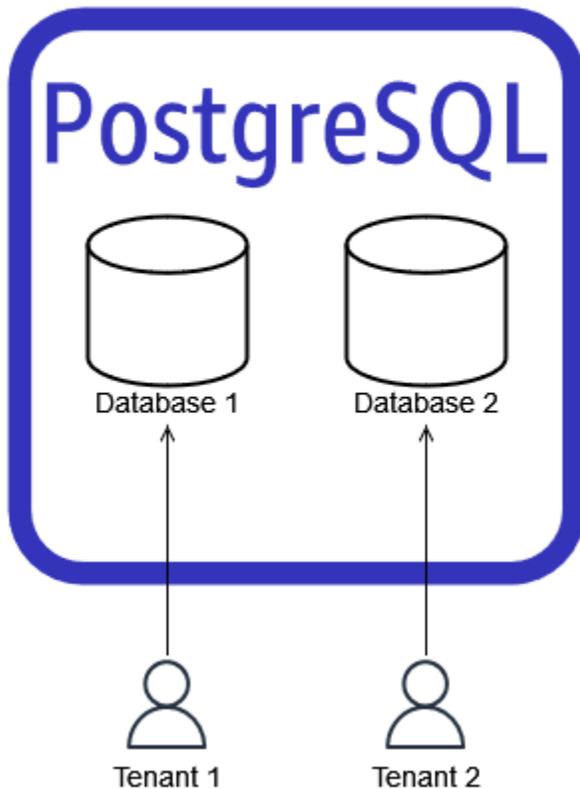
modelo de puente de PostgreSQL

El modelo de puente de PostgreSQL es una combinación de enfoques agrupados y en silos. Al igual que en el modelo agrupado, aprovisionas una única instancia de PostgreSQL para cada inquilino. Para mantener el aislamiento de los datos de los inquilinos, utilice construcciones lógicas de PostgreSQL. En el siguiente diagrama, las bases de datos PostgreSQL se utilizan para separar los datos de forma lógica.

Note

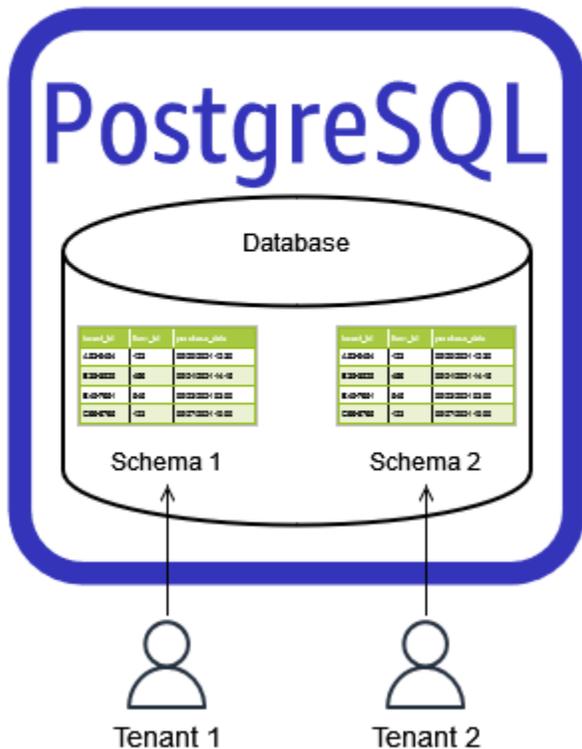
Una base de datos PostgreSQL no hace referencia a una instancia de base de datos independiente compatible con Amazon RDS for PostgreSQL o Aurora PostgreSQL. En cambio, se refiere a una construcción lógica del sistema de administración de bases de datos PostgreSQL para separar los datos.

Bridge model with separate databases (separate databases in a single instance)



También puede implementar el modelo puente mediante una única base de datos PostgreSQL, con esquemas específicos para cada inquilino en cada base de datos, como se ilustra en el siguiente diagrama.

Bridge model with separate schemas (separate schemas in a single database)



El modelo de puente tiene los mismos problemas ruidosos de aislamiento del desempeño de vecinos e inquilinos que el modelo de piscina. También conlleva algunos gastos operativos y de aprovisionamiento adicionales al requerir que se aprovisionen bases de datos o esquemas separados por inquilino. Requiere un monitoreo efectivo para responder rápidamente a las preocupaciones de desempeño de los inquilinos. También requiere la instrumentación de la aplicación para monitorear el uso específico del inquilino. En general, el modelo puente puede considerarse una alternativa al RLS, que aumenta ligeramente el esfuerzo de incorporación de los inquilinos al requerir nuevas bases de datos o esquemas de PostgreSQL. Al igual que con el modelo de silos, una aplicación o una capa de acceso a datos deberán mantener un mapeo de los inquilinos con sus bases de datos o esquemas de PostgreSQL asociados.

Matriz de decisiones

Para decidir qué modelo de particionamiento SaaS multiusuario debe utilizar con PostgreSQL, consulte la siguiente matriz de decisiones. La matriz analiza estas cuatro opciones de particionamiento:

- Silo: una instancia o clúster de PostgreSQL independiente para cada inquilino.
- Puente con bases de datos independientes: una base de datos independiente para cada inquilino en una única instancia o clúster de PostgreSQL.
- Puente con esquemas separados: un esquema independiente para cada inquilino en una única base de datos de PostgreSQL, en una única instancia o clúster de PostgreSQL.
- Piscina: mesas compartidas para inquilinos en una sola instancia y esquema.

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Caso de uso	El aislamiento de los datos con un control total del uso de los recursos es un requisito clave, o tendrá inquilinos muy grandes y muy sensibles al rendimiento.	El aislamiento de los datos es un requisito clave y se requiere una referencia cruzada limitada o nula de los datos de los inquilinos.	Número moderado de inquilinos con una cantidad moderada de datos. Este es el modelo preferido si tiene que cruzar los datos de los inquilinos.	Gran cantidad de inquilinos con menos datos por inquilino.
Agilidad de incorporación de nuevos inquilinos	Muy lentos. (Se requiere una nueva instancia o clúster para cada inquilino).	Moderadamente lentos. (Requiere crear una nueva base de datos para que cada inquilino almacene los objetos del esquema).	Moderadamente lentos. (Requiere crear un nuevo esquema para que cada inquilino almacene los objetos).	La opción más rápida. (Se requiere una configuración mínima).
Esfuerzo y eficiencia de configuración	Se requiere un esfuerzo significativo.	Se requiere un esfuerzo significativo.	Requiere menos esfuerzo. (Una configuración	Se requiere el mínimo esfuerzo.

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
del grupo de conexiones a bases de datos	<p>(Un grupo de conexiones por inquilino).</p> <p>Menos eficiente . (No se comparten conexiones de bases de datos entre los inquilinos).</p>	<p>(Una configuración de grupo de conexiones por inquilino , a menos que utilice Amazon RDS Proxy).</p> <p>Menos eficiente . (No se comparten conexiones de bases de datos entre los inquilinos y el número total de conexiones. El uso para todos los inquilinos está limitado en función de la clase de instancia de base de datos.)</p>	<p>de grupo de conexiones para todos los inquilinos).</p> <p>Moderadamente eficiente . (Reutilización de la conexión mediante el SET SCHEMA comando SET ROLE o únicamente en el modo de grupo de sesiones. SET los comandos también bloquean la sesión cuando se utiliza Amazon RDS Proxy, pero se pueden eliminar los grupos de conexiones de los clientes y se pueden establecer conexiones directas para cada solicitud</p>	<p>Lo más eficiente . (Un grupo de conexiones para todos los inquilinos y una reutilización eficiente de las conexiones para todos los inquilinos. Los límites de conexión a la base de datos se basan en la clase de instancia de base de datos.)</p>

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
			para mayor eficiencia).	
Mantenimiento de bases de datos (gestión de vacíos) y uso de recursos	Administración más sencilla.	Complejidad media. (Puede provocar un alto consumo de recursos, ya que después hay que iniciar un aspirador para cada base de datos <code>vacuum_naptime</code> , lo que lleva a un uso elevado de la CPU del lanzador de vacío automático. También puede haber una sobrecarga adicional asociada a la aspiración de las tablas del catálogo del sistema PostgreSQL para cada base de datos).	Tablas de catálogo de sistemas PostgreSQL de gran tamaño. (<code>pg_catalog</code> Tamaño total en decenas de GB, según el número de inquilinos y los parientes. Es probable que se requieran modificaciones en los parámetros relacionados con la aspiración para controlar la hinchazón de la mesa).	Las tablas pueden ser grandes, según el número de inquilinos y los datos de cada inquilino. (Es probable que se requieran modificaciones en los parámetros relacionados con la aspiración para controlar la hinchazón de la mesa).

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Esfuerzo de gestión de extensiones	Esfuerzo significativo (para cada base de datos en instancias separadas).	Esfuerzo significativo (en cada nivel de base de datos).	Esfuerzo mínimo (una vez en la base de datos común).	Esfuerzo mínimo (una vez en la base de datos común).
Cambiar el esfuerzo de despliegue	Esfuerzo significativo. (Connect a cada instancia por separado e implemente los cambios).	Esfuerzo significativo. (Connect a cada base de datos y esquema e implemente los cambios).	Esfuerzo moderado. (Connect a una base de datos común e implemente los cambios para cada esquema).	Esfuerzo mínimo. (Connect a una base de datos común e implemente los cambios).
Despliegue de cambios: alcance del impacto	Mínimo. (Inquilin o único afectado)	Mínimo. (Inquilin o único afectado)	Mínimo. (Inquilin o único afectado)	Muy grande. (Todos los inquilinos afectados).

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Gestión y esfuerzo del rendimiento de las consultas	Rendimiento de las consultas administrable.	Rendimiento de las consultas administrable.	Rendimiento de las consultas administrable.	Es probable que se requiera un esfuerzo significativo para mantener el rendimiento de las consultas. (Con el tiempo, es posible que las consultas se ejecuten más lentamente debido al aumento del tamaño de las tablas. Puede utilizar las particiones de tablas y la fragmentación de bases de datos para mantener el rendimiento.)

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Impacto en los recursos entre inquilinos	Sin impacto. (No se comparten recursos entre los inquilinos).	Impacto moderado. (Los inquilinos comparten recursos comunes, como la CPU y la memoria de la instancia).	Impacto moderado. (Los inquilinos comparten recursos comunes, como la CPU y la memoria de la instancia).	Impacto fuerte. (Los inquilinos se afectan unos a otros en términos de recursos, conflictos de bloqueo, etc.)
Ajuste a nivel de inquilino (por ejemplo, creación de índices adicionales por inquilino o ajuste de parámetros de base de datos para un inquilino en particular)	posibles.	Algo posible. (Se pueden realizar cambios a nivel de esquema para cada inquilino, pero los parámetros de la base de datos son globales para todos los inquilinos).	Algo posible. (Se pueden realizar cambios a nivel de esquema para cada inquilino, pero los parámetros de la base de datos son globales para todos los inquilinos).	No es posible. (Todos los inquilinos comparten las mesas).

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Reequilibrar los esfuerzos para los inquilinos sensibles al desempeño	Mínimo. (No es necesario reequilibrar. Amplíe los recursos de servidor y E/S para gestionar este escenario).	Moderado. (Utilice la replicación lógica <code>opg_dump</code> para exportar la base de datos, pero el tiempo de inactividad puede ser prolongado según el tamaño de los datos. Puede usar la función de base de datos transportable de Amazon RDS for PostgreSQL para copiar bases de datos entre instancias con mayor rapidez).	Moderado, pero probablemente implique un tiempo de inactividad prolongado. (Utilice la replicación lógica <code>opg_dump</code> para exportar el esquema, pero el tiempo de inactividad puede ser prolongado según el tamaño de los datos).	Significativo, porque todos los inquilinos comparten las mismas mesas. (La fragmentación de la base de datos requiere copiar todo a otra instancia y realizar un paso adicional para limpiar los datos de los inquilinos). Lo más probable es que requiera un cambio en la lógica de la aplicación.

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Tiempo de base de datos para actualizaciones de la versión principal	Tiempo de inactividad estándar. (Depende del tamaño del catálogo del sistema PostgreSQL).	Probablemente un tiempo de inactividad más prolongado. (Según el tamaño del catálogo del sistema, el tiempo variará. (Las tablas del catálogo del sistema PostgreSQL también se duplican en todas las bases de datos)	Probablemente un tiempo de inactividad más prolongado. (Según el tamaño del catálogo del sistema PostgreSQL, el tiempo variará).	Tiempo de inactividad estándar. (Depende del tamaño del catálogo del sistema PostgreSQL).
Sobrecarga de administración (por ejemplo, para el análisis de registros de bases de datos o la supervisión de trabajos de respaldo)	esfuerzo significativo	Esfuerzo mínimo.	Esfuerzo mínimo.	Esfuerzo mínimo.

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Disponibilidad a nivel de inquilino	El más alto. (Cada inquilino fracasa y se recupera de forma independiente).	Mayor alcance de impacto. (Todos los inquilinos fallan y se recuperan juntos en caso de problemas de hardware o recursos).	Mayor alcance de impacto. (Todos los inquilinos fallan y se recuperan juntos en caso de problemas de hardware o recursos).	Mayor alcance de impacto. (Todos los inquilinos fallan y se recuperan juntos en caso de problemas de hardware o recursos).
Esfuerzo de respaldo y recuperación a nivel de inquilino	Menor esfuerzo. (Se puede realizar la copia de seguridad y de cada inquilino de seguridad.)	Esfuerzo moderado. (Utilice la exportación e importación lógicas para cada inquilino. Se requiere algo de codificación y automatización).	Esfuerzo moderado. (Utilice la exportación e importación lógicas para cada inquilino. Se requiere algo de codificación y automatización).	Esfuerzo significativo. (Todos los inquilinos comparten las mismas mesas).
Esfuerzo de point-in-time recuperación a nivel de inquilino	Esfuerzo mínimo. (Utilice la recuperación puntual mediante instantáneas o utilice el retroceso en Amazon Aurora).	Esfuerzo moderado. (Utilice la restauración de instantáneas, seguida de la exportación/importación. Sin embargo, será una operación lenta.)	Esfuerzo moderado. (Utilice la restauración de instantáneas, seguida de la exportación/importación. Sin embargo, será una operación lenta.)	Esfuerzo y complejidad significativos.

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Nombre de esquema uniforme	El mismo nombre de esquema para cada inquilino.	El mismo nombre de esquema para cada inquilino.	Esquema diferente para cada inquilino.	Esquema común.
Personalización por inquilino (por ejemplo, columnas de tabla adicionales para un inquilino específico)	posibles.	posibles.	posibles.	Complicado (porque todos los inquilinos comparten las mismas mesas).
Eficiencia de la administración de catálogos en la capa de mapeo relaciona l de objetos (ORM) (por ejemplo, Ruby)	Eficiente (porque la conexión con el cliente es específica para un inquilino).	Eficiente (porque la conexión del cliente es específica de una base de datos).	Moderadamente eficiente. (Según el ORM utilizado , el modelo de seguridad del usuario y el rol y la <code>search_path</code> configuración, el cliente a veces almacena en caché los metadatos de todos los inquilinos, lo que provoca un uso elevado de la memoria de la conexión a la base de datos).	Eficiente (porque todos los inquilinos comparten las mismas mesas).

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Esfuerzo de presentación de informes consolidados	Esfuerzo significativo. (Debe utilizar contenedores de datos extranjeros [FDW] para consolidar los datos de todos los inquilinos o extraer, transformar y cargar [ETL] en otra base de datos de informes).	Esfuerzo significativo. (Debe utilizar los FDW para consolidar los datos de todos los inquilinos o ETL en otra base de datos de informes).	Esfuerzo moderado. (Puede agregar datos en todos los esquemas mediante uniones).	Esfuerzo mínimo. (Todos los datos de los inquilinos están en las mismas tablas, por lo que los informes son sencillos).
Instancia de solo lectura específica del inquilino para la presentación de informes (por ejemplo, en función de la suscripción)	Menor esfuerzo. (Cree una réplica de lectura).	Esfuerzo moderado. (Puede usar la replicación lógica o el AWS Database Migration Service [AWS DMS] para configurar).	Esfuerzo moderado. (Puede usar la replicación lógica o AWS DMS para configurar).	Complicado (porque todos los inquilinos comparten las mismas mesas).

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Aislamiento de datos	Lo mejor.	Mejor. (Puede administrar los permisos a nivel de base de datos mediante los roles de PostgreSQL).	Mejor. (Puede administrar los permisos a nivel de esquema mediante los roles de PostgreSQL).	Peor. (Como todos los inquilinos comparten las mismas tablas, hay que implementar funciones como la seguridad a nivel de fila [RLS] para aislar a los inquilinos).
Clave de cifrado de almacenamiento específica del inquilino	posibles. (Cada clúster de PostgreSQL puede tener su propia claveAWS Key Management Service [AWS KMS] para el cifrado del almacenamiento).	No es posible. (Todos los inquilinos comparten la misma clave KMS para el cifrado del almacenamiento).	No es posible. (Todos los inquilinos comparten la misma clave KMS para el cifrado del almacenamiento).	No es posible. (Todos los inquilinos comparten la misma clave KMS para el cifrado del almacenamiento).

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Uso de AWS Identity and Access Management (IAM) para la autenticación de bases de datos para cada inquilino	posibles.	posibles.	Posible (al tener usuarios de PostgreSQL separados para cada esquema).	No es posible (porque las mesas las comparten todos los inquilinos).
costo de infraestructura	El más alto (porque no se comparte nada).	Moderado.	Moderado.	Más baja.
Duplicación de datos y uso del almacenamiento	El agregado más alto de todos los inquilinos. (Las tablas del catálogo del sistema PostgreSQL y los datos estáticos y comunes de la aplicación se duplican en todos los inquilinos).	El agregado más alto de todos los inquilinos. (Las tablas del catálogo del sistema PostgreSQL y los datos estáticos y comunes de la aplicación se duplican en todos los inquilinos).	Moderado. (Los datos estáticos y comunes de la aplicación pueden estar en un esquema común y otros inquilinos pueden acceder a ellos).	Mínimo. (Sin duplicación de datos. Los datos estáticos y comunes de la aplicación pueden estar en el mismo esquema).

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Monitoreo centrado en el inquilino (averigüe rápidamente qué inquilino está causando problemas)	Menor esfuerzo. (Como cada inquilino es monitoreado por separado, es fácil comprobar la actividad de un inquilino específico).	Esfuerzo moderado. (Como todos los inquilinos comparten el mismo recurso físico, hay que aplicar filtros adicionales para comprobar la actividad de un inquilino específico).	Esfuerzo moderado. (Como todos los inquilinos comparten el mismo recurso físico, hay que aplicar filtros adicionales para comprobar la actividad de un inquilino específico).	Esfuerzo significativo. (Como todos los inquilinos comparten todos los recursos, incluidas las tablas, debe utilizar la captura de variables de enlace para comprobar a qué inquilino pertenece una consulta SQL específica).
Gestión centralizada y supervisión de la salud y la actividad	Esfuerzo significativo (para establecer un monitoreo central y un centro de comando central).	Esfuerzo moderado (porque todos los inquilinos comparten la misma instancia).	Esfuerzo moderado (porque todos los inquilinos comparten la misma instancia).	Esfuerzo mínimo (porque todos los inquilinos comparten los mismos recursos, incluido el esquema).

	Silo	Puente con bases de datos independientes	Puente con esquemas separados	piscina
Posibilidades de que el identificador de objeto (OID) y el ID de la transacción (XID) coincidan	Mínimo.	Alta. (Debido a que OID, XID es un contador único para todo el clúster de PostgreSQL y puede haber problemas al pasar la aspiradora de forma eficaz en las bases de datos físicas).	Moderado. (Porque OID, XID es un contador único para todo el clúster de PostgreSQL).	Alta. (Por ejemplo, una sola tabla puede alcanzar el límite de 4 mil millones de TOAST OID, según el número de out-of-line columnas).

Recomendaciones de seguridad de nivel de fila

La seguridad de nivel de fila (RLS) es necesaria para mantener el aislamiento de los datos de los inquilinos en un modelo agrupado con PostgreSQL. El RLS centraliza la aplicación de las políticas de aislamiento a nivel de base de datos y elimina la carga de mantener este aislamiento para los desarrolladores de software. La forma más común de implementar RLS es habilitar esta función en el DBMS de PostgreSQL. El RLS implica filtrar el acceso a las filas de datos en función de un valor de una columna específica. Puede utilizar dos métodos para filtrar el acceso a los datos:

- Una columna de datos especificada en una tabla se compara con el valor del usuario actual de PostgreSQL. Ese usuario puede acceder a los valores de la columna que son equivalentes a los del usuario de PostgreSQL que ha iniciado sesión.
- Una columna de datos especificada en una tabla se compara con el valor de una variable de tiempo de ejecución establecida por la aplicación. Los valores de la columna que son equivalentes a la variable de tiempo de ejecución están disponibles durante esa sesión.

Se prefiere la segunda opción, porque la primera requiere la creación de un nuevo usuario de PostgreSQL para cada inquilino. En cambio, una aplicación SaaS que utilice PostgreSQL debería ser responsable de establecer un contexto específico para cada inquilino en tiempo de ejecución cuando consulta PostgreSQL. Esto tendrá el efecto de hacer cumplir el RLS. También puede activar el RLS sobre una table-by-table base. Como práctica recomendada, se debe habilitar el RLS en todas las tablas que contienen datos de inquilinos.

En el siguiente ejemplo se crean dos tablas y se ha activado el RLS. En este ejemplo se compara una columna de datos con el valor de la variable de tiempo de ejecución `app.current_tenant`.

```
-- Create a table for our tenants with indexes on the primary key and the tenant's name
CREATE TABLE tenant (
    tenant_id UUID DEFAULT uuid_generate_v4() PRIMARY KEY,
    name VARCHAR(255) UNIQUE,
    status VARCHAR(64) CHECK (status IN ('active', 'suspended', 'disabled')),
    tier VARCHAR(64) CHECK (tier IN ('gold', 'silver', 'bronze'))
);

-- Create a table for users of a tenant
CREATE TABLE tenant_user (
    user_id UUID DEFAULT uuid_generate_v4() PRIMARY KEY,
    tenant_id UUID NOT NULL REFERENCES tenant (tenant_id) ON DELETE RESTRICT,
```

```
email VARCHAR(255) NOT NULL UNIQUE,  
given_name VARCHAR(255) NOT NULL CHECK (given_name <> ''),  
family_name VARCHAR(255) NOT NULL CHECK (family_name <> '')  
);  
  
-- Turn on RLS  
ALTER TABLE tenant ENABLE ROW LEVEL SECURITY;  
  
-- Restrict read and write actions so tenants can only see their rows  
-- Cast the UUID value in tenant_id to match the type current_setting  
-- This policy implies a WITH CHECK that matches the USING clause  
CREATE POLICY tenant_isolation_policy ON tenant  
USING (tenant_id = current_setting('app.current_tenant')::UUID);  
  
-- And do the same for the tenant users  
ALTER TABLE tenant_user ENABLE ROW LEVEL SECURITY;  
  
CREATE POLICY tenant_user_isolation_policy ON tenant_user  
USING (tenant_id = current_setting('app.current_tenant')::UUID);
```

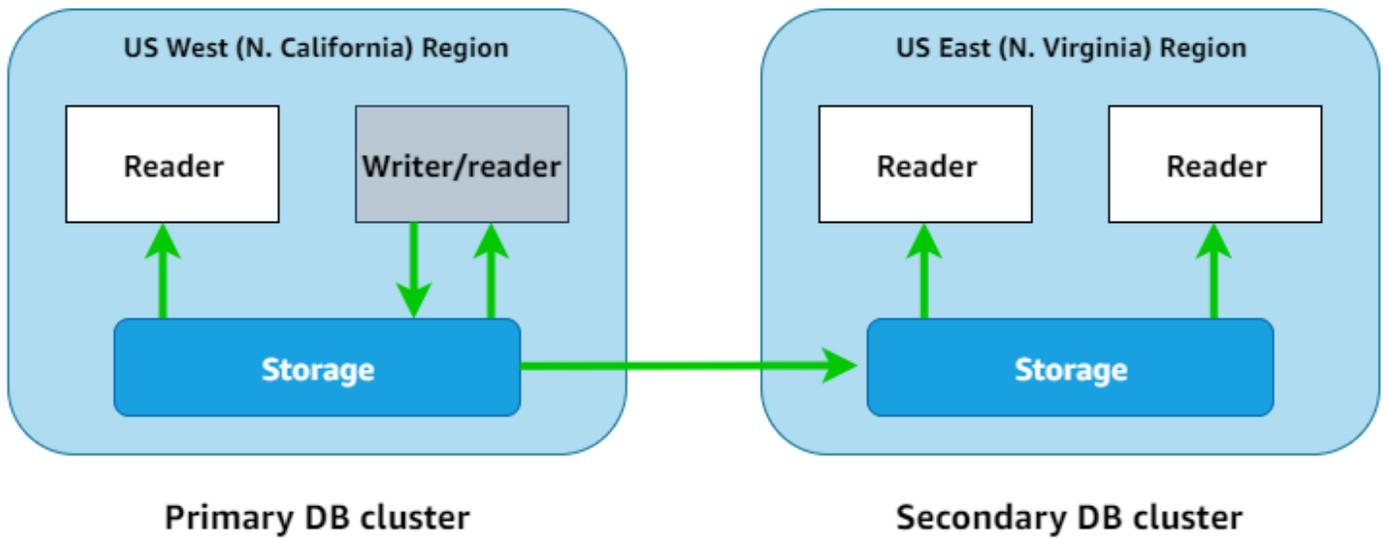
Para obtener más información, consulte la publicación de blog sobre el [aislamiento de datos de varios usuarios con seguridad de nivel de fila de PostgreSQL](#). El equipo de AWS SaaS Factory también tiene [algunos ejemplos GitHub](#) para ayudar a implementar el RLS.

Disponibilidad de PostgreSQL para el modelo de pool

Los modelos de pool, por su naturaleza, solo tienen una instancia de PostgreSQL. Por lo tanto, diseñar su aplicación para una alta disponibilidad es crucial. Una falla o interrupción de una base de datos agrupada provoca que su aplicación se degrade o quede inaccesible para todos sus usuarios.

Las instancias de base de datos de Amazon RDS for PostgreSQL se pueden hacer redundantes en dos zonas de disponibilidad si se habilita la función de alta disponibilidad. Para obtener más información, consulte [Alta disponibilidad \(Multi-AZ\) para Amazon RDS](#) en la documentación de Amazon RDS. Para la conmutación por error entre regiones, puede crear una réplica de lectura en otra región. AWS (Esta réplica de lectura debe promocionarse como parte de un proceso de conmutación por error). Además, puede replicar las copias de seguridad replicadas en todas AWS las regiones para su recuperación. Para obtener más información, consulte [Replicación de copias de seguridad automatizadas en otra AWS región](#) en la documentación de Amazon RDS.

Aurora, compatible con PostgreSQL, realiza copias de seguridad automáticas de los datos de manera que puede soportar el fallo de varias zonas de disponibilidad. (Consulte [Alta disponibilidad de Amazon Aurora](#) en la documentación de Aurora). Para que Aurora sea más resistente y se recupere más rápido, puede crear réplicas de lectura de Aurora en otras zonas de disponibilidad. Puede usar las bases de datos globales de Aurora para replicar datos en cinco AWS regiones adicionales para la recuperación entre regiones y la conmutación por error automática. (Consulte [Uso de las bases de datos globales de Amazon Aurora](#) en la documentación de Aurora). Además, puede habilitar el [reenvío de escritura con las](#) bases de datos globales de Aurora para lograr una alta disponibilidad en múltiples Regiones de AWS.



Independientemente de si utiliza Amazon RDS para PostgreSQL o si es compatible con Aurora PostgreSQL, le recomendamos que implemente funciones de alta disponibilidad para mitigar el impacto de cualquier interrupción en todas las aplicaciones SaaS multiusuario que utilizan un modelo de grupo.

Prácticas recomendadas

En esta sección se enumeran algunas de las conclusiones de alto nivel de esta guía. Para obtener información detallada sobre cada punto, siga los enlaces a las secciones correspondientes.

CompareAWS las opciones de PostgreSQL administrado

AWS ofrece dos formas principales de ejecutar PostgreSQL en un entorno administrado. (En este contexto, administrado significa que unAWS servicio admite parcial o totalmente la infraestructura de PostgreSQL y el DBMS). Las opciones administradas de PostgreSQLAWS tienen la ventaja de automatizar las copias de seguridad, la conmutación por error, la optimización y cierta administración de PostgreSQL. Como opciones administradas,AWS ofrece Amazon Aurora PostgreSQL-Compatible Edition y Amazon Relational Database Service (Amazon RDS) para PostgreSQL. Puede seleccionar la mejor opción entre estos dos modelos analizando su caso de uso de PostgreSQL. Para obtener más información, consulte la sección [Cómo elegir entre Amazon RDS y Aurora](#) de esta guía.

Seleccione un modelo de particionamiento SaaS para varios usuarios

Puede elegir entre tres modelos de particionamiento SaaS aplicables a PostgreSQL: silo, bridge y pool. Cada modelo tiene ventajas y desventajas, y usted debe elegir el modelo más óptimo según su caso de uso. Amazon RDS for PostgreSQL y Aurora compatible con PostgreSQL son compatibles con los tres modelos. La elección de un modelo es fundamental para mantener el aislamiento de los datos de los inquilinos en sus aplicaciones SaaS. Para obtener una descripción detallada de estos modelos, consulte la sección [Modelos de particionamiento SaaS multiusuario para PostgreSQL](#) de esta guía.

Utilice la seguridad de nivel de fila para los modelos de particionamiento SaaS de grupos

La seguridad de nivel de fila (RLS) es necesaria para mantener el aislamiento de los datos de los inquilinos en un modelo de grupo con PostgreSQL. Esto se debe a que no existe una separación lógica entre la infraestructura, las bases de datos de PostgreSQL o los esquemas por inquilino en un modelo de grupo. El RLS centraliza la aplicación de las políticas de aislamiento a nivel de base de

datos y elimina la carga de mantener este aislamiento para los desarrolladores de software. Puede usar RLS para limitar las operaciones de la base de datos a un inquilino específico. Para obtener más información y un ejemplo, consulte la sección [Recomendaciones de seguridad a nivel de fila](#) de esta guía.

Preguntas frecuentes

Esta sección proporciona respuestas a las preguntas más frecuentes sobre la implementación de PostgreSQL administrado en aplicaciones SaaS para varios usuarios.

¿Qué opciones administradas de PostgreSQL AWS ofrece?

AWS ofrece [Amazon Aurora compatible con PostgreSQL](#) y [Amazon Relational Database Service \(Amazon RDS\) para PostgreSQL](#). AWS también cuenta con un [amplio catálogo de ofertas de bases de datos gestionadas](#).

¿Qué servicio es óptimo para las aplicaciones SaaS?

Puede usar aplicaciones SaaS compatibles con Aurora PostgreSQL y Amazon RDS for PostgreSQL para SaaS y todos los modelos de particionamiento de SaaS que se describen en esta guía. Estos dos servicios tienen diferencias en cuanto a la escalabilidad, la recuperación ante fallos, la conmutación por error, las opciones de almacenamiento, la alta disponibilidad, la recuperación ante desastres, el respaldo y las clases de instancias disponibles para cada opción. La elección óptima dependerá de su caso de uso específico. Utilice la [matriz de decisiones](#) de esta guía para elegir la mejor opción para su caso de uso.

¿Qué requisitos únicos debo tener en cuenta si decido utilizar una base de datos PostgreSQL con una aplicación SaaS multiusuario?

Como ocurre con cualquier almacén de datos utilizado con una aplicación SaaS, la consideración más importante es el método para mantener el aislamiento de los datos de los inquilinos. Como se explica en esta guía, hay varias maneras de lograr el aislamiento de los datos de los inquilinos con las ofertas AWS administradas de PostgreSQL. Además, debe considerar el aislamiento del rendimiento por inquilino para cualquier implementación de PostgreSQL.

¿Qué modelos puedo usar para mantener el aislamiento de los datos de los inquilinos con PostgreSQL?

Puede utilizar los modelos de silos, puentes y agrupaciones como estrategias de particionamiento de SaaS para mantener el aislamiento de los datos de los inquilinos. Para analizar estos modelos

y cómo se pueden aplicar a PostgreSQL, consulte la sección [Modelos de particionamiento SaaS multiusuario para PostgreSQL](#) de esta guía.

¿Cómo mantengo el aislamiento de los datos de los inquilinos con una única base de datos PostgreSQL que se comparte entre varios inquilinos?

PostgreSQL admite una función de seguridad de nivel de fila (RLS) que puede utilizar para reforzar el aislamiento de los datos de los inquilinos en una única base de datos o instancia de PostgreSQL. Además, puede aprovisionar bases de datos PostgreSQL independientes por inquilino en una sola instancia o crear esquemas por inquilino para lograr este objetivo. Para conocer las ventajas y desventajas de estos enfoques, consulte la sección [Recomendaciones de seguridad a nivel de fila](#) de esta guía.

Pasos siguientes

AWS ofrece dos opciones para operar PostgreSQL administrado: compatible con Aurora PostgreSQL y Amazon RDS for PostgreSQL. Le recomendamos que evalúe los dos servicios y elija la opción que mejor se adapte a su caso de uso específico para sus aplicaciones SaaS para varios usuarios. Cumplir con un modelo de particionamiento de SaaS puede garantizar que una aplicación SaaS que utilice PostgreSQL cumpla estrictamente con las mejores prácticas para mantener la tenencia. Los modelos de particionamiento de silos, puentes y grupos de SaaS admiten muchos casos de uso de SaaS. Estos modelos ofrecen diversas ventajas entre factores como el aislamiento del rendimiento, los gastos operativos y la seguridad de los inquilinos.

Próximos pasos:

- [Evalúe la compatibilidad con Aurora PostgreSQL y Amazon RDS for PostgreSQL](#) y elija la mejor opción para su aplicación SaaS.
- [Seleccione un modelo de particionamiento SaaS](#) que cumpla con los requisitos de su aplicación: silo, puente o piscina.
- Implemente PostgreSQL de acuerdo con el modelo de particionamiento SaaS seleccionado.

Recursos

Referencias

- [Estrategias de almacenamiento de SaaS: creación de un modelo de almacenamiento multiusuario](#) (documento técnico) AWS AWS
- [Recuperación ante desastres entre regiones mediante Amazon Aurora Global Database para Amazon Aurora AWS PostgreSQL](#) (entrada del blog)
- [Aislamiento de datos multiusuario con seguridad a nivel de fila de PostgreSQL](#) (entrada del blog) AWS
- [Uso de Amazon Aurora PostgreSQL](#) (documentación de Aurora)
- [PostgreSQL en Amazon RDS](#) (documentación de Amazon RDS)

Socios

- [Amazon Aurora para socios de PostgreSQL](#)
- [Amazon RDS para socios de PostgreSQL](#)

Historial de documentos

En la siguiente tabla, se describen cambios significativos de esta guía. Si quiere recibir notificaciones de futuras actualizaciones, puede suscribirse a las [notificaciones RSS](#).

Cambio	Descripción	Fecha
Actualización	Actualizaciones para reflejar la disponibilidad del reenvío de escrituras en Aurora.	29 de abril de 2024
Actualización	Se actualizó la tabla comparativa de Amazon RDS y Aurora .	21 de octubre de 2022
=	Publicación inicial	30 de septiembre de 2021

AWS Glosario de orientación prescriptiva

Los siguientes son términos de uso común en las estrategias, guías y patrones proporcionados por AWS Prescriptive Guidance. Para sugerir entradas, utilice el enlace [Enviar comentarios](#) al final del glosario.

Números

Las 7 R

Siete estrategias de migración comunes para trasladar aplicaciones a la nube. Estas estrategias se basan en las 5 R que Gartner identificó en 2011 y consisten en lo siguiente:

- **Refactorizar/rediseñar:** traslade una aplicación y modifique su arquitectura mediante el máximo aprovechamiento de las características nativas en la nube para mejorar la agilidad, el rendimiento y la escalabilidad. Por lo general, esto implica trasladar el sistema operativo y la base de datos. Ejemplo: Migre la base de datos de Oracle en las instalaciones a Amazon Aurora PostgreSQL-Compatible Edition.
- **Redefinir la plataforma (transportar y redefinir):** traslade una aplicación a la nube e introduzca algún nivel de optimización para aprovechar las capacidades de la nube. Ejemplo: migre su base de datos Oracle local a Amazon Relational Database Service (Amazon RDS) para Oracle in the Cloud. AWS
- **Recomprar (readquirir):** cambie a un producto diferente, lo cual se suele llevar a cabo al pasar de una licencia tradicional a un modelo SaaS. Ejemplo: Migre el sistema de administración de las relaciones con los clientes (CRM) a Salesforce.com.
- **Volver a alojar (migrar mediante lift-and-shift):** traslade una aplicación a la nube sin realizar cambios para aprovechar las capacidades de la nube. Ejemplo: migre su base de datos Oracle local a Oracle en una instancia EC2 en la nube. AWS
- **Reubicar:** (migrar el hipervisor mediante lift and shift): traslade la infraestructura a la nube sin comprar equipo nuevo, reescribir aplicaciones o modificar las operaciones actuales. Este escenario de migración es específico de VMware Cloud on AWS, que admite la compatibilidad de máquinas virtuales (VM) y la portabilidad de las cargas de trabajo entre su entorno local y. AWS Puede utilizar las tecnologías de VMware Cloud Foundation desde los centros de datos en las instalaciones al migrar una infraestructura a VMware Cloud en AWS. Ejemplo: traslade el hipervisor que aloja su base de datos de Oracle a VMware Cloud on. AWS

- **Retener (revisitar):** conserve las aplicaciones en el entorno de origen. Estas pueden incluir las aplicaciones que requieren una refactorización importante, que desee posponer para más adelante, y las aplicaciones heredadas que desee retener, ya que no hay ninguna justificación empresarial para migrarlas.
- **Retirar:** retire o elimine las aplicaciones que ya no sean necesarias en un entorno de origen.

A

ABAC

Consulte el control de acceso basado en [atributos](#).

servicios abstractos

Consulte [servicios gestionados](#).

ACID

Consulte [atomicidad, consistencia, aislamiento y durabilidad](#).

migración activa-activa

Método de migración de bases de datos en el que las bases de datos de origen y destino se mantienen sincronizadas (mediante una herramienta de replicación bidireccional o mediante operaciones de escritura doble) y ambas bases de datos gestionan las transacciones de las aplicaciones conectadas durante la migración. Este método permite la migración en lotes pequeños y controlados, en lugar de requerir una transición única. Es más flexible, pero requiere más trabajo que la migración [activa-pasiva](#).

migración activa-pasiva

Método de migración de bases de datos en el que las bases de datos de origen y destino se mantienen sincronizadas, pero solo la base de datos de origen gestiona las transacciones de las aplicaciones conectadas, mientras los datos se replican en la base de datos de destino. La base de datos de destino no acepta ninguna transacción durante la migración.

función agregada

Función SQL que opera en un grupo de filas y calcula un único valor de retorno para el grupo. Entre los ejemplos de funciones agregadas se incluyen SUM y MAX.

IA

Véase [inteligencia artificial](#).

AIOps

Consulte las [operaciones de inteligencia artificial](#).

anonimización

El proceso de eliminar permanentemente la información personal de un conjunto de datos. La anonimización puede ayudar a proteger la privacidad personal. Los datos anonimizados ya no se consideran datos personales.

antipatrones

Una solución que se utiliza con frecuencia para un problema recurrente en el que la solución es contraproducente, ineficaz o menos eficaz que una alternativa.

control de aplicaciones

Un enfoque de seguridad que permite el uso únicamente de aplicaciones aprobadas para ayudar a proteger un sistema contra el malware.

cartera de aplicaciones

Recopilación de información detallada sobre cada aplicación que utiliza una organización, incluido el costo de creación y mantenimiento de la aplicación y su valor empresarial. Esta información es clave para [el proceso de detección y análisis de la cartera](#) y ayuda a identificar y priorizar las aplicaciones que se van a migrar, modernizar y optimizar.

inteligencia artificial (IA)

El campo de la informática que se dedica al uso de tecnologías informáticas para realizar funciones cognitivas que suelen estar asociadas a los seres humanos, como el aprendizaje, la resolución de problemas y el reconocimiento de patrones. Para más información, consulte [¿Qué es la inteligencia artificial?](#)

operaciones de inteligencia artificial (AIOps)

El proceso de utilizar técnicas de machine learning para resolver problemas operativos, reducir los incidentes operativos y la intervención humana, y mejorar la calidad del servicio. Para obtener más información sobre cómo se utiliza AIOps en la estrategia de migración de AWS , consulte la [Guía de integración de operaciones](#).

cifrado asimétrico

Algoritmo de cifrado que utiliza un par de claves, una clave pública para el cifrado y una clave privada para el descifrado. Puede compartir la clave pública porque no se utiliza para el descifrado, pero el acceso a la clave privada debe estar sumamente restringido.

atomicidad, consistencia, aislamiento, durabilidad (ACID)

Conjunto de propiedades de software que garantizan la validez de los datos y la fiabilidad operativa de una base de datos, incluso en caso de errores, cortes de energía u otros problemas.

control de acceso basado en atributos (ABAC)

La práctica de crear permisos detallados basados en los atributos del usuario, como el departamento, el puesto de trabajo y el nombre del equipo. Para obtener más información, consulte [ABAC AWS en la](#) documentación AWS Identity and Access Management (IAM).

origen de datos fidedigno

Ubicación en la que se almacena la versión principal de los datos, que se considera la fuente de información más fiable. Puede copiar los datos del origen de datos autorizado a otras ubicaciones con el fin de procesarlos o modificarlos, por ejemplo, anonimizarlos, redactarlos o seudonimizarlos.

Zona de disponibilidad

Una ubicación distinta dentro de una Región de AWS que está aislada de los fallos en otras zonas de disponibilidad y que proporciona una conectividad de red económica y de baja latencia a otras zonas de disponibilidad de la misma región.

AWS Marco de adopción de la nube (AWS CAF)

Un marco de directrices y mejores prácticas AWS para ayudar a las organizaciones a desarrollar un plan eficiente y eficaz para migrar con éxito a la nube. AWS CAF organiza la orientación en seis áreas de enfoque denominadas perspectivas: negocios, personas, gobierno, plataforma, seguridad y operaciones. Las perspectivas empresariales, humanas y de gobernanza se centran en las habilidades y los procesos empresariales; las perspectivas de plataforma, seguridad y operaciones se centran en las habilidades y los procesos técnicos. Por ejemplo, la perspectiva humana se dirige a las partes interesadas que se ocupan de los Recursos Humanos (RR. HH.), las funciones del personal y la administración de las personas. Desde esta perspectiva, AWS CAF proporciona orientación para el desarrollo, la formación y la comunicación de las personas a fin de preparar a la organización para una adopción exitosa de la nube. Para obtener más información, consulte la [Página web de AWS CAF](#) y el [Documento técnico de AWS CAF](#).

AWS Marco de calificación de la carga de trabajo (AWS WQF)

Herramienta que evalúa las cargas de trabajo de migración de bases de datos, recomienda estrategias de migración y proporciona estimaciones de trabajo. AWS WQF se incluye con AWS Schema Conversion Tool (). AWS SCT Analiza los esquemas de bases de datos y los objetos de código, el código de las aplicaciones, las dependencias y las características de rendimiento y proporciona informes de evaluación.

B

Un bot malo

Un [bot](#) destinado a interrumpir o causar daño a personas u organizaciones.

BCP

Consulte la [planificación de la continuidad del negocio](#).

gráfico de comportamiento

Una vista unificada e interactiva del comportamiento de los recursos y de las interacciones a lo largo del tiempo. Puede utilizar un gráfico de comportamiento con Amazon Detective para examinar los intentos de inicio de sesión fallidos, las llamadas sospechosas a la API y acciones similares. Para obtener más información, consulte [Datos en un gráfico de comportamiento](#) en la documentación de Detective.

sistema big-endian

Un sistema que almacena primero el byte más significativo. Véase también [endianness](#).

clasificación binaria

Un proceso que predice un resultado binario (una de las dos clases posibles). Por ejemplo, es posible que su modelo de ML necesite predecir problemas como “¿Este correo electrónico es spam o no es spam?” o “¿Este producto es un libro o un automóvil?”.

filtro de floración

Estructura de datos probabilística y eficiente en términos de memoria que se utiliza para comprobar si un elemento es miembro de un conjunto.

implementación azul/verde

Una estrategia de despliegue en la que se crean dos entornos separados pero idénticos. La versión actual de la aplicación se ejecuta en un entorno (azul) y la nueva versión de la aplicación en el otro entorno (verde). Esta estrategia le ayuda a revertirla rápidamente con un impacto mínimo.

bot

Una aplicación de software que ejecuta tareas automatizadas a través de Internet y simula la actividad o interacción humana. Algunos bots son útiles o beneficiosos, como los rastreadores web que indexan información en Internet. Algunos otros bots, conocidos como bots malos, tienen como objetivo interrumpir o causar daños a personas u organizaciones.

botnet

Redes de [bots](#) que están infectadas por [malware](#) y que están bajo el control de una sola parte, conocida como pastor u operador de bots. Las botnets son el mecanismo más conocido para escalar los bots y su impacto.

rama

Área contenida de un repositorio de código. La primera rama que se crea en un repositorio es la rama principal. Puede crear una rama nueva a partir de una rama existente y, a continuación, desarrollar características o corregir errores en la rama nueva. Una rama que se genera para crear una característica se denomina comúnmente rama de característica. Cuando la característica se encuentra lista para su lanzamiento, se vuelve a combinar la rama de característica con la rama principal. Para obtener más información, consulte [Acerca de las sucursales](#) (GitHub documentación).

acceso con cristales rotos

En circunstancias excepcionales y mediante un proceso aprobado, un usuario puede acceder rápidamente a un sitio para el Cuenta de AWS que normalmente no tiene permisos de acceso. Para obtener más información, consulte el indicador [Implemente procedimientos de rotura de cristales en la guía Well-Architected AWS](#) .

estrategia de implementación sobre infraestructura existente

La infraestructura existente en su entorno. Al adoptar una estrategia de implementación sobre infraestructura existente para una arquitectura de sistemas, se diseña la arquitectura en función de las limitaciones de los sistemas y la infraestructura actuales. Si está ampliando

la infraestructura existente, puede combinar las estrategias de implementación sobre infraestructuras existentes y de [implementación desde cero](#).

caché de búfer

El área de memoria donde se almacenan los datos a los que se accede con más frecuencia.

capacidad empresarial

Lo que hace una empresa para generar valor (por ejemplo, ventas, servicio al cliente o marketing). Las arquitecturas de microservicios y las decisiones de desarrollo pueden estar impulsadas por las capacidades empresariales. Para obtener más información, consulte la sección [Organizado en torno a las capacidades empresariales](#) del documento técnico [Ejecutar microservicios en contenedores en AWS](#).

planificación de la continuidad del negocio (BCP)

Plan que aborda el posible impacto de un evento disruptivo, como una migración a gran escala en las operaciones y permite a la empresa reanudar las operaciones rápidamente.

C

CAF

[Consulte el marco AWS de adopción de la nube](#).

despliegue canario

El lanzamiento lento e incremental de una versión para los usuarios finales. Cuando se tiene confianza, se despliega la nueva versión y se reemplaza la versión actual en su totalidad.

CCoE

Consulte el [Centro de excelencia en la nube](#).

CDC

Consulte la [captura de datos de cambios](#).

captura de datos de cambio (CDC)

Proceso de seguimiento de los cambios en un origen de datos, como una tabla de base de datos, y registro de los metadatos relacionados con el cambio. Puede utilizar los CDC para

diversos fines, como auditar o replicar los cambios en un sistema de destino para mantener la sincronización.

ingeniería del caos

Introducir intencionalmente fallos o eventos disruptivos para poner a prueba la resiliencia de un sistema. Puedes usar [AWS Fault Injection Service \(AWS FIS\)](#) para realizar experimentos que estresen tus AWS cargas de trabajo y evalúen su respuesta.

CI/CD

Consulte la [integración continua y la entrega continua](#).

clasificación

Un proceso de categorización que permite generar predicciones. Los modelos de ML para problemas de clasificación predicen un valor discreto. Los valores discretos siempre son distintos entre sí. Por ejemplo, es posible que un modelo necesite evaluar si hay o no un automóvil en una imagen.

cifrado del cliente

Cifrado de datos localmente, antes de que el objetivo los Servicio de AWS reciba.

Centro de excelencia en la nube (CCoE)

Equipo multidisciplinario que impulsa los esfuerzos de adopción de la nube en toda la organización, incluido el desarrollo de las prácticas recomendadas en la nube, la movilización de recursos, el establecimiento de plazos de migración y la dirección de la organización durante las transformaciones a gran escala. Para obtener más información, consulte las [publicaciones de CCoE](#) en el blog de estrategia empresarial en la AWS nube.

computación en la nube

La tecnología en la nube que se utiliza normalmente para la administración de dispositivos de IoT y el almacenamiento de datos de forma remota. La computación en la nube suele estar conectada a la tecnología de [computación perimetral](#).

modelo operativo en la nube

En una organización de TI, el modelo operativo que se utiliza para crear, madurar y optimizar uno o más entornos de nube. Para obtener más información, consulte [Creación de su modelo operativo de nube](#).

etapas de adopción de la nube

Las cuatro fases por las que suelen pasar las organizaciones cuando migran a la AWS nube:

- Proyecto: ejecución de algunos proyectos relacionados con la nube con fines de prueba de concepto y aprendizaje
- Fundamento: realización de inversiones fundamentales para escalar la adopción de la nube (p. ej., crear una zona de aterrizaje, definir un CCoE, establecer un modelo de operaciones)
- Migración: migración de aplicaciones individuales
- Reinención: optimización de productos y servicios e innovación en la nube

Stephen Orban definió estas etapas en la entrada del blog [The Journey Toward Cloud-First & the Stages of Adoption](#), del blog AWS Cloud Enterprise Strategy. Para obtener información sobre su relación con la estrategia de AWS migración, consulte la guía de [preparación para la migración](#).

CMDB

Consulte la [base de datos de administración de la configuración](#).

repositorio de código

Una ubicación donde el código fuente y otros activos, como documentación, muestras y scripts, se almacenan y actualizan mediante procesos de control de versiones. Los repositorios en la nube más comunes incluyen GitHub o AWS CodeCommit. Cada versión del código se denomina rama. En una estructura de microservicios, cada repositorio se encuentra dedicado a una única funcionalidad. Una sola canalización de CI/CD puede utilizar varios repositorios.

caché en frío

Una caché de búfer que está vacía no está bien poblada o contiene datos obsoletos o irrelevantes. Esto afecta al rendimiento, ya que la instancia de la base de datos debe leer desde la memoria principal o el disco, lo que es más lento que leer desde la memoria caché del búfer.

datos fríos

Datos a los que se accede con poca frecuencia y que suelen ser históricos. Al consultar este tipo de datos, normalmente se aceptan consultas lentas. Trasladar estos datos a niveles o clases de almacenamiento de menor rendimiento y menos costosos puede reducir los costos.

visión artificial (CV)

Campo de la [IA](#) que utiliza el aprendizaje automático para analizar y extraer información de formatos visuales, como imágenes y vídeos digitales. Por ejemplo, AWS Panorama ofrece

dispositivos que añaden CV a las redes de cámaras locales, y Amazon SageMaker proporciona algoritmos de procesamiento de imágenes para CV.

desviación de configuración

En el caso de una carga de trabajo, un cambio de configuración con respecto al estado esperado. Puede provocar que la carga de trabajo deje de cumplir las normas y, por lo general, es gradual e involuntario.

base de datos de administración de configuración (CMDB)

Repositorio que almacena y administra información sobre una base de datos y su entorno de TI, incluidos los componentes de hardware y software y sus configuraciones. Por lo general, los datos de una CMDB se utilizan en la etapa de detección y análisis de la cartera de productos durante la migración.

paquete de conformidad

Conjunto de AWS Config reglas y medidas correctivas que puede reunir para personalizar sus comprobaciones de conformidad y seguridad. Puede implementar un paquete de conformidad como una entidad única en una región Cuenta de AWS y, o en una organización, mediante una plantilla YAML. Para obtener más información, consulta los [paquetes de conformidad](#) en la documentación. AWS Config

integración y entrega continuas (CI/CD)

El proceso de automatización de las etapas de origen, compilación, prueba, presentación y producción del proceso de lanzamiento del software. La CI/CD se describe comúnmente como una canalización. La CI/CD puede ayudarlo a automatizar los procesos, mejorar la productividad, mejorar la calidad del código y entregar con mayor rapidez. Para obtener más información, consulte [Beneficios de la entrega continua](#). CD también puede significar implementación continua. Para obtener más información, consulte [Entrega continua frente a implementación continua](#).

CV

Consulte [visión artificial](#).

D

datos en reposo

Datos que están estacionarios en la red, como los datos que se encuentran almacenados.

clasificación de datos

Un proceso para identificar y clasificar los datos de su red en función de su importancia y sensibilidad. Es un componente fundamental de cualquier estrategia de administración de riesgos de ciberseguridad porque lo ayuda a determinar los controles de protección y retención adecuados para los datos. La clasificación de datos es un componente del pilar de seguridad del AWS Well-Architected Framework. Para obtener más información, consulte [Clasificación de datos](#).

desviación de datos

Una variación significativa entre los datos de producción y los datos que se utilizaron para entrenar un modelo de machine learning, o un cambio significativo en los datos de entrada a lo largo del tiempo. La desviación de los datos puede reducir la calidad, la precisión y la imparcialidad generales de las predicciones de los modelos de machine learning.

datos en tránsito

Datos que se mueven de forma activa por la red, por ejemplo, entre los recursos de la red.

malla de datos

Un marco arquitectónico que proporciona una propiedad de datos distribuida y descentralizada con una administración y un gobierno centralizados.

minimización de datos

El principio de recopilar y procesar solo los datos estrictamente necesarios. Practicar la minimización de los datos Nube de AWS puede reducir los riesgos de privacidad, los costos y la huella de carbono de la analítica.

perímetro de datos

Un conjunto de barreras preventivas en su AWS entorno que ayudan a garantizar que solo las identidades confiables accedan a los recursos confiables desde las redes esperadas. Para obtener más información, consulte [Crear un perímetro de datos sobre](#) AWS

preprocesamiento de datos

Transformar los datos sin procesar en un formato que su modelo de ML pueda analizar fácilmente. El preprocesamiento de datos puede implicar eliminar determinadas columnas o filas y corregir los valores faltantes, incoherentes o duplicados.

procedencia de los datos

El proceso de rastrear el origen y el historial de los datos a lo largo de su ciclo de vida, por ejemplo, la forma en que se generaron, transmitieron y almacenaron los datos.

titular de los datos

Persona cuyos datos se recopilan y procesan.

almacenamiento de datos

Un sistema de administración de datos que respalde la inteligencia empresarial, como el análisis. Los almacenes de datos suelen contener grandes cantidades de datos históricos y, por lo general, se utilizan para consultas y análisis.

lenguaje de definición de datos (DDL)

Instrucciones o comandos para crear o modificar la estructura de tablas y objetos de una base de datos.

lenguaje de manipulación de datos (DML)

Instrucciones o comandos para modificar (insertar, actualizar y eliminar) la información de una base de datos.

DDL

Consulte el [lenguaje de definición de bases de datos](#) de datos.

conjunto profundo

Combinar varios modelos de aprendizaje profundo para la predicción. Puede utilizar conjuntos profundos para obtener una predicción más precisa o para estimar la incertidumbre de las predicciones.

aprendizaje profundo

Un subcampo del ML que utiliza múltiples capas de redes neuronales artificiales para identificar el mapeo entre los datos de entrada y las variables objetivo de interés.

defense-in-depth

Un enfoque de seguridad de la información en el que se distribuyen cuidadosamente una serie de mecanismos y controles de seguridad en una red informática para proteger la confidencialidad, la integridad y la disponibilidad de la red y de los datos que contiene. Al adoptar esta estrategia AWS, se añaden varios controles en diferentes capas de la AWS Organizations estructura para

ayudar a proteger los recursos. Por ejemplo, un *defense-in-depth* enfoque podría combinar la autenticación multifactorial, la segmentación de la red y el cifrado.

administrador delegado

En AWS Organizations, un servicio compatible puede registrar una cuenta de AWS miembro para administrar las cuentas de la organización y gestionar los permisos de ese servicio. Esta cuenta se denomina administrador delegado para ese servicio. Para obtener más información y una lista de servicios compatibles, consulte [Servicios que funcionan con AWS Organizations](#) en la documentación de AWS Organizations .

Implementación

El proceso de hacer que una aplicación, características nuevas o correcciones de código se encuentren disponibles en el entorno de destino. La implementación abarca implementar cambios en una base de código y, a continuación, crear y ejecutar esa base en los entornos de la aplicación.

entorno de desarrollo

Consulte [entorno](#).

control de detección

Un control de seguridad que se ha diseñado para detectar, registrar y alertar después de que se produzca un evento. Estos controles son una segunda línea de defensa, ya que lo advierten sobre los eventos de seguridad que han eludido los controles preventivos establecidos. Para obtener más información, consulte [Controles de detección](#) en Implementación de controles de seguridad en AWS.

asignación de flujos de valor para el desarrollo (DVSM)

Proceso que se utiliza para identificar y priorizar las restricciones que afectan negativamente a la velocidad y la calidad en el ciclo de vida del desarrollo de software. DVSM amplía el proceso de asignación del flujo de valor diseñado originalmente para las prácticas de fabricación ajustada. Se centra en los pasos y los equipos necesarios para crear y transferir valor a través del proceso de desarrollo de software.

gemelo digital

Representación virtual de un sistema del mundo real, como un edificio, una fábrica, un equipo industrial o una línea de producción. Los gemelos digitales son compatibles con el mantenimiento predictivo, la supervisión remota y la optimización de la producción.

tabla de dimensiones

En un [esquema en estrella](#), tabla más pequeña que contiene los atributos de datos sobre los datos cuantitativos de una tabla de hechos. Los atributos de la tabla de dimensiones suelen ser campos de texto o números discretos que se comportan como texto. Estos atributos se utilizan habitualmente para restringir consultas, filtrar y etiquetar conjuntos de resultados.

desastre

Un evento que impide que una carga de trabajo o un sistema cumplan sus objetivos empresariales en su ubicación principal de implementación. Estos eventos pueden ser desastres naturales, fallos técnicos o el resultado de acciones humanas, como una configuración incorrecta involuntaria o un ataque de malware.

recuperación de desastres (DR)

La estrategia y el proceso que se utilizan para minimizar el tiempo de inactividad y la pérdida de datos ocasionados por un [desastre](#). Para obtener más información, consulte [Recuperación ante desastres de cargas de trabajo en AWS: Recovery in the Cloud in the AWS Well-Architected Framework](#).

DML

Consulte el lenguaje de manipulación de [bases de datos](#).

diseño basado en el dominio

Un enfoque para desarrollar un sistema de software complejo mediante la conexión de sus componentes a dominios en evolución, o a los objetivos empresariales principales, a los que sirve cada componente. Este concepto lo introdujo Eric Evans en su libro, *Diseño impulsado por el dominio: abordando la complejidad en el corazón del software* (Boston: Addison-Wesley Professional, 2003). Para obtener información sobre cómo utilizar el diseño basado en dominios con el patrón de higos estranguladores, consulte [Modernización gradual de los servicios web antiguos de Microsoft ASP.NET \(ASMX\) mediante contenedores y Amazon API Gateway](#).

DR

Consulte [recuperación ante desastres](#).

detección de deriva

Seguimiento de las desviaciones con respecto a una configuración de referencia. Por ejemplo, puedes usarlo AWS CloudFormation para [detectar desviaciones en los recursos del sistema](#) o

puedes usarlo AWS Control Tower para [detectar cambios en tu landing zone](#) que puedan afectar al cumplimiento de los requisitos de gobierno.

DVSM

Consulte [el mapeo del flujo de valor del desarrollo](#).

E

EDA

Consulte el [análisis exploratorio de datos](#).

computación en la periferia

La tecnología que aumenta la potencia de cálculo de los dispositivos inteligentes en la periferia de una red de IoT. En comparación con [la computación en nube, la computación](#) perimetral puede reducir la latencia de la comunicación y mejorar el tiempo de respuesta.

cifrado

Proceso informático que transforma datos de texto plano, legibles por humanos, en texto cifrado.

clave de cifrado

Cadena criptográfica de bits aleatorios que se genera mediante un algoritmo de cifrado. Las claves pueden variar en longitud y cada una se ha diseñado para ser impredecible y única.

endianidad

El orden en el que se almacenan los bytes en la memoria del ordenador. Los sistemas big-endianos almacenan primero el byte más significativo. Los sistemas Little-Endian almacenan primero el byte menos significativo.

punto de conexión

[Consulte el punto final del servicio](#).

servicio de punto de conexión

Servicio que puede alojarse en una nube privada virtual (VPC) para compartir con otros usuarios. Puede crear un servicio de punto final AWS PrivateLink y conceder permisos a otros directores de cuentas de AWS o a AWS Identity and Access Management (IAM). Estas cuentas o entidades

principales pueden conectarse a su servicio de punto de conexión de forma privada mediante la creación de puntos de conexión de VPC de interfaz. Para obtener más información, consulte [Creación de un servicio de punto de conexión](#) en la documentación de Amazon Virtual Private Cloud (Amazon VPC).

planificación de recursos empresariales (ERP)

Un sistema que automatiza y gestiona los procesos empresariales clave (como la contabilidad, el [MES](#) y la gestión de proyectos) de una empresa.

cifrado de sobre

El proceso de cifrar una clave de cifrado con otra clave de cifrado. Para obtener más información, consulte el [cifrado de sobres](#) en la documentación de AWS Key Management Service (AWS KMS).

environment

Una instancia de una aplicación en ejecución. Los siguientes son los tipos de entornos más comunes en la computación en la nube:

- entorno de desarrollo: instancia de una aplicación en ejecución que solo se encuentra disponible para el equipo principal responsable del mantenimiento de la aplicación. Los entornos de desarrollo se utilizan para probar los cambios antes de promocionarlos a los entornos superiores. Este tipo de entorno a veces se denomina entorno de prueba.
- entornos inferiores: todos los entornos de desarrollo de una aplicación, como los que se utilizan para las compilaciones y pruebas iniciales.
- entorno de producción: instancia de una aplicación en ejecución a la que pueden acceder los usuarios finales. En una canalización de CI/CD, el entorno de producción es el último entorno de implementación.
- entornos superiores: todos los entornos a los que pueden acceder usuarios que no sean del equipo de desarrollo principal. Esto puede incluir un entorno de producción, entornos de preproducción y entornos para las pruebas de aceptación por parte de los usuarios.

epopeya

En las metodologías ágiles, son categorías funcionales que ayudan a organizar y priorizar el trabajo. Las epopeyas brindan una descripción detallada de los requisitos y las tareas de implementación. Por ejemplo, las epopeyas AWS de seguridad de CAF incluyen la gestión de identidades y accesos, los controles de detección, la seguridad de la infraestructura, la protección

de datos y la respuesta a incidentes. Para obtener más información sobre las epopeyas en la estrategia de migración de AWS , consulte la [Guía de implementación del programa](#).

PERP

Consulte [planificación de recursos empresariales](#).

análisis de datos de tipo exploratorio (EDA)

El proceso de analizar un conjunto de datos para comprender sus características principales. Se recopilan o agregan datos y, a continuación, se realizan las investigaciones iniciales para encontrar patrones, detectar anomalías y comprobar las suposiciones. El EDA se realiza mediante el cálculo de estadísticas resumidas y la creación de visualizaciones de datos.

F

tabla de datos

La tabla central de un [esquema en forma de estrella](#). Almacena datos cuantitativos sobre las operaciones comerciales. Normalmente, una tabla de hechos contiene dos tipos de columnas: las que contienen medidas y las que contienen una clave externa para una tabla de dimensiones.

fallan rápidamente

Una filosofía que utiliza pruebas frecuentes e incrementales para reducir el ciclo de vida del desarrollo. Es una parte fundamental de un enfoque ágil.

límite de aislamiento de fallas

En el Nube de AWS, un límite, como una zona de disponibilidad Región de AWS, un plano de control o un plano de datos, que limita el efecto de una falla y ayuda a mejorar la resiliencia de las cargas de trabajo. Para obtener más información, consulte [Límites de AWS aislamiento](#) de errores.

rama de característica

Consulte la [sucursal](#).

características

Los datos de entrada que se utilizan para hacer una predicción. Por ejemplo, en un contexto de fabricación, las características pueden ser imágenes que se capturan periódicamente desde la línea de fabricación.

importancia de las características

La importancia que tiene una característica para las predicciones de un modelo. Por lo general, esto se expresa como una puntuación numérica que se puede calcular mediante diversas técnicas, como las explicaciones aditivas de Shapley (SHAP) y los gradientes integrados. Para obtener más información, consulte [Interpretabilidad del modelo de aprendizaje automático con:AWS](#).

transformación de funciones

Optimizar los datos para el proceso de ML, lo que incluye enriquecer los datos con fuentes adicionales, escalar los valores o extraer varios conjuntos de información de un solo campo de datos. Esto permite que el modelo de ML se beneficie de los datos. Por ejemplo, si divide la fecha del “27 de mayo de 2021 00:15:37” en “jueves”, “mayo”, “2021” y “15”, puede ayudar al algoritmo de aprendizaje a aprender patrones matizados asociados a los diferentes componentes de los datos.

FGAC

Consulte el control [de acceso detallado](#).

control de acceso preciso (FGAC)

El uso de varias condiciones que tienen por objetivo permitir o denegar una solicitud de acceso.

migración relámpago

Método de migración de bases de datos que utiliza la replicación continua de datos mediante la [captura de datos modificados](#) para migrar los datos en el menor tiempo posible, en lugar de utilizar un enfoque gradual. El objetivo es reducir al mínimo el tiempo de inactividad.

G

bloqueo geográfico

Consulta [las restricciones geográficas](#).

restricciones geográficas (bloqueo geográfico)

En Amazon CloudFront, una opción para impedir que los usuarios de países específicos accedan a las distribuciones de contenido. Puede utilizar una lista de permitidos o bloqueados para especificar los países aprobados y prohibidos. Para obtener más información, consulta [Restringir la distribución geográfica del contenido](#) en la CloudFront documentación.

Flujo de trabajo de Gitflow

Un enfoque en el que los entornos inferiores y superiores utilizan diferentes ramas en un repositorio de código fuente. El flujo de trabajo de Gitflow se considera heredado, y el [flujo de trabajo basado en enlaces troncales](#) es el enfoque moderno preferido.

estrategia de implementación desde cero

La ausencia de infraestructura existente en un entorno nuevo. Al adoptar una estrategia de implementación desde cero para una arquitectura de sistemas, puede seleccionar todas las tecnologías nuevas sin que estas deban ser compatibles con una infraestructura existente, lo que también se conoce como [implementación sobre infraestructura existente](#). Si está ampliando la infraestructura existente, puede combinar las estrategias de implementación sobre infraestructuras existentes y de implementación desde cero.

barrera de protección

Una regla de alto nivel que ayuda a regular los recursos, las políticas y la conformidad en todas las unidades organizativas (OU). Las barreras de protección preventivas aplican políticas para garantizar la alineación con los estándares de conformidad. Se implementan mediante políticas de control de servicios y límites de permisos de IAM. Las barreras de protección de detección detectan las vulneraciones de las políticas y los problemas de conformidad, y generan alertas para su corrección. Se implementan mediante Amazon AWS Config AWS Security Hub GuardDuty AWS Trusted Advisor, Amazon Inspector y AWS Lambda cheques personalizados.

H

JA

Consulte [alta disponibilidad](#).

migración heterogénea de bases de datos

Migración de la base de datos de origen a una base de datos de destino que utilice un motor de base de datos diferente (por ejemplo, de Oracle a Amazon Aurora). La migración heterogénea suele ser parte de un esfuerzo de rediseño de la arquitectura y convertir el esquema puede ser una tarea compleja. [AWS ofrece AWS SCT](#), lo cual ayuda con las conversiones de esquemas.

alta disponibilidad (HA)

La capacidad de una carga de trabajo para funcionar de forma continua, sin intervención, en caso de desafíos o desastres. Los sistemas de alta disponibilidad están diseñados para realizar una

conmutación por error automática, ofrecer un rendimiento de alta calidad de forma constante y gestionar diferentes cargas y fallos con un impacto mínimo en el rendimiento.

modernización histórica

Un enfoque utilizado para modernizar y actualizar los sistemas de tecnología operativa (TO) a fin de satisfacer mejor las necesidades de la industria manufacturera. Un histórico es un tipo de base de datos que se utiliza para recopilar y almacenar datos de diversas fuentes en una fábrica.

migración homogénea de bases de datos

Migración de la base de datos de origen a una base de datos de destino que comparte el mismo motor de base de datos (por ejemplo, Microsoft SQL Server a Amazon RDS para SQL Server). La migración homogénea suele formar parte de un esfuerzo para volver a alojar o redefinir la plataforma. Puede utilizar las utilidades de bases de datos nativas para migrar el esquema.

datos recientes

Datos a los que se accede con frecuencia, como datos en tiempo real o datos traslacionales recientes. Por lo general, estos datos requieren un nivel o una clase de almacenamiento de alto rendimiento para proporcionar respuestas rápidas a las consultas.

hotfix

Una solución urgente para un problema crítico en un entorno de producción. Debido a su urgencia, las revisiones suelen realizarse fuera del flujo de trabajo habitual de las DevOps versiones.

periodo de hiperatención

Periodo, inmediatamente después de la transición, durante el cual un equipo de migración administra y monitorea las aplicaciones migradas en la nube para solucionar cualquier problema. Por lo general, este periodo dura de 1 a 4 días. Al final del periodo de hiperatención, el equipo de migración suele transferir la responsabilidad de las aplicaciones al equipo de operaciones en la nube.

I

laC

Vea [la infraestructura como código](#).

I

políticas basadas en identidad

Política asociada a uno o más directores de IAM que define sus permisos en el Nube de AWS entorno.

aplicación inactiva

Aplicación que utiliza un promedio de CPU y memoria de entre 5 y 20 por ciento durante un periodo de 90 días. En un proyecto de migración, es habitual retirar estas aplicaciones o mantenerlas en las instalaciones.

IIoT

Consulte [Internet de las cosas industrial](#).

infraestructura inmutable

Un modelo que implementa una nueva infraestructura para las cargas de trabajo de producción en lugar de actualizar, parchear o modificar la infraestructura existente. [Las infraestructuras inmutables son intrínsecamente más consistentes, fiables y predecibles que las infraestructuras mutables](#). Para obtener más información, consulte las prácticas recomendadas para [implementar con una infraestructura inmutable](#) en Well-Architected Framework AWS .

VPC entrante (de entrada)

En una arquitectura de AWS cuentas múltiples, una VPC que acepta, inspecciona y enruta las conexiones de red desde fuera de una aplicación. La [Arquitectura de referencia de seguridad de AWS](#) recomienda configurar su cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

migración gradual

Estrategia de transición en la que se migra la aplicación en partes pequeñas en lugar de realizar una transición única y completa. Por ejemplo, puede trasladar inicialmente solo unos pocos microservicios o usuarios al nuevo sistema. Tras comprobar que todo funciona correctamente, puede trasladar microservicios o usuarios adicionales de forma gradual hasta que pueda retirar su sistema heredado. Esta estrategia reduce los riesgos asociados a las grandes migraciones.

Industria 4.0

Un término que [Klaus Schwab](#) introdujo en 2016 para referirse a la modernización de los procesos de fabricación mediante avances en la conectividad, los datos en tiempo real, la automatización, el análisis y la inteligencia artificial/aprendizaje automático.

infraestructura

Todos los recursos y activos que se encuentran en el entorno de una aplicación.

infraestructura como código (IaC)

Proceso de aprovisionamiento y administración de la infraestructura de una aplicación mediante un conjunto de archivos de configuración. La IaC se ha diseñado para ayudarlo a centralizar la administración de la infraestructura, estandarizar los recursos y escalar con rapidez a fin de que los entornos nuevos sean repetibles, fiables y consistentes.

Internet de las cosas industrial (IIoT)

El uso de sensores y dispositivos conectados a Internet en los sectores industriales, como el productivo, el eléctrico, el automotriz, el sanitario, el de las ciencias de la vida y el de la agricultura. Para obtener más información, consulte [Creación de una estrategia de transformación digital del Internet de las cosas industrial \(IIoT\)](#).

VPC de inspección

En una arquitectura de AWS cuentas múltiples, una VPC centralizada que gestiona las inspecciones del tráfico de red entre las VPC (iguales o Regiones de AWS diferentes), Internet y las redes locales. La [Arquitectura de referencia de seguridad de AWS](#) recomienda configurar su cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

Internet de las cosas (IoT)

Red de objetos físicos conectados con sensores o procesadores integrados que se comunican con otros dispositivos y sistemas a través de Internet o de una red de comunicación local. Para obtener más información, consulte [¿Qué es IoT?](#).

interpretabilidad

Característica de un modelo de machine learning que describe el grado en que un ser humano puede entender cómo las predicciones del modelo dependen de sus entradas. Para más información, consulte [Interpretabilidad del modelo de machine learning con AWS](#).

IoT

[Consulte Internet de las cosas.](#)

biblioteca de información de TI (ITIL)

Conjunto de prácticas recomendadas para ofrecer servicios de TI y alinearlos con los requisitos empresariales. La ITIL proporciona la base para la ITSM.

administración de servicios de TI (ITSM)

Actividades asociadas con el diseño, la implementación, la administración y el soporte de los servicios de TI para una organización. Para obtener información sobre la integración de las operaciones en la nube con las herramientas de ITSM, consulte la [Guía de integración de operaciones](#).

ITIL

Consulte la [biblioteca de información de TI](#).

ITSM

Consulte [Administración de servicios de TI](#).

L

control de acceso basado en etiquetas (LBAC)

Una implementación del control de acceso obligatorio (MAC) en la que a los usuarios y a los propios datos se les asigna explícitamente un valor de etiqueta de seguridad. La intersección entre la etiqueta de seguridad del usuario y la etiqueta de seguridad de los datos determina qué filas y columnas puede ver el usuario.

zona de aterrizaje

Una landing zone es un AWS entorno multicuenta bien diseñado, escalable y seguro. Este es un punto de partida desde el cual las empresas pueden lanzar e implementar rápidamente cargas de trabajo y aplicaciones con confianza en su entorno de seguridad e infraestructura. Para obtener más información sobre las zonas de aterrizaje, consulte [Configuración de un entorno de AWS seguro y escalable con varias cuentas](#).

migración grande

Migración de 300 servidores o más.

LBAC

Consulte control de [acceso basado en etiquetas](#).

privilegio mínimo

La práctica recomendada de seguridad que consiste en conceder los permisos mínimos necesarios para realizar una tarea. Para obtener más información, consulte [Aplicar permisos de privilegio mínimo](#) en la documentación de IAM.

migrar mediante lift-and-shift

Ver [7 Rs](#).

sistema little-endian

Un sistema que almacena primero el byte menos significativo. Véase también [endianness](#).

entornos inferiores

[Véase entorno](#).

M

machine learning (ML)

Un tipo de inteligencia artificial que utiliza algoritmos y técnicas para el reconocimiento y el aprendizaje de patrones. El ML analiza y aprende de los datos registrados, como los datos del Internet de las cosas (IoT), para generar un modelo estadístico basado en patrones. Para más información, consulte [Machine learning](#).

rama principal

Ver [sucursal](#).

malware

Software diseñado para comprometer la seguridad o la privacidad de la computadora. El malware puede interrumpir los sistemas informáticos, filtrar información confidencial u obtener acceso no autorizado. Algunos ejemplos de malware son los virus, los gusanos, el ransomware, los troyanos, el spyware y los keyloggers.

servicios gestionados

Servicios de AWS para los que AWS opera la capa de infraestructura, el sistema operativo y las plataformas, y usted accede a los puntos finales para almacenar y recuperar datos. Amazon Simple Storage Service (Amazon S3) y Amazon DynamoDB son ejemplos de servicios gestionados. También se conocen como servicios abstractos.

sistema de ejecución de fabricación (MES)

Un sistema de software para rastrear, monitorear, documentar y controlar los procesos de producción que convierten las materias primas en productos terminados en el taller.

MAP

Consulte [Migration Acceleration Program](#).

mecanismo

Un proceso completo en el que se crea una herramienta, se impulsa su adopción y, a continuación, se inspeccionan los resultados para realizar ajustes. Un mecanismo es un ciclo que se refuerza y mejora a sí mismo a medida que funciona. Para obtener más información, consulte [Creación de mecanismos](#) en el AWS Well-Architected Framework.

cuenta de miembro

Todas las Cuentas de AWS demás cuentas, excepto la de administración, que forman parte de una organización. AWS Organizations Una cuenta no puede pertenecer a más de una organización a la vez.

MES

Consulte el [sistema de ejecución de la fabricación](#).

Transporte telemétrico de Message Queue Queue (MQTT)

[Un protocolo de comunicación ligero machine-to-machine \(M2M\), basado en el patrón de publicación/suscripción, para dispositivos de IoT con recursos limitados.](#)

microservicio

Un servicio pequeño e independiente que se comunica a través de API bien definidas y que, por lo general, es propiedad de equipos pequeños e independientes. Por ejemplo, un sistema de seguros puede incluir microservicios que se adapten a las capacidades empresariales, como las de ventas o marketing, o a subdominios, como las de compras, reclamaciones o análisis. Los beneficios de los microservicios incluyen la agilidad, la escalabilidad flexible, la facilidad de implementación, el código reutilizable y la resiliencia. Para obtener más información, consulte [Integrar](#) microservicios mediante servicios sin servidor. AWS

arquitectura de microservicios

Un enfoque para crear una aplicación con componentes independientes que ejecutan cada proceso de la aplicación como un microservicio. Estos microservicios se comunican a través de

una interfaz bien definida mediante API ligeras. Cada microservicio de esta arquitectura se puede actualizar, implementar y escalar para satisfacer la demanda de funciones específicas de una aplicación. Para obtener más información, consulte [Implementación de microservicios](#) en AWS

Programa de aceleración de la migración (MAP)

Un AWS programa que proporciona soporte de consultoría, formación y servicios para ayudar a las organizaciones a crear una base operativa sólida para migrar a la nube y para ayudar a compensar el costo inicial de las migraciones. El MAP incluye una metodología de migración para ejecutar las migraciones antiguas de forma metódica y un conjunto de herramientas para automatizar y acelerar los escenarios de migración más comunes.

migración a escala

Proceso de transferencia de la mayoría de la cartera de aplicaciones a la nube en oleadas, con más aplicaciones desplazadas a un ritmo más rápido en cada oleada. En esta fase, se utilizan las prácticas recomendadas y las lecciones aprendidas en las fases anteriores para implementar una fábrica de migración de equipos, herramientas y procesos con el fin de agilizar la migración de las cargas de trabajo mediante la automatización y la entrega ágil. Esta es la tercera fase de la [estrategia de migración de AWS](#).

fábrica de migración

Equipos multifuncionales que agilizan la migración de las cargas de trabajo mediante enfoques automatizados y ágiles. Los equipos de las fábricas de migración suelen incluir a analistas y propietarios de operaciones, empresas, ingenieros de migración, desarrolladores y DevOps profesionales que trabajan a pasos agigantados. Entre el 20 y el 50 por ciento de la cartera de aplicaciones empresariales se compone de patrones repetidos que pueden optimizarse mediante un enfoque de fábrica. Para obtener más información, consulte la [discusión sobre las fábricas de migración](#) y la [Guía de fábricas de migración a la nube](#) en este contenido.

metadatos de migración

Información sobre la aplicación y el servidor que se necesita para completar la migración. Cada patrón de migración requiere un conjunto diferente de metadatos de migración. Algunos ejemplos de metadatos de migración son la subred de destino, el grupo de seguridad y AWS la cuenta.

patrón de migración

Tarea de migración repetible que detalla la estrategia de migración, el destino de la migración y la aplicación o el servicio de migración utilizados. Ejemplo: rehospede la migración a Amazon EC2 AWS con Application Migration Service.

Migration Portfolio Assessment (MPA)

Una herramienta en línea que proporciona información para validar el modelo de negocio para la migración a la nube. AWS La MPA ofrece una evaluación detallada de la cartera (adecuación del tamaño de los servidores, precios, comparaciones del costo total de propiedad, análisis de los costos de migración), así como una planificación de la migración (análisis y recopilación de datos de aplicaciones, agrupación de aplicaciones, priorización de la migración y planificación de oleadas). La [herramienta MPA](#) (requiere iniciar sesión) está disponible de forma gratuita para todos los AWS consultores y consultores asociados de APN.

Evaluación de la preparación para la migración (MRA)

Proceso que consiste en obtener información sobre el estado de preparación de una organización para la nube, identificar sus puntos fuertes y débiles y elaborar un plan de acción para cerrar las brechas identificadas mediante el AWS CAF. Para obtener más información, consulte la [Guía de preparación para la migración](#). La MRA es la primera fase de la [estrategia de migración de AWS](#).

estrategia de migración

El enfoque utilizado para migrar una carga de trabajo a la AWS nube. Para obtener más información, consulte la entrada de las [7 R](#) de este glosario y consulte [Movilice a su organización para acelerar las migraciones a gran escala](#).

ML

[Consulte el aprendizaje automático.](#)

modernización

Transformar una aplicación obsoleta (antigua o monolítica) y su infraestructura en un sistema ágil, elástico y de alta disponibilidad en la nube para reducir los gastos, aumentar la eficiencia y aprovechar las innovaciones. Para obtener más información, consulte [Estrategia para modernizar las aplicaciones en el Nube de AWS](#).

evaluación de la preparación para la modernización

Evaluación que ayuda a determinar la preparación para la modernización de las aplicaciones de una organización; identifica los beneficios, los riesgos y las dependencias; y determina qué tan bien la organización puede soportar el estado futuro de esas aplicaciones. El resultado de la evaluación es un esquema de la arquitectura objetivo, una hoja de ruta que detalla las fases de desarrollo y los hitos del proceso de modernización y un plan de acción para abordar las brechas identificadas. Para obtener más información, consulte [Evaluación de la preparación para la modernización de las aplicaciones en la nube de AWS](#).

aplicaciones monolíticas (monolitos)

Aplicaciones que se ejecutan como un único servicio con procesos estrechamente acoplados. Las aplicaciones monolíticas presentan varios inconvenientes. Si una característica de la aplicación experimenta un aumento en la demanda, se debe escalar toda la arquitectura. Agregar o mejorar las características de una aplicación monolítica también se vuelve más complejo a medida que crece la base de código. Para solucionar problemas con la aplicación, puede utilizar una arquitectura de microservicios. Para obtener más información, consulte [Descomposición de monolitos en microservicios](#).

MAPA

Consulte [la evaluación de la cartera de migración](#).

MQTT

Consulte [Message Queue Queue Telemetría](#) y Transporte.

clasificación multiclase

Un proceso que ayuda a generar predicciones para varias clases (predice uno de más de dos resultados). Por ejemplo, un modelo de ML podría preguntar “¿Este producto es un libro, un automóvil o un teléfono?” o “¿Qué categoría de productos es más interesante para este cliente?”.

infraestructura mutable

Un modelo que actualiza y modifica la infraestructura existente para las cargas de trabajo de producción. Para mejorar la coherencia, la fiabilidad y la previsibilidad, el AWS Well-Architected Framework recomienda el uso [de una infraestructura inmutable](#) como práctica recomendada.

O

OAC

[Consulte el control de acceso de origen](#).

OAI

Consulte la [identidad de acceso de origen](#).

OCM

Consulte [gestión del cambio organizacional](#).

migración fuera de línea

Método de migración en el que la carga de trabajo de origen se elimina durante el proceso de migración. Este método implica un tiempo de inactividad prolongado y, por lo general, se utiliza para cargas de trabajo pequeñas y no críticas.

OI

Consulte [integración de operaciones](#).

OLA

Véase el [acuerdo a nivel operativo](#).

migración en línea

Método de migración en el que la carga de trabajo de origen se copia al sistema de destino sin que se desconecte. Las aplicaciones que están conectadas a la carga de trabajo pueden seguir funcionando durante la migración. Este método implica un tiempo de inactividad nulo o mínimo y, por lo general, se utiliza para cargas de trabajo de producción críticas.

OPC-UA

Consulte [Open Process Communications: arquitectura unificada](#).

Comunicaciones de proceso abierto: arquitectura unificada (OPC-UA)

Un protocolo de comunicación machine-to-machine (M2M) para la automatización industrial. El OPC-UA proporciona un estándar de interoperabilidad con esquemas de cifrado, autenticación y autorización de datos.

acuerdo de nivel operativo (OLA)

Acuerdo que aclara lo que los grupos de TI operativos se comprometen a ofrecerse entre sí, para respaldar un acuerdo de nivel de servicio (SLA).

revisión de la preparación operativa (ORR)

Una lista de preguntas y las mejores prácticas asociadas que le ayudan a comprender, evaluar, prevenir o reducir el alcance de los incidentes y posibles fallos. Para obtener más información, consulte [Operational Readiness Reviews \(ORR\)](#) en AWS Well-Architected Framework.

tecnología operativa (OT)

Sistemas de hardware y software que funcionan con el entorno físico para controlar las operaciones, los equipos y la infraestructura industriales. En la industria manufacturera, la

integración de los sistemas de TO y tecnología de la información (TI) es un enfoque clave para las transformaciones de [la industria 4.0](#).

integración de operaciones (OI)

Proceso de modernización de las operaciones en la nube, que implica la planificación de la preparación, la automatización y la integración. Para obtener más información, consulte la [Guía de integración de las operaciones](#).

registro de seguimiento organizativo

Un registro creado por el AWS CloudTrail que se registran todos los eventos para todos Cuentas de AWS los miembros de una organización AWS Organizations. Este registro de seguimiento se crea en cada Cuenta de AWS que forma parte de la organización y realiza un seguimiento de la actividad en cada cuenta. Para obtener más información, consulte [Crear un registro para una organización](#) en la CloudTrail documentación.

administración del cambio organizacional (OCM)

Marco para administrar las transformaciones empresariales importantes y disruptivas desde la perspectiva de las personas, la cultura y el liderazgo. La OCM ayuda a las empresas a prepararse para nuevos sistemas y estrategias y a realizar la transición a ellos, al acelerar la adopción de cambios, abordar los problemas de transición e impulsar cambios culturales y organizacionales. En la estrategia de AWS migración, este marco se denomina aceleración del personal, debido a la velocidad de cambio que requieren los proyectos de adopción de la nube. Para obtener más información, consulte la [Guía de OCM](#).

control de acceso de origen (OAC)

En CloudFront, una opción mejorada para restringir el acceso y proteger el contenido del Amazon Simple Storage Service (Amazon S3). El OAC admite todos los buckets de S3 Regiones de AWS, el cifrado del lado del servidor AWS KMS (SSE-KMS) y las solicitudes dinámicas PUT y DELETE dirigidas al bucket de S3.

identidad de acceso de origen (OAI)

En CloudFront, una opción para restringir el acceso y proteger el contenido de Amazon S3. Cuando utiliza OAI, CloudFront crea un principal con el que Amazon S3 puede autenticarse. Los directores autenticados solo pueden acceder al contenido de un bucket de S3 a través de una distribución específica. CloudFront Consulte también el [OAC](#), que proporciona un control de acceso más detallado y mejorado.

O

Consulte la [revisión de la preparación operativa](#).

NO

Consulte [tecnología operativa](#).

VPC saliente (de salida)

En una arquitectura de AWS cuentas múltiples, una VPC que gestiona las conexiones de red que se inician desde una aplicación. La [Arquitectura de referencia de seguridad de AWS](#) recomienda configurar su cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

P

límite de permisos

Una política de administración de IAM que se adjunta a las entidades principales de IAM para establecer los permisos máximos que puede tener el usuario o el rol. Para obtener más información, consulte [Límites de permisos](#) en la documentación de IAM.

información de identificación personal (PII)

Información que, vista directamente o combinada con otros datos relacionados, puede utilizarse para deducir de manera razonable la identidad de una persona. Algunos ejemplos de información de identificación personal son los nombres, las direcciones y la información de contacto.

PII

Consulte la información de [identificación personal](#).

manual de estrategias

Conjunto de pasos predefinidos que capturan el trabajo asociado a las migraciones, como la entrega de las funciones de operaciones principales en la nube. Un manual puede adoptar la forma de scripts, manuales de procedimientos automatizados o resúmenes de los procesos o pasos necesarios para operar un entorno modernizado.

PLC

Consulte [controlador lógico programable](#).

PLM

Consulte la [gestión del ciclo de vida del producto](#).

política

Un objeto que puede definir los permisos (consulte la [política basada en la identidad](#)), especifique las condiciones de acceso (consulte la [política basada en los recursos](#)) o defina los permisos máximos para todas las cuentas de una organización AWS Organizations (consulte la política de control de [servicios](#)).

persistencia políglota

Elegir de forma independiente la tecnología de almacenamiento de datos de un microservicio en función de los patrones de acceso a los datos y otros requisitos. Si sus microservicios tienen la misma tecnología de almacenamiento de datos, pueden enfrentarse a desafíos de implementación o experimentar un rendimiento deficiente. Los microservicios se implementan más fácilmente y logran un mejor rendimiento y escalabilidad si utilizan el almacén de datos que mejor se adapte a sus necesidades. Para obtener más información, consulte [Habilitación de la persistencia de datos en los microservicios](#).

evaluación de cartera

Proceso de detección, análisis y priorización de la cartera de aplicaciones para planificar la migración. Para obtener más información, consulte la [Evaluación de la preparación para la migración](#).

predicate

Una condición de consulta que devuelve true o false, por lo general, se encuentra en una cláusula. WHERE

pulsar un predicado

Técnica de optimización de consultas de bases de datos que filtra los datos de la consulta antes de transferirlos. Esto reduce la cantidad de datos que se deben recuperar y procesar de la base de datos relacional y mejora el rendimiento de las consultas.

control preventivo

Un control de seguridad diseñado para evitar que ocurra un evento. Estos controles son la primera línea de defensa para evitar el acceso no autorizado o los cambios no deseados en la red. Para obtener más información, consulte [Controles preventivos](#) en Implementación de controles de seguridad en AWS.

entidad principal

Una entidad AWS que puede realizar acciones y acceder a los recursos. Esta entidad suele ser un usuario raíz para un Cuenta de AWS rol de IAM o un usuario. Para obtener más información, consulte Entidad principal en [Términos y conceptos de roles](#) en la documentación de IAM.

Privacidad desde el diseño

Un enfoque de ingeniería de sistemas que tiene en cuenta la privacidad durante todo el proceso de ingeniería.

zonas alojadas privadas

Contenedor que aloja información acerca de cómo desea que responda Amazon Route 53 a las consultas de DNS de un dominio y sus subdominios en una o varias VPC. Para obtener más información, consulte [Uso de zonas alojadas privadas](#) en la documentación de Route 53.

control proactivo

Un [control de seguridad](#) diseñado para evitar el despliegue de recursos no conformes. Estos controles escanean los recursos antes de aprovisionarlos. Si el recurso no cumple con el control, significa que no está aprovisionado. Para obtener más información, consulte la [guía de referencia de controles](#) en la AWS Control Tower documentación y consulte [Controles proactivos](#) en Implementación de controles de seguridad en AWS.

gestión del ciclo de vida del producto (PLM)

La gestión de los datos y los procesos de un producto a lo largo de todo su ciclo de vida, desde el diseño, el desarrollo y el lanzamiento, pasando por el crecimiento y la madurez, hasta el rechazo y la retirada.

entorno de producción

Consulte [el entorno](#).

controlador lógico programable (PLC)

En la fabricación, una computadora adaptable y altamente confiable que monitorea las máquinas y automatiza los procesos de fabricación.

seudonimización

El proceso de reemplazar los identificadores personales de un conjunto de datos por valores de marcadores de posición. La seudonimización puede ayudar a proteger la privacidad personal. Los datos seudonimizados siguen considerándose datos personales.

publicar/suscribirse (pub/sub)

Un patrón que permite las comunicaciones asíncronas entre microservicios para mejorar la escalabilidad y la capacidad de respuesta. Por ejemplo, en un [MES](#) basado en microservicios, un microservicio puede publicar mensajes de eventos en un canal al que se puedan suscribir otros microservicios. El sistema puede añadir nuevos microservicios sin cambiar el servicio de publicación.

Q

plan de consulta

Serie de pasos, como instrucciones, que se utilizan para acceder a los datos de un sistema de base de datos relacional SQL.

regresión del plan de consulta

El optimizador de servicios de la base de datos elige un plan menos óptimo que antes de un cambio determinado en el entorno de la base de datos. Los cambios en estadísticas, restricciones, configuración del entorno, enlaces de parámetros de consultas y actualizaciones del motor de base de datos PostgreSQL pueden provocar una regresión del plan.

R

Matriz RACI

Véase [responsable, responsable, consultado, informado \(RACI\)](#).

ransomware

Software malicioso que se ha diseñado para bloquear el acceso a un sistema informático o a los datos hasta que se efectúe un pago.

Matriz RASCI

Véase [responsable, responsable, consultado, informado \(RACI\)](#).

RCAC

Consulte control de [acceso por filas y columnas](#).

read replica

Una copia de una base de datos que se utiliza con fines de solo lectura. Puede enrutar las consultas a la réplica de lectura para reducir la carga en la base de datos principal.

rediseñar

Ver [7 Rs.](#)

objetivo de punto de recuperación (RPO)

La cantidad de tiempo máximo aceptable desde el último punto de recuperación de datos. Esto determina qué se considera una pérdida de datos aceptable entre el último punto de recuperación y la interrupción del servicio.

objetivo de tiempo de recuperación (RTO)

La demora máxima aceptable entre la interrupción del servicio y el restablecimiento del servicio.

refactorizar

Ver [7 Rs.](#)

Región

Una colección de AWS recursos en un área geográfica. Cada uno Región de AWS está aislado y es independiente de los demás para proporcionar tolerancia a las fallas, estabilidad y resiliencia. Para obtener más información, consulte [Regiones de AWS Especificar qué cuenta puede usar.](#)

regresión

Una técnica de ML que predice un valor numérico. Por ejemplo, para resolver el problema de “¿A qué precio se venderá esta casa?”, un modelo de ML podría utilizar un modelo de regresión lineal para predecir el precio de venta de una vivienda en función de datos conocidos sobre ella (por ejemplo, los metros cuadrados).

volver a alojar

Consulte [7 Rs.](#)

versión

En un proceso de implementación, el acto de promover cambios en un entorno de producción.

trasladarse

Ver [7 Rs.](#)

redefinir la plataforma

Ver [7 Rs](#).

recompra

Ver [7 Rs](#).

resiliencia

La capacidad de una aplicación para resistir las interrupciones o recuperarse de ellas. [La alta disponibilidad](#) y la [recuperación ante desastres](#) son consideraciones comunes a la hora de planificar la resiliencia en el. Nube de AWS Para obtener más información, consulte [Nube de AWS Resiliencia](#).

política basada en recursos

Una política asociada a un recurso, como un bucket de Amazon S3, un punto de conexión o una clave de cifrado. Este tipo de política especifica a qué entidades principales se les permite el acceso, las acciones compatibles y cualquier otra condición que deba cumplirse.

matriz responsable, confiable, consultada e informada (RACI)

Una matriz que define las funciones y responsabilidades de todas las partes involucradas en las actividades de migración y las operaciones de la nube. El nombre de la matriz se deriva de los tipos de responsabilidad definidos en la matriz: responsable (R), contable (A), consultado (C) e informado (I). El tipo de soporte (S) es opcional. Si incluye el soporte, la matriz se denomina matriz RASCI y, si la excluye, se denomina matriz RACI.

control receptivo

Un control de seguridad que se ha diseñado para corregir los eventos adversos o las desviaciones con respecto a su base de seguridad. Para obtener más información, consulte [Controles receptivos](#) en Implementación de controles de seguridad en AWS.

retain

Consulte [7 Rs](#).

jubilarse

Ver [7 Rs](#).

rotación

Proceso de actualizar periódicamente un [secreto](#) para dificultar el acceso de un atacante a las credenciales.

control de acceso por filas y columnas (RCAC)

El uso de expresiones SQL básicas y flexibles que tienen reglas de acceso definidas. El RCAC consta de permisos de fila y máscaras de columnas.

RPO

Consulte el [objetivo del punto de recuperación](#).

RTO

Consulte el [objetivo de tiempo de recuperación](#).

manual de procedimientos

Conjunto de procedimientos manuales o automatizados necesarios para realizar una tarea específica. Por lo general, se diseñan para agilizar las operaciones o los procedimientos repetitivos con altas tasas de error.

S

SAML 2.0

Un estándar abierto que utilizan muchos proveedores de identidad (IdPs). Esta función permite el inicio de sesión único (SSO) federado, de modo que los usuarios pueden iniciar sesión AWS Management Console o llamar a las operaciones de la AWS API sin tener que crear un usuario en IAM para todos los miembros de la organización. Para obtener más información sobre la federación basada en SAML 2.0, consulte [Acerca de la federación basada en SAML 2.0](#) en la documentación de IAM.

SCADA

Consulte el [control de supervisión y la adquisición de datos](#).

SCP

Consulte la [política de control de servicios](#).

secreta

Información confidencial o restringida, como una contraseña o credenciales de usuario, que almacene de forma cifrada. AWS Secrets Manager Se compone del valor secreto y sus

metadatos. El valor secreto puede ser binario, una sola cadena o varias cadenas. Para obtener más información, consulte la documentación de [Secret](#) in the Secrets Manager.

control de seguridad

Barrera de protección técnica o administrativa que impide, detecta o reduce la capacidad de un agente de amenazas para aprovechar una vulnerabilidad de seguridad. Existen cuatro tipos principales de controles de seguridad: [preventivos](#), [de detección](#), con [capacidad](#) de [respuesta](#) y [proactivos](#).

refuerzo de la seguridad

Proceso de reducir la superficie expuesta a ataques para hacerla más resistente a los ataques. Esto puede incluir acciones, como la eliminación de los recursos que ya no se necesitan, la implementación de prácticas recomendadas de seguridad consistente en conceder privilegios mínimos o la desactivación de características innecesarias en los archivos de configuración.

sistema de información sobre seguridad y administración de eventos (SIEM)

Herramientas y servicios que combinan sistemas de administración de información sobre seguridad (SIM) y de administración de eventos de seguridad (SEM). Un sistema de SIEM recopila, monitorea y analiza los datos de servidores, redes, dispositivos y otras fuentes para detectar amenazas y brechas de seguridad y generar alertas.

automatización de la respuesta de seguridad

Una acción predefinida y programada que está diseñada para responder automáticamente a un evento de seguridad o remediarlo. Estas automatizaciones sirven como controles de seguridad [detectables](#) o [adaptables](#) que le ayudan a implementar las mejores prácticas AWS de seguridad. Algunos ejemplos de acciones de respuesta automatizadas incluyen la modificación de un grupo de seguridad de VPC, la aplicación de parches a una instancia de Amazon EC2 o la rotación de credenciales.

cifrado del servidor

Cifrado de los datos en su destino, por parte de quien Servicio de AWS los recibe.

política de control de servicio (SCP)

Una política que proporciona un control centralizado de los permisos de todas las cuentas de una organización en AWS Organizations. Las SCP definen barreras de protección o establecen límites a las acciones que un administrador puede delegar en los usuarios o roles. Puede utilizar las SCP como listas de permitidos o rechazados, para especificar qué servicios o acciones se

encuentra permitidos o prohibidos. Para obtener más información, consulte [las políticas de control de servicios](#) en la AWS Organizations documentación.

punto de enlace de servicio

La URL del punto de entrada de un Servicio de AWS. Para conectarse mediante programación a un servicio de destino, puede utilizar un punto de conexión. Para obtener más información, consulte [Puntos de conexión de Servicio de AWS](#) en Referencia general de AWS.

acuerdo de nivel de servicio (SLA)

Acuerdo que aclara lo que un equipo de TI se compromete a ofrecer a los clientes, como el tiempo de actividad y el rendimiento del servicio.

indicador de nivel de servicio (SLI)

Medición de un aspecto del rendimiento de un servicio, como la tasa de errores, la disponibilidad o el rendimiento.

objetivo de nivel de servicio (SLO)

[Una métrica objetivo que representa el estado de un servicio, medido mediante un indicador de nivel de servicio.](#)

modelo de responsabilidad compartida

Un modelo que describe la responsabilidad que compartes con respecto a la seguridad y AWS el cumplimiento de la nube. AWS es responsable de la seguridad de la nube, mientras que usted es responsable de la seguridad en la nube. Para obtener más información, consulte el [Modelo de responsabilidad compartida](#).

SIEM

Consulte [la información de seguridad y el sistema de gestión de eventos](#).

punto único de fallo (SPOF)

Una falla en un único componente crítico de una aplicación que puede interrumpir el sistema.

SLA

Consulte el acuerdo [de nivel de servicio](#).

SLI

Consulte el indicador de [nivel de servicio](#).

ASÍ QUE

Consulte el objetivo de [nivel de servicio](#).

split-and-seed modelo

Un patrón para escalar y acelerar los proyectos de modernización. A medida que se definen las nuevas funciones y los lanzamientos de los productos, el equipo principal se divide para crear nuevos equipos de productos. Esto ayuda a ampliar las capacidades y los servicios de su organización, mejora la productividad de los desarrolladores y apoya la innovación rápida. Para obtener más información, consulte [Enfoque gradual para modernizar las aplicaciones en el](#). Nube de AWS

SPOT

Consulte el [punto único de falla](#).

esquema en forma de estrella

Estructura organizativa de una base de datos que utiliza una tabla de datos grande para almacenar datos transaccionales o medidos y una o más tablas dimensionales más pequeñas para almacenar los atributos de los datos. Esta estructura está diseñada para usarse en un [almacén de datos](#) o con fines de inteligencia empresarial.

patrón de higo estrangulador

Un enfoque para modernizar los sistemas monolíticos mediante la reescritura y el reemplazo gradual de las funciones del sistema hasta que se pueda dismantelar el sistema heredado. Este patrón utiliza la analogía de una higuera que crece hasta convertirse en un árbol estable y, finalmente, se apodera y reemplaza a su host. El patrón fue [presentado por Martin Fowler](#) como una forma de gestionar el riesgo al reescribir sistemas monolíticos. Para ver un ejemplo con la aplicación de este patrón, consulte [Modernización gradual de los servicios web antiguos de Microsoft ASP.NET \(ASMX\) mediante contenedores y Amazon API Gateway](#).

subred

Un intervalo de direcciones IP en la VPC. Una subred debe residir en una sola zona de disponibilidad.

supervisión, control y adquisición de datos (SCADA)

En la industria manufacturera, un sistema que utiliza hardware y software para monitorear los activos físicos y las operaciones de producción.

cifrado simétrico

Un algoritmo de cifrado que utiliza la misma clave para cifrar y descifrar los datos.

pruebas sintéticas

Probar un sistema de manera que simule las interacciones de los usuarios para detectar posibles problemas o monitorear el rendimiento. Puede usar [Amazon CloudWatch Synthetics](#) para crear estas pruebas.

T

etiquetas

Pares clave-valor que actúan como metadatos para organizar los recursos. AWS Las etiquetas pueden ayudarle a administrar, identificar, organizar, buscar y filtrar recursos. Para obtener más información, consulte [Etiquetado de los recursos de AWS](#).

variable de destino

El valor que intenta predecir en el ML supervisado. Esto también se conoce como variable de resultado. Por ejemplo, en un entorno de fabricación, la variable objetivo podría ser un defecto del producto.

lista de tareas

Herramienta que se utiliza para hacer un seguimiento del progreso mediante un manual de procedimientos. La lista de tareas contiene una descripción general del manual de procedimientos y una lista de las tareas generales que deben completarse. Para cada tarea general, se incluye la cantidad estimada de tiempo necesario, el propietario y el progreso.

entorno de prueba

[Consulte entorno.](#)

entrenamiento

Proporcionar datos de los que pueda aprender su modelo de ML. Los datos de entrenamiento deben contener la respuesta correcta. El algoritmo de aprendizaje encuentra patrones en los datos de entrenamiento que asignan los atributos de los datos de entrada al destino (la respuesta que desea predecir). Genera un modelo de ML que captura estos patrones. Luego, el modelo de ML se puede utilizar para obtener predicciones sobre datos nuevos para los que no se conoce el destino.

puerta de enlace de tránsito

Centro de tránsito de red que puede utilizar para interconectar las VPC y las redes en las instalaciones. Para obtener más información, consulte [Qué es una pasarela de tránsito](#) en la AWS Transit Gateway documentación.

flujo de trabajo basado en enlaces troncales

Un enfoque en el que los desarrolladores crean y prueban características de forma local en una rama de característica y, a continuación, combinan esos cambios en la rama principal. Luego, la rama principal se adapta a los entornos de desarrollo, preproducción y producción, de forma secuencial.

acceso de confianza

Otorgar permisos a un servicio que especifique para realizar tareas en su organización AWS Organizations y en sus cuentas en su nombre. El servicio de confianza crea un rol vinculado al servicio en cada cuenta, cuando ese rol es necesario, para realizar las tareas de administración por usted. Para obtener más información, consulte [AWS Organizations Utilización con otros AWS servicios](#) en la AWS Organizations documentación.

ajuste

Cambiar aspectos de su proceso de formación a fin de mejorar la precisión del modelo de ML. Por ejemplo, puede entrenar el modelo de ML al generar un conjunto de etiquetas, incorporar etiquetas y, luego, repetir estos pasos varias veces con diferentes ajustes para optimizar el modelo.

equipo de dos pizzas

Un DevOps equipo pequeño al que puedes alimentar con dos pizzas. Un equipo formado por dos integrantes garantiza la mejor oportunidad posible de colaboración en el desarrollo de software.

U

incertidumbre

Un concepto que hace referencia a información imprecisa, incompleta o desconocida que puede socavar la fiabilidad de los modelos predictivos de ML. Hay dos tipos de incertidumbre: la incertidumbre epistémica se debe a datos limitados e incompletos, mientras que la incertidumbre aleatoria se debe al ruido y la aleatoriedad inherentes a los datos. Para más información, consulte la guía [Cuantificación de la incertidumbre en los sistemas de aprendizaje profundo](#).

tareas indiferenciadas

También conocido como tareas arduas, es el trabajo que es necesario para crear y operar una aplicación, pero que no proporciona un valor directo al usuario final ni proporciona una ventaja competitiva. Algunos ejemplos de tareas indiferenciadas son la adquisición, el mantenimiento y la planificación de la capacidad.

entornos superiores

Ver [entorno](#).

V

succión

Una operación de mantenimiento de bases de datos que implica limpiar después de las actualizaciones incrementales para recuperar espacio de almacenamiento y mejorar el rendimiento.

control de versión

Procesos y herramientas que realizan un seguimiento de los cambios, como los cambios en el código fuente de un repositorio.

Emparejamiento de VPC

Conexión entre dos VPC que permite enrutar el tráfico mediante direcciones IP privadas. Para obtener más información, consulte [¿Qué es una interconexión de VPC?](#) en la documentación de Amazon VPC.

vulnerabilidad

Defecto de software o hardware que pone en peligro la seguridad del sistema.

W

caché caliente

Un búfer caché que contiene datos actuales y relevantes a los que se accede con frecuencia. La instancia de base de datos puede leer desde la caché del búfer, lo que es más rápido que leer desde la memoria principal o el disco.

datos templados

Datos a los que el acceso es infrecuente. Al consultar este tipo de datos, normalmente se aceptan consultas moderadamente lentas.

función de ventana

Función SQL que realiza un cálculo en un grupo de filas que se relacionan de alguna manera con el registro actual. Las funciones de ventana son útiles para procesar tareas, como calcular una media móvil o acceder al valor de las filas en función de la posición relativa de la fila actual.

carga de trabajo

Conjunto de recursos y código que ofrece valor comercial, como una aplicación orientada al cliente o un proceso de backend.

flujo de trabajo

Grupos funcionales de un proyecto de migración que son responsables de un conjunto específico de tareas. Cada flujo de trabajo es independiente, pero respalda a los demás flujos de trabajo del proyecto. Por ejemplo, el flujo de trabajo de la cartera es responsable de priorizar las aplicaciones, planificar las oleadas y recopilar los metadatos de migración. El flujo de trabajo de la cartera entrega estos recursos al flujo de trabajo de migración, que luego migra los servidores y las aplicaciones.

GUSANO

Mira, [escribe una vez, lee muchas](#).

WQF

Consulte el [marco de calificación de cargas de trabajo de AWS](#).

escribe una vez, lee muchas (WORM)

Un modelo de almacenamiento que escribe los datos una sola vez y evita que los datos se eliminen o modifiquen. Los usuarios autorizados pueden leer los datos tantas veces como sea necesario, pero no pueden cambiarlos. Esta infraestructura de almacenamiento de datos se considera [inmutable](#).

Z

ataque de día cero

Un ataque, normalmente de malware, que aprovecha una vulnerabilidad de [día cero](#).

vulnerabilidad de día cero

Un defecto o una vulnerabilidad sin mitigación en un sistema de producción. Los agentes de amenazas pueden usar este tipo de vulnerabilidad para atacar el sistema. Los desarrolladores suelen darse cuenta de la vulnerabilidad a raíz del ataque.

aplicación zombi

Aplicación que utiliza un promedio de CPU y memoria menor al 5 por ciento. En un proyecto de migración, es habitual retirar estas aplicaciones.

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la versión original de inglés, prevalecerá la versión en inglés.