



Atividades de manutenção de bancos de dados PostgreSQL no Amazon RDS e no Amazon Aurora para evitar problemas de performance

## AWS Orientação prescritiva



# AWS Orientação prescritiva: Atividades de manutenção de bancos de dados PostgreSQL no Amazon RDS e no Amazon Aurora para evitar problemas de performance

# Table of Contents

Introdução .....	1
Resultados de negócios desejados .....	1
Controle de concorrência multiversão (MVCC) .....	1
Limpeza e análise automática de tabelas .....	3
Parâmetros relacionados à memória do autovacuum .....	5
Ajuste dos parâmetros de autovacuum .....	6
Nível de cluster ou instância .....	6
Nível de tabela .....	6
Uso de configurações agressivas de autovacuum no nível de tabela .....	6
Vantagens e limitações .....	7
Limpeza e análise manual de tabelas .....	9
Execução de operações de vacuum e limpeza em paralelo .....	10
Reescrita de toda uma tabela com VACUUM FULL .....	14
Remoção do inchaço com pg_repack .....	15
Recriação de índices .....	17
Criação de um novo índice .....	21
Recriação de um índice .....	21
Exemplos .....	22
Exemplo: recuperação de espaço usando autovacuum e VACUUM FULL .....	24
Recursos .....	29
Histórico do documento .....	30
Glossário .....	31
# .....	31
A .....	32
B .....	35
C .....	37
D .....	40
E .....	45
F .....	47
G .....	49
H .....	50
eu .....	51
L .....	54
M .....	55

---

O .....	59
P .....	62
Q .....	65
R .....	65
S .....	68
T .....	72
U .....	74
V .....	74
W .....	75
Z .....	76
.....	lxxvii

# Atividades de manutenção de bancos de dados PostgreSQL no Amazon RDS e no Amazon Aurora para evitar problemas de performance

Anuradha Chintha, Rajesh Madiwale e Srinivas Potlachervoo, Amazon Web Services (AWS)

Agosto de 2025 ([histórico do documento](#))

A edição do Amazon Aurora compatível com PostgreSQL e o Amazon Relational Database Service (Amazon RDS) para PostgreSQL são serviços de banco de dados relacionais totalmente gerenciados para bancos de dados PostgreSQL. Esses serviços gerenciados liberam o administrador do banco de dados de muitas tarefas de manutenção e gerenciamento. No entanto, algumas tarefas de manutenção, como VACUUM, exigem monitoramento e configuração rigorosos com base no uso do banco de dados. Este guia descreve as atividades de manutenção do PostgreSQL no Amazon RDS e no Aurora.

## Resultados de negócios desejados

A performance do banco de dados é uma medida fundamental que sustenta o sucesso de uma empresa. A realização de atividades de manutenção em seus bancos de dados do Aurora compatível com PostgreSQL e do Amazon RDS para PostgreSQL proporciona as seguintes vantagens:

- Ajuda a alcançar a performance ideal de consulta
- Libera espaço inchado para reutilização em transações futuras
- Evita o wraparound da transação
- Ajuda o otimizador a gerar bons planos
- Garante o uso adequado do índice

## Controle de concorrência multiversão (MVCC)

A manutenção do banco de dados PostgreSQL requer uma compreensão do controle de concorrência multiversão (MVCC), que é um mecanismo do PostgreSQL. Quando várias transações são processadas simultaneamente no banco de dados, o MVCC garante que a atomicidade e o

isolamento, que são duas características das transações de atomicidade, consistência, isolamento e durabilidade (ACID), sejam mantidos. No MVCC, cada operação de gravação gera uma nova versão dos dados e armazena a versão anterior. Leitores e gravadores não bloqueiam uns aos outros. Quando uma transação lê dados, o sistema escolhe uma das versões para fornecer isolamento da transação. O PostgreSQL e alguns bancos de dados relacionais usam uma adaptação do MVCC chamada isolamento de snapshot (SI). Por exemplo, o Oracle implementa o SI usando segmentos de reversão. Durante uma operação de gravação, o Oracle grava a versão antiga dos dados em um segmento de reversão e substitui a área de dados pela nova versão. Os bancos de dados PostgreSQL implementam o SI usando regras de verificação de visibilidade para avaliar as versões. Quando novos dados são colocados em uma página de tabela, o PostgreSQL usa essas regras para selecionar a versão apropriada dos dados para uma operação de leitura.

Quando você modifica dados em uma linha da tabela, o PostgreSQL usa o MVCC para manter várias versões da linha. Durante as operações UPDATE e DELETE na tabela, o banco de dados mantém as versões antigas das linhas para outras transações em execução que talvez precisem de uma visão consistente dos dados. Essas versões antigas são chamadas de linhas inativas (tuplas). Uma coleção de tuplas inativas produz inchaço. Um grande volume de inchaço no banco de dados pode causar vários problemas, incluindo geração deficiente de planos de consulta, performance lenta de consultas e maior uso de espaço em disco para armazenar as versões mais antigas.

A remoção do inchaço e a manutenção de um banco de dados íntegro requer manutenção periódica, que inclui essas atividades, que são abordadas nas seguintes seções:

- [Limpeza e análise automática de tabelas](#)
- [Limpeza e análise manual de tabelas](#)
- [Remoção do inchaço com pg\\_repack](#)
- [Recriação de índices](#)

## Limpeza e análise automática de tabelas

O Autovacuum é um daemon (ou seja, ele é executado em segundo plano) que aspira (limpa) automaticamente as tuplas inativas, recupera o armazenamento e reúne estatísticas. Ele verifica se há tabelas inchadas no banco de dados e limpa o inchaço para reutilizar o espaço. Ele monitora tabelas e índices do banco de dados e os adiciona a uma tarefa de vacuum após atingirem um limite específico de operações de atualização ou exclusão.

O Autovacuum gerencia a limpeza automatizando os comandos VACUUM e ANALYZE do PostgreSQL. O VACUUM remove o inchaço das tabelas e recupera o espaço, enquanto o ANALYZE atualiza as estatísticas que permitem que o otimizador gere planos eficientes. O VACUUM também executa uma tarefa importante chamada congelamento de vacuum para evitar problemas de wraparound de IDs de transação no banco de dados. Cada linha atualizada no banco de dados recebe um ID de transação do mecanismo de controle de transações do PostgreSQL. Eles IDs controlam a visibilidade da linha em relação a outras transações simultâneas. O ID de transação é um número de 32 bits. Dois bilhões IDs são sempre mantidos no passado visível. O restante (cerca de 2,2 bilhões) IDs é preservado para transações que ocorrerão no futuro e está oculto da transação atual. O PostgreSQL exige uma limpeza e um congelamento ocasionais de linhas antigas para evitar que as transações sejam agrupadas e tornem as linhas antigas e existentes invisíveis quando novas transações forem criadas. Para obter mais informações, consulte [Evitar falhas de conclusão do ID da transação](#) na documentação do PostgreSQL.

O autovacuum é recomendado e habilitado por padrão. Seus parâmetros incluem o seguinte:

Parâmetro	Descrição	Padrão para Amazon RDS	Padrão para Aurora
autovacuum_vacuum_threshold	O número mínimo de operações de atualização ou exclusão de tuplas que devem ocorrer em uma tabela antes que o autovacuum a limpe.	50 operações	50 operações

---

<code>autovacuum_analyze_threshold</code>	O número mínimo de inserções, atualizações ou exclusões de tuplas que devem ocorrer em uma tabela antes que o autovacuum a analise.	50 operações	50 operações
<code>autovacuum_scale_factor</code>	A porcentagem de tuplas que devem ser modificadas em uma tabela antes que o autovacuum a limpe.	0.1	0.1
<code>autovacuum_analyze_scale_factor</code>	A porcentagem de tuplas que devem ser modificadas em uma tabela antes que o autovacuum a analise.	0,05	0,05
<code>autovacuum_freeze_max_age</code>	A idade máxima de congelamento de IDs antes de uma mesa ser limpa para evitar problemas de encapsulamento do ID da transação.	200 milhões de transações	200 milhões de transações

O Autovacuum faz uma lista de tabelas para processar com base em fórmulas de limite específicas, conforme a seguir.

- Limite de execução do VACUUM em uma tabela:

```
vacuum threshold = autovacuum_vacuum_threshold + (autovacuum_vacuum_scale_factor *  
Total row count of table)
```

- Limite de execução do ANALYZE em uma tabela:

```
analyze threshold = autovacuum_analyze_threshold + (autovacuum_analyze_scale_factor *  
Total row count of table)
```

Para tabelas de pequeno a médio porte, os valores padrão podem ser suficientes. No entanto, uma tabela grande com modificações de dados frequentes terá um número maior de tuplas inativas. Nesse caso, o autovacuum pode processar a tabela com frequência para fins de manutenção, e a manutenção de outras mesas pode ser atrasada ou ignorada até que a tabela grande seja concluída. Para evitar isso, você pode ajustar os parâmetros de autovacuum descritos na seção a seguir.

## Parâmetros relacionados à memória do autovacuum

### **autovacuum\_max\_workers**

Especifica o número máximo de processos de autovacuum (exceto o inicializador de autovacuum) que podem ser executados ao mesmo tempo. Esse parâmetro só pode ser definido quando você inicia o servidor. Se o processo de autovacuum estiver ocupado com uma tabela grande, esse parâmetro ajudará a executar a limpeza de outras tabelas.

### **maintenance\_work\_mem**

Ele especifica a quantidade máxima de memória a ser utilizada por operações de manutenção, como VACUUM, CREATE INDEX e ALTER. No Amazon RDS e no Aurora, a memória é alocada com base na classe da instância usando a fórmula `GREATEST({DBInstanceClassMemory/63963136*1024}, 65536)`. Quando o autovacuum é executado, até `autovacuum_max_workers` vezes esse valor calculado pode ser alocado, portanto, tenha cuidado para não definir o valor muito alto. Para controlar isso, você pode definir `autovacuum_work_mem` separadamente.

### **autovacuum\_work\_mem**

Especifica a memória máxima a ser usada por cada processo de operador de autovacuum. Esse parâmetro é padronizado para -1, o que indica que você deve usar o valor de `maintenance_work_mem`.

Para obter mais informações sobre parâmetros de memória de autovacuum, consulte [Alocar memória para autovacuum](#) na documentação do Amazon RDS.

## Ajuste dos parâmetros de autovacuum

Os usuários talvez precisem ajustar os parâmetros de autovacuum, dependendo das operações de atualização e exclusão. As configurações dos parâmetros a seguir podem ser definidas no nível de tabela, instância ou cluster.

### Nível de cluster ou instância

Como exemplo, vejamos um banco de dados do setor financeiro em que se espera um volume contínuo de operações de Linguagem de Manipulação de Dados (DML). Para manter a integridade do banco de dados, você deve ajustar os parâmetros de autovacuum no nível de cluster para o Aurora e no nível de instância para o Amazon RDS, além de aplicar o mesmo grupo de parâmetros ao leitor. No caso de um failover, os mesmos parâmetros devem ser aplicados ao novo gravador.

### Nível de tabela

Por exemplo, em um banco de dados para entrega de alimentos em que se espera operações contínuas de DML em uma única tabela chamada `orders`, você deve considerar o ajuste do parâmetro `autovacuum_analyze_threshold` no nível de tabela usando o seguinte comando:

```
ALTER TABLE <table_name> SET (autovacuum_analyze_threshold = <threshold rows>)
```

### Uso de configurações agressivas de autovacuum no nível de tabela

A tabela `orders` de exemplo que tem operações contínuas de atualização e exclusão torna-se candidata à limpeza devido às configurações padrão de autovacuum. Isso leva à geração incorreta de planos e à lentidão nas consultas. Limpar o inchaço e atualizar as estatísticas requer configurações agressivas de autovacuum em nível de tabela.

Para determinar as configurações, acompanhe a duração das consultas em execução nessa tabela e identifique a porcentagem de operações de DML que resultam em alterações no plano. A visualização de `pg_stat_user_tables` ajuda a monitorar as operações de inserção, atualização e exclusão.

Exemplo:

Vamos supor que o otimizador gere planos ruins sempre que 5% da tabela `orders` muda. Nesse caso, você deve alterar o limite do fator de escala para 2% da seguinte forma:

```
ALTER TABLE orders SET (autovacuum_vacuum_scale_factor = 0.02)
```

### Tip

Selecione configurações agressivas de autovacuum com cuidado para evitar o alto consumo de recursos.

Para obter mais informações, consulte o seguinte:

- [Entendendo o autovacuum nos ambientes Amazon RDS for AWS PostgreSQL](#) (postagem no blog)
- [Automatic Vacuuming](#) (documentação do PostgreSQL)
- [Ajuste dos parâmetros do PostgreSQL no Amazon RDS e no Amazon Aurora](#) (orientação prescritiva)AWS

Para garantir que o autovacuum funcione de forma eficaz, monitore as linhas inativas, o uso do disco e a última vez em que o autovacuum ou o ANALYZE foi executado regularmente. A visualização `pg_stat_all_tables` fornece informações sobre cada tabela (`relname`) e quantas tuplas inativas (`n_dead_tup`) estão na tabela.

O monitoramento do número de tuplas inativas em cada tabela, especialmente em tabelas atualizadas com frequência, ajuda a determinar se os processos de autovacuum estão removendo periodicamente as tuplas inativas para que seu espaço em disco possa ser reutilizado para uma melhor performance. É possível usar a consulta abaixo para conferir o número de tuplas inativas e quando o último autovacuum foi executado nas tabelas:

```
SELECT
relname AS TableName,n_live_tup AS LiveTuples,n_dead_tup AS DeadTuples,
last_autovacuum AS Autovacuum,last_autoanalyze AS Autoanalyze_FROM
pg_stat_user_tables;
```

## Vantagens e limitações

O autovacuum oferece as seguintes vantagens:

- Ele remove o inchaço das tabelas automaticamente.
- Isso evita o wraparound do ID de transação.
- Ele mantém as estatísticas do banco de dados atualizadas.

#### Limitações:

- Se as consultas usarem processamento paralelo, o número de processos worker poderá não ser suficiente para o autovacuum.
- Se o autovacuum for executado nos horários de pico, a utilização de recursos poderá aumentar. Você deve ajustar os parâmetros para lidar com esse problema.
- Se as páginas da tabela estiverem ocupadas em outra sessão, o autovacuum poderá ignorá-las.
- O autovacuum não pode acessar tabelas temporárias.

## Limpeza e análise manual de tabelas

Se seu banco de dados for limpo pelo processo de autovacuum, é uma prática recomendada evitar a execução de verificações manuais em todo o banco de dados com muita frequência. Uma aspiração manual pode resultar em I/O cargas desnecessárias ou picos de CPU e também pode falhar na remoção de tuplas mortas. Execute aspiradores manuais somente table-by-table se for realmente necessário, como quando a proporção de tuplas vivas e mortas é baixa ou quando há longos espaços entre os aspiradores automáticos. Além disso, você deve executar aspiradores manuais quando a atividade do usuário for mínima.

O autovacuum também mantém as estatísticas de uma tabela atualizadas. Quando você executa o comando ANALYZE manualmente, ele recria essas estatísticas em vez de atualizá-las. A recriação de estatísticas quando elas já estão atualizadas pelo processo normal de autovacuum pode resultar na utilização de recursos do sistema.

Recomendamos que você execute os comandos [VACUUM](#) e [ANALYZE](#) manualmente nos seguintes cenários:

- Durante horários de baixo pico em tabelas mais movimentadas, quando a limpeza automática pode não ser suficiente.
- Imediatamente depois de carregar dados em massa na tabela de destino. Nesse caso, executar ANALYZE manualmente recria completamente as estatísticas, o que é uma opção melhor do que esperar o início do autovacuum.
- Para limpar tabelas temporárias (o autovacuum não pode acessá-las).

Para reduzir o I/O impacto ao executar os ANALYZE comandos VACUUM e na atividade simultânea do banco de dados, você pode usar o `vacuum_cost_delay` parâmetro. Em muitas situações, comandos de manutenção como VACUUM e ANALYZE não precisam ser concluídos rapidamente. No entanto, esses comandos não devem interferir na capacidade do sistema de realizar outras operações de banco de dados. Para evitar isso, você pode habilitar atrasos de vacuum baseados em custos usando o parâmetro `vacuum_cost_delay`. Por padrão, esse parâmetro é desabilitado para comandos VACUUM emitidos manualmente. Para habilitá-lo, defina-o com um valor diferente de zero.

## Execução de operações de vacuum e limpeza em paralelo

A opção [PARALLEL](#) do comando VACUUM usa operadores paralelos para as fases de vacuum e limpeza do índice e está desabilitada por padrão. O número de operadores paralelos (o grau de paralelismo) é determinado pelo número de índices na tabela e pode ser especificado pelo usuário. Se você estiver executando operações VACUUM paralelas sem um argumento inteiro, o grau de paralelismo será calculado com base no número de índices na tabela.

Os parâmetros a seguir ajudam você a configurar a operação de vacuum paralela no Amazon RDS para PostgreSQL e no Aurora compatível com PostgreSQL:

- [max\\_worker\\_processes](#) define o número máximo de processos de operadores simultâneos.
- [min\\_parallel\\_index\\_scan\\_size](#) define a quantidade mínima de dados de índice que devem ser verificados para que uma verificação paralela seja considerada.
- [max\\_parallel\\_maintenance\\_workers](#) define o número máximo de operadores paralelos que podem ser iniciados por um único comando de utilitário.

### Note

A opção PARALLEL é usada somente para fins de limpeza. Não afeta o comando ANALYZE.

O exemplo a seguir ilustra o comportamento do banco de dados quando você usa o VACUUM e ANALYZE manuais em um banco de dados.

Confira um exemplo de tabela em que o autovacuum foi desabilitado (apenas para fins ilustrativos; desabilitar o autovacuum não é recomendado):

```
create table t1 ( a int, b int, c int );
alter table t1 set (autovacuum_enabled=false);
```

```
apgl=> \d+ t1
Table "public.t1"
Column | Type | Collation | Nullable | Default | Storage | Stats target | Description
-----+-----+-----+-----+-----+-----+-----+-----
+-----+
a | integer | | | plain | |
b | integer | | | plain | |
```

```
c | integer | | | plain | |
Access method: heap
Options: autovacuum_enabled=false
```

Adicionar 1 milhão de linhas à tabela t1:

```
apgl=> select count(*) from t1;
count
1000000
(1 row)
```

Estatísticas da tabela t1:

```
select * from pg_stat_all_tables where relname='t1';
-[ RECORD 1 ]-----+-----
relid          | 914744
schemaname     | public
relname        | t1
seq_scan       | 0
seq_tup_read   | 0
idx_scan       |
idx_tup_fetch  |
n_tup_ins      | 1000000
n_tup_upd      | 0
n_tup_del      | 0
n_tup_hot_upd  | 0
n_live_tup     | 1000000
n_dead_tup     | 0
n_mod_since_analyze | 1000000
last_vacuum    |
last_autovacuum |
last_analyze   |
last_autoanalyze |
vacuum_count   | 0
autovacuum_count | 0
analyze_count  | 0
autoanalyze_count | 0
```

Adicionar um índice:

```
create index i2 on t1 (b,a);
```

**Execute o comando EXPLAIN (Plano 1):**

```
Bitmap Heap Scan on t1 (cost=10521.17..14072.67 rows=5000 width=4)
Recheck Cond: (a = 5)
# Bitmap Index Scan on i2 (cost=0.00..10519.92 rows=5000 width=0)
Index Cond: (a = 5)
(4 rows)
```

**Execute o comando EXPLAIN ANALYZE (Plano 2):**

```
explain (analyze, buffers, costs off) select a from t1 where b = 5;
QUERY PLAN
Bitmap Heap Scan on t1 (actual time=0.023..0.024 rows=1 loops=1)
Recheck Cond: (b = 5)
Heap Blocks: exact=1
Buffers: shared hit=4
# Bitmap Index Scan on i2 (actual time=0.016..0.016 rows=1 loops=1)
Index Cond: (b = 5)
Buffers: shared hit=3
Planning Time: 0.054 ms
Execution Time: 0.076 ms
(9 rows)
```

Os comandos EXPLAIN e EXPLAIN ANALYZE exibem planos diferentes, porque o autovacuum foi desabilitado na tabela e o comando ANALYZE não foi executado manualmente. Agora vamos atualizar um valor na tabela e regenerar o plano EXPLAIN ANALYZE:

```
update t1 set a=8 where b=5;
explain (analyze, buffers, costs off) select a from t1 where b = 5;
```

**O comando EXPLAIN ANALYZE (Plano 3) agora exhibe:**

```
apgl=> explain (analyze, buffers, costs off) select a from t1 where b = 5;
QUERY PLAN
Bitmap Heap Scan on t1 (actual time=0.075..0.076 rows=1 loops=1)
Recheck Cond: (b = 5)
Heap Blocks: exact=1
Buffers: shared hit=5
# Bitmap Index Scan on i2 (actual time=0.017..0.017 rows=2 loops=1)
Index Cond: (b = 5)
```

```

Buffers: shared hit=3
Planning Time: 0.053 ms
Execution Time: 0.125 ms

```

Se você comparar os custos entre o Plano 2 e o Plano 3, verá as diferenças no tempo de planejamento e execução, porque ainda não coletamos estatísticas.

Agora vamos executar um ANALYZE manual na tabela, verificar as estatísticas e regenerar o plano:

```

apgl=> analyze t1
apgl# ;
ANALYZE
Time: 212.223 ms

apgl=> select * from pg_stat_all_tables where relname='t1';
-[ RECORD 1 ]-----+-----
relid          | 914744
schemaname     | public
relname        | t1
seq_scan       | 3
seq_tup_read   | 1000000
idx_scan       | 3
idx_tup_fetch  | 3
n_tup_ins      | 1000000
n_tup_upd      | 1
n_tup_del      | 0
n_tup_hot_upd  | 0
n_live_tup     | 1000000
n_dead_tup     | 1
n_mod_since_analyze | 0
last_vacuum    |
last_autovacuum |
last_analyze   | 2023-04-15 11:39:02.075089+00
last_autoanalyze |
vacuum_count   | 0
autovacuum_count | 0
analyze_count  | 1
autoanalyze_count | 0

Time: 148.347 ms

```

Execute o comando EXPLAIN ANALYZE (Plano 4):

```
apgl=> explain (analyze,buffers,costs off) select a from t1 where b = 5;
QUERY PLAN
Index Only Scan using i2 on t1 (actual time=0.022..0.023 rows=1 loops=1)
Index Cond: (b = 5)
Heap Fetches: 1
Buffers: shared hit=4
Planning Time: 0.056 ms
Execution Time: 0.068 ms
(6 rows)

Time: 138.462 ms
```

Se comparar todos os resultados do plano depois de analisar manualmente a tabela e coletar estatísticas, você observará que o Plano 4 do otimizador é melhor do que os outros e que também diminui o tempo de execução da consulta. Este exemplo mostra como é importante executar atividades de manutenção no banco de dados.

## Reescrita de toda uma tabela com VACUUM FULL

A execução do comando `VACUUM` com o parâmetro `FULL` regrava todo o conteúdo de uma tabela em um novo arquivo de disco sem espaço extra e retorna espaço não utilizado para o sistema operacional. Essa operação é muito mais lenta e requer um bloqueio `ACCESS EXCLUSIVE` em cada tabela. Também requer espaço extra em disco, pois grava uma nova cópia da tabela e não libera a cópia antiga até que a operação seja concluída.

O `VACUUM FULL` pode ser útil nos seguintes casos:

- Quando você quiser recuperar um volume significativo de espaço nas tabelas.
- Quando você quiser recuperar espaço inchado em tabelas de chaves não primárias.

Recomendamos que você use `VACUUM FULL` quando tiver tabelas de chaves não primárias, se seu banco de dados puder tolerar tempo de inatividade.

Como o `VACUUM FULL` requer mais bloqueio do que outras operações, é mais caro executá-lo em bancos de dados cruciais. Para substituir esse método, você pode usar a extensão `pg_repack`, descrita na [próxima seção](#). Essa opção é semelhante ao `VACUUM FULL`, mas requer bloqueio mínimo e é compatível com o Amazon RDS para PostgreSQL e o Aurora compatível com PostgreSQL.

## Remoção do inchaço com pg\_repack

Você pode usar a extensão `pg_repack` para remover o inchaço da tabela e do índice com o mínimo de bloqueio do banco de dados. Você pode criar essa extensão na instância do banco de dados e executar o cliente `pg_repack` (em que a versão do cliente corresponde à versão da extensão) do Amazon Elastic Compute Cloud (Amazon EC2) ou de um computador que possa se conectar ao seu banco de dados.

Diferentemente do `VACUUM FULL`, o `pg_repack` não exige tempo de inatividade ou janela de manutenção e não bloqueia outras sessões.

O `pg_repack` é útil em situações em que `VACUUM FULL`, `CLUSTER` ou `REINDEX` pode não funcionar. Ele cria uma tabela que contém os dados da tabela inchada, monitora as alterações na tabela original e, em seguida, substitui a tabela original pela nova. Ele não bloqueia a tabela original para operações de leitura ou gravação enquanto está criando a outra tabela.

Você pode usar `pg_repack` para uma tabela completa ou para um índice. Para ver uma lista de tarefas, consulte a [documentação do pg\\_repack](#).

Limitações:

- Para executar o `pg_repack`, sua tabela deve ter uma chave primária ou um índice exclusivo.
- O `pg_repack` não funcionará com tabelas temporárias.
- O `pg_repack` não funcionará em tabelas que tenham índices globais.
- Quando o `pg_repack` está em andamento, não é possível realizar operações de DDL em tabelas.

A tabela a seguir descreve as diferenças entre `pg_repack` e `VACUUM FULL`.

VACUUM FULL	pg_repack
Comando integrado	Uma extensão que você executa no Amazon EC2 ou no seu computador local
Requer um bloqueio ACCESS EXCLUSIVE enquanto está operando em uma tabela	Requer um bloqueio ACCESS EXCLUSIVE apenas por um curto período

Funciona com todas as tabelas

Funciona em tabelas que têm somente chaves primárias e exclusivas

Requer o dobro do armazenamento consumido pela tabela e pelos índices

Requer o dobro do armazenamento consumido pela tabela e pelos índices

Para executar `pg_repack` em uma tabela, use o comando:

```
pg_repack -h <host> -d <dbname> --table <tablename> -k
```

Para executar `pg_repack` em um índice, use o comando:

```
pg_repack -h <host> -d <dbname> --index <index name>
```

Para obter mais informações, consulte a publicação do Blog da AWS [Remove bloat from Amazon Aurora and RDS for PostgreSQL with pg\\_repack](#).

#### Advertência

O erro `error-on-invalid-index` no `pg_repack` geralmente significa que um ou mais índices na tabela estão corrompidos ou são inválidos. O `pg_repack` não pode operar com segurança em tabelas com índices inválidos, porque depende dos índices para a consistência de dados durante o processo de empacotamento.

Esse erro ocorre quando:

- O índice está marcado como inválido (por exemplo, devido a uma instrução `CREATE INDEX CONCURRENTLY` com falha).
- O índice está corrompido (possivelmente devido a problemas de hardware ou desligamentos abruptos).

Use a consulta a seguir para identificar índices inválidos e eliminá-los primeiro, caso você os encontre.

```
SELECT indexrelid::regclass, indisvalid FROM pg_index WHERE indrelid =  
'orders'::regclass AND NOT indisvalid; Drop the invalid index: DROP INDEX  
index_name;
```

## Recriação de índices

O comando [REINDEX](#) do PostgreSQL recria um índice usando os dados armazenados na tabela do índice e substituindo a cópia antiga do índice. Recomendamos que use REINDEX nos seguintes cenários:

- Quando um índice está corrompido e não contém mais dados válidos. Isso pode acontecer como resultado de falhas de software ou hardware.
- Quando as consultas que anteriormente usavam o índice param de usá-lo.
- Quando o índice fica inchado com um grande número de páginas vazias ou praticamente vazias. Você deve executar REINDEX quando a porcentagem de inchaço (`bloat_pct`) for maior que 20.

A consulta a seguir ajuda você a encontrar `bloat_pct`:

```
SELECT current_database(), nspname AS schemaname, tblname, idxname,
bs*(relpages)::bigint AS real_size,
bs*(relpages-est_pages)::bigint AS extra_size,
100 * (relpages-est_pages)::float / relpages AS extra_pct,
fillfactor,
CASE WHEN relpages > est_pages_ff
  THEN bs*(relpages-est_pages_ff)
  ELSE 0
END AS bloat_size,
100 * (relpages-est_pages_ff)::float / relpages AS bloat_pct,
is_na
-- , 100-(pst).avg_leaf_density AS pst_avg_bloat, est_pages, index_tuple_hdr_bm,
maxalign, pagehdr, nulldatawidth, nulldatahdrwidth, reltuples, relpages -- (DEBUG
INFO)
FROM (
  SELECT coalesce(1 +
    ceil(reltuples/floor((bs-pageopqdata-pagehdr)/(4+nulldatahdrwidth)::float)), 0
-- ItemIdData size + computed avg size of a tuple (nulldatahdrwidth)
  ) AS est_pages,
  coalesce(1 +
    ceil(reltuples/floor((bs-pageopqdata-pagehdr)*fillfactor/
(100*(4+nulldatahdrwidth)::float))), 0
  ) AS est_pages_ff,
  bs, nspname, tblname, idxname, relpages, fillfactor, is_na
```

```

-- , pgstatindex(idxoid) AS pst, index_tuple_hdr_bm, maxalign, pagehdr,
nulldatawidth, nulldatahdrwidth, reltuples -- (DEBUG INFO)
FROM (
    SELECT maxalign, bs, nspname, tblname, idxname, reltuples, relpages, idxoid,
fillfactor,
        ( index_tuple_hdr_bm +
            maxalign - CASE -- Add padding to the index tuple header to align on
MAXALIGN
                WHEN index_tuple_hdr_bm%maxalign = 0 THEN maxalign
                ELSE index_tuple_hdr_bm%maxalign
            END
        + nulldatawidth + maxalign - CASE -- Add padding to the data to align on
MAXALIGN
                WHEN nulldatawidth = 0 THEN 0
                WHEN nulldatawidth::integer%maxalign = 0 THEN maxalign
                ELSE nulldatawidth::integer%maxalign
            END
        )::numeric AS nulldatahdrwidth, pagehdr, pageopqdata, is_na
-- , index_tuple_hdr_bm, nulldatawidth -- (DEBUG INFO)
FROM (
    SELECT n.nspname, i.tblname, i.idxname, i.reltuples, i.relpages,
        i.idxoid, i.fillfactor, current_setting('block_size')::numeric AS bs,
        CASE -- MAXALIGN: 4 on 32bits, 8 on 64bits (and mingw32 ?)
            WHEN version() ~ 'mingw32' OR version() ~ '64-bit|x86_64|ppc64|ia64|
amd64' THEN 8
            ELSE 4
        END AS maxalign,
        /* per page header, fixed size: 20 for 7.X, 24 for others */
        24 AS pagehdr,
        /* per page btree opaque data */
        16 AS pageopqdata,
        /* per tuple header: add IndexAttributeBitMapData if some cols are null-
able */
        CASE WHEN max(coalesce(s.null_frac,0)) = 0
            THEN 8 -- IndexTupleData size
            ELSE 8 + (( 32 + 8 - 1 ) / 8) -- IndexTupleData size +
IndexAttributeBitMapData size ( max num filed per index + 8 - 1 /8)
        END AS index_tuple_hdr_bm,
        /* data len: we remove null values save space using it fractionnal part
from stats */
        sum( (1-coalesce(s.null_frac, 0)) * coalesce(s.avg_width, 1024)) AS
nulldatawidth,

```

```

        max( CASE WHEN i.atttypid = 'pg_catalog.name'::regtype THEN 1 ELSE 0
END ) > 0 AS is_na
    FROM (
        SELECT ct.relname AS tblname, ct.relnamespace, ic.idxname, ic.attpos,
ic.indkey, ic.indkey[ic.attpos], ic.reltuples, ic.relpages, ic.tbloid, ic.idxoid,
ic.fillfactor,
        coalesce(a1.attnum, a2.attnum) AS attnum, coalesce(a1.attname,
a2.attname) AS attname, coalesce(a1.atttypid, a2.atttypid) AS atttypid,
        CASE WHEN a1.attnum IS NULL
        THEN ic.idxname
        ELSE ct.relname
        END AS attrelname
    FROM (
        SELECT idxname, reltuples, relpages, tbloid, idxoid, fillfactor,
indkey,
        pg_catalog.generate_series(1,indnatts) AS attpos
    FROM (
        SELECT ci.relname AS idxname, ci.reltuples, ci.relpages,
i.indrelid AS tbloid,
        i.indexrelid AS idxoid,
        coalesce(substring(
            array_to_string(ci.reloptions, ' ')
            from 'fillfactor=([0-9]+)')::smallint, 90) AS fillfactor,
        i.indnatts,
        pg_catalog.string_to_array(pg_catalog.textin(
            pg_catalog.int2vectorout(i.indkey)), ' ')::int[] AS indkey
    FROM pg_catalog.pg_index i
    JOIN pg_catalog.pg_class ci ON ci.oid = i.indexrelid
    WHERE ci.relam=(SELECT oid FROM pg_am WHERE amname = 'btree')
    AND ci.relpages > 0
    ) AS idx_data
    ) AS ic
    JOIN pg_catalog.pg_class ct ON ct.oid = ic.tbloid
    LEFT JOIN pg_catalog.pg_attribute a1 ON
        ic.indkey[ic.attpos] <> 0
        AND a1.attrelid = ic.tbloid
        AND a1.attnum = ic.indkey[ic.attpos]
    LEFT JOIN pg_catalog.pg_attribute a2 ON
        ic.indkey[ic.attpos] = 0
        AND a2.attrelid = ic.idxoid
        AND a2.attnum = ic.attpos
    ) i
    JOIN pg_catalog.pg_namespace n ON n.oid = i.relnamespace

```

```

        JOIN pg_catalog.pg_stats s ON s.schemaname = n.nspname
                                AND s.tablename = i.attrelname
                                AND s.attname = i.attname
        GROUP BY 1,2,3,4,5,6,7,8,9,10,11
    ) AS rows_data_stats
) AS rows_hdr_pdg_stats
) AS relation_stats
ORDER BY nspname, tblname, idxname;

```

As páginas de índice que estão completamente vazias são recuperadas para reutilização. No entanto, recomendamos a reindexação periódica se as chaves de índice em uma página tiverem sido excluídas, mas o espaço permanecer alocado.

A recriação do índice ajuda a melhorar a performance da consulta. Você pode recriar um índice de três maneiras, conforme descrito na tabela a seguir.

Method (Método)	Descrição	Limitações
CREATE INDEX e DROP INDEX com a opção CONCURRENTLY	Cria um novo índice e remove o índice antigo. O otimizador gera planos usando o índice recém-criado em vez do índice antigo. Durante os horários de baixo pico, você pode descartar o índice antigo.	A criação do índice leva mais tempo quando você usa a opção CONCURRENTLY , pois ela precisa rastrear todas as alterações recebidas. Quando as alterações são congeladas, o processo é marcado como concluído.
REINDEX com a opção CONCURRENTLY	Bloqueia as operações de gravação durante o processo de recriação. A versão 12 do PostgreSQL e as versões posteriores oferecem a opção CONCURRENTLY , que evita esses bloqueios.	O uso de CONCURRENTLY requer mais tempo para recriar o índice.
Extensão do pg_repack	Limpa o inchaço de uma tabela e recria o índice.	Você deve executar essa extensão de uma instância do EC2 ou de um computador

local conectado ao banco de dados.

## Criação de um novo índice

Os comandos `DROP INDEX` e `CREATE INDEX`, quando usados juntos, recriam um índice:

```
DROP INDEX <index_name>
CREATE INDEX <index_name> ON TABLE <table_name> (<column1>[,<column2>])
```

A desvantagem dessa abordagem é seu bloqueio exclusivo na tabela, o que afeta a performance durante essa atividade. O comando `DROP INDEX` adquire um bloqueio exclusivo, que bloqueia as operações de leitura e gravação na tabela. O comando `CREATE INDEX` bloqueia as operações de gravação na tabela. Ele permite operações de leitura, mas elas são caras durante a criação do índice.

## Recriação de um índice

O comando `REINDEX` ajuda você a manter a performance consistente do banco de dados. Quando você executa um grande número de operações de DML em uma tabela, elas resultam em inchaço da tabela e do índice. Os índices são usados para acelerar a pesquisa em tabelas e melhorar a performance da consulta. O inchaço do índice afeta as pesquisas e a performance das consultas. Portanto, recomendamos que você execute a reindexação em tabelas que tenham um alto volume de operações de DML para manter a consistência na performance das consultas.

O comando `REINDEX` recria o índice do zero bloqueando as operações de gravação na tabela subjacente, mas permite operações de leitura na tabela. No entanto, ele bloqueia as operações de leitura no índice. As consultas que usam o índice correspondente são bloqueadas, mas outras consultas não.

A versão 12 do PostgreSQL introduziu um novo parâmetro opcional, `CONCURRENTLY`, que recria o índice do zero, mas não bloqueia as operações de gravação ou leitura na tabela ou nas consultas que usam o índice. No entanto, leva mais tempo para concluir o processo quando você usa essa opção.

# Exemplos

## Criação e exclusão de um índice

Crie um novo índice com a opção CONCURRENTLY:

```
create index CONCURRENTLY on table(columns) ;
```

Exclua o índice antigo com a opção CONCURRENTLY:

```
drop index CONCURRENTLY <index name> ;
```

## Recriação de um índice

Para recriar um único índice:

```
reindex index <index name> ;
```

Para recriar todos os índices em uma tabela:

```
reindex table <table name> ;
```

Para recriar todos os índices em um esquema:

```
reindex schema <schema name> ;
```

## Recriação de um índice simultaneamente

Para recriar um único índice:

```
reindex index CONCURRENTLY <indexname> ;
```

Para recriar todos os índices em uma tabela:

```
reindex table CONCURRENTLY <tablename> ;
```

Para recriar todos os índices em um esquema:

```
reindex schema CONCURRENTLY <schemaname> ;
```

## Recriação ou realocamento somente de índices

Para recriar um único índice:

```
pg_repack -h <hostname> -d <dbname> -i <indexname> -k
```

Para recriar todos os índices:

```
pg_repack -h <hostname> -d <dbname> -x <indexname> -t <tablename> -k
```

## Exemplo: recuperação de espaço usando autovacuum e VACUUM FULL

Como exemplo, vamos criar uma tabela emp com 500 mil linhas e, em seguida, atualizar as linhas com novos valores. O Autovacuum está habilitado, então ele executará os comandos VACUUM e ANALYZE nessa tabela para remover o inchaço e recuperar espaço. O espaço recuperado pode ser reutilizado, mas não será devolvido ao sistema operacional.

A consulta a seguir determina o inchaço na tabela:

```
-- WARNING: When run with a non-superuser role, the query inspects only indexes on
tables you are granted to read.
-- WARNING: Rows with is_na = 't' are known to have bad statistics ("name" type is not
supported).
-- This query is compatible with PostgreSQL 8.2 and later.
SELECT current_database(), nspname AS schemaname, tblname, idxname,
bs*(relpages)::bigint AS real_size,
bs*(relpages-est_pages)::bigint AS extra_size,
100 * (relpages-est_pages)::float / relpages AS extra_pct,
fillfactor,
CASE WHEN relpages > est_pages_ff
THEN bs*(relpages-est_pages_ff)
ELSE 0
END AS bloat_size,
100 * (relpages-est_pages_ff)::float / relpages AS bloat_pct,
is_na
-- , 100-(pst).avg_leaf_density AS pst_avg_bloat, est_pages, index_tuple_hdr_bm,
maxalign, pagehdr, nulldatawidth, nulldatahdrwidth, reltuples, relpages -- (DEBUG
INFO)
FROM (
SELECT coalesce(1 +
ceil(reltuples/floor((bs-pageopqdata-pagehdr)/(4+nulldatahdrwidth)::float)), 0
-- ItemIdData size + computed avg size of a tuple (nulldatahdrwidth)
) AS est_pages,
coalesce(1 +
ceil(reltuples/floor((bs-pageopqdata-pagehdr)*fillfactor/
(100*(4+nulldatahdrwidth)::float))), 0
) AS est_pages_ff,
bs, nspname, tblname, idxname, relpages, fillfactor, is_na
```

```

-- , pgstatindex(idxoid) AS pst, index_tuple_hdr_bm, maxalign, pagehdr,
nulldatawidth, nulldatahdrwidth, reltuples -- (DEBUG INFO)
FROM (
    SELECT maxalign, bs, nspname, tblname, idxname, reltuples, relpages, idxoid,
fillfactor,
        ( index_tuple_hdr_bm +
            maxalign - CASE -- Add padding to the index tuple header to align on
MAXALIGN
                WHEN index_tuple_hdr_bm%maxalign = 0 THEN maxalign
                ELSE index_tuple_hdr_bm%maxalign
            END
        + nulldatawidth + maxalign - CASE -- Add padding to the data to align on
MAXALIGN
                WHEN nulldatawidth = 0 THEN 0
                WHEN nulldatawidth::integer%maxalign = 0 THEN maxalign
                ELSE nulldatawidth::integer%maxalign
            END
        )::numeric AS nulldatahdrwidth, pagehdr, pageopqdata, is_na
-- , index_tuple_hdr_bm, nulldatawidth -- (DEBUG INFO)
FROM (
    SELECT n.nspname, i.tblname, i.idxname, i.reltuples, i.relpages,
        i.idxoid, i.fillfactor, current_setting('block_size')::numeric AS bs,
        CASE -- MAXALIGN: 4 on 32bits, 8 on 64bits (and mingw32 ?)
            WHEN version() ~ 'mingw32' OR version() ~ '64-bit|x86_64|ppc64|ia64|
amd64' THEN 8
            ELSE 4
        END AS maxalign,
        /* per page header, fixed size: 20 for 7.X, 24 for others */
        24 AS pagehdr,
        /* per page btree opaque data */
        16 AS pageopqdata,
        /* per tuple header: add IndexAttributeBitMapData if some cols are null-
able */
        CASE WHEN max(coalesce(s.null_frac,0)) = 0
            THEN 8 -- IndexTupleData size
            ELSE 8 + (( 32 + 8 - 1 ) / 8) -- IndexTupleData size +
IndexAttributeBitMapData size ( max num filed per index + 8 - 1 /8)
        END AS index_tuple_hdr_bm,
        /* data len: we remove null values save space using it fractionnal part
from stats */
        sum( (1-coalesce(s.null_frac, 0)) * coalesce(s.avg_width, 1024)) AS
nulldatawidth,

```

```

        max( CASE WHEN i.atttypid = 'pg_catalog.name'::regtype THEN 1 ELSE 0
END ) > 0 AS is_na
    FROM (
        SELECT ct.relname AS tblname, ct.relnamespace, ic.idxname, ic.attpos,
ic.indkey, ic.indkey[ic.attpos], ic.reltuples, ic.relpages, ic.tbloid, ic.idxoid,
ic.fillfactor,
        coalesce(a1.attnum, a2.attnum) AS attnum, coalesce(a1.attname,
a2.attname) AS attname, coalesce(a1.atttypid, a2.atttypid) AS atttypid,
        CASE WHEN a1.attnum IS NULL
        THEN ic.idxname
        ELSE ct.relname
        END AS attrelname
    FROM (
        SELECT idxname, reltuples, relpages, tbloid, idxoid, fillfactor,
indkey,
        pg_catalog.generate_series(1,indnatts) AS attpos
    FROM (
        SELECT ci.relname AS idxname, ci.reltuples, ci.relpages,
i.indrelid AS tbloid,
        i.indexrelid AS idxoid,
        coalesce(substring(
            array_to_string(ci.reloptions, ' ')
            from 'fillfactor=([0-9]+)')::smallint, 90) AS fillfactor,
        i.indnatts,
        pg_catalog.string_to_array(pg_catalog.textin(
            pg_catalog.int2vectorout(i.indkey)), ' ')::int[] AS indkey
    FROM pg_catalog.pg_index i
    JOIN pg_catalog.pg_class ci ON ci.oid = i.indexrelid
    WHERE ci.relam=(SELECT oid FROM pg_am WHERE amname = 'btree')
    AND ci.relpages > 0
    ) AS idx_data
    ) AS ic
    JOIN pg_catalog.pg_class ct ON ct.oid = ic.tbloid
    LEFT JOIN pg_catalog.pg_attribute a1 ON
        ic.indkey[ic.attpos] <> 0
        AND a1.attrelid = ic.tbloid
        AND a1.attnum = ic.indkey[ic.attpos]
    LEFT JOIN pg_catalog.pg_attribute a2 ON
        ic.indkey[ic.attpos] = 0
        AND a2.attrelid = ic.idxoid
        AND a2.attnum = ic.attpos
    ) i
    JOIN pg_catalog.pg_namespace n ON n.oid = i.relnamespace

```

```

        JOIN pg_catalog.pg_stats s ON s.schemaname = n.nspname
        AND s.tablename = i.attrelname
        AND s.attname = i.attname
    GROUP BY 1,2,3,4,5,6,7,8,9,10,11
) AS rows_data_stats
) AS rows_hdr_pdg_stats
) AS relation_stats
ORDER BY nspname, tblname, idxname;

```

Os resultados da consulta mostram que a tabela tem um inchaço de cerca de 51%:

```

current_database | schemaname | tblname | real_size | extra_size | extra_pct |
fillfactor | bloat_size | bloat_pct | is_na
-----+-----+-----+-----+-----+-----
apgl | public | emp | 60383232 | 30744576 | 50.91575091575091 | 100 | 30744576 |
50.91575091575091 | f

```

Aqui estão as estatísticas da visualização pg\_stat\_all\_tables:

```

relid          | 914748
schemaname     | public
relname        | emp
seq_scan       | 5
seq_tup_read   | 1500000
idx_scan       | 0
idx_tup_fetch  | 0
n_tup_ins      | 600000
n_tup_upd      | 500000
n_tup_del      | 0
n_tup_hot_upd  | 0
n_live_tup     | 500000
n_dead_tup     | 0
n_mod_since_analyze | 0
last_vacuum    |
last_autovacuum | 2023-04-15 11:59:54.957449+00
last_analyze   |
last_autoanalyze | 2023-04-15 11:59:55.016352+00
vacuum_count   | 0
autovacuum_count | 2
analyze_count  | 0
autoanalyze_count | 3

```

Observe que o autovacuum atualizou as colunas `last_autovacuum` e `last_autoanalyze` após sua execução.

Agora, vamos inserir algumas linhas na tabela e verificar `extra_size(bloat_size)`, pois o espaço vazio também é considerado inchaço.

```
apgl=> select count(*) from emp;
count | 900000

current_database | schemaname | tblname | real_size | extra_size | extra_pct |
fillfactor | bloat_size | bloat_pct | is_na
-----+-----+-----+-----+-----+-----
+-----+-----+-----+-----+-----
apgl | public | emp | 61349888 | 327680 | 0.5341167044999332 | 100 | 327680 |
0.5341167044999332 | f
(1 row)
```

A coluna `bloat_pct` na saída indica que o espaço limpo foi ocupado por novas inserções. Vamos executar o `VACUUM FULL`:

```
apgl=> vacuum full emp ;
VACUUM

current_database | schemaname | tblname | real_size | extra_size | extra_pct |
fillfactor | bloat_size | bloat_pct | is_na
-----+-----+-----+-----+-----+-----
+-----+-----+-----+-----+-----
apgl | public | emp | 60792832 | -229376 | 0 | 100 | 0 | 0 | f
(1 row)
```

Nessa saída, você pode ver que o espaço vazio e o inchaço foram removidos e que o espaço foi retornado ao sistema operacional.

#### Note

Em vez do `VACUUM FULL`, você pode executar `pg_repack` para obter os mesmos resultados.

---

# Recursos

- [Entendendo o autovacuum nos ambientes Amazon RDS forAWS PostgreSQL](#) (postagem no blog)
- [Automatic vacuuming](#) (documentação do PostgreSQL)
- [Alocar memória para autovacuum](#) (documentação do Amazon RDS)
- [Prevenindo falhas no Wraparound do ID da transação; \(documentação do PostgreSQL\)](#)
- [Remove bloat from Amazon Aurora and RDS for PostgreSQL with pg\\_repack](#) (publicação do Blog da AWS )

## Histórico do documento

A tabela a seguir descreve alterações significativas feitas neste guia. Se desejar receber notificações sobre futuras atualizações, inscreva-se em um [feed RSS](#).

Alteração	Descrição	Data
<a href="#">Atualizações</a>	Erros corrigidos e informações adicionadas às seções <a href="#">Limpeza e análise automáticas de tabelas</a> e <a href="#">Remoção do inchaço com pg_repack</a> .	22 de agosto de 2025
<a href="#">Sintaxe reindex corrigida</a>	Na seção para <a href="#">recriar um índice simultaneamente</a> , os exemplos de reindex foram corrigidos.	30 de junho de 2025
<a href="#">Publicação inicial</a>	—	22 de dezembro de 2023

# AWS Glossário de orientação prescritiva

A seguir estão os termos comumente usados em estratégias, guias e padrões fornecidos pela Orientação AWS Prescritiva. Para sugerir entradas, use o link Fornecer feedback no final do glossário.

## Números

### 7 Rs

Sete estratégias comuns de migração para mover aplicações para a nuvem. Essas estratégias baseiam-se nos 5 Rs identificados pela Gartner em 2011 e consistem em:

- Refatorar/rearquitetar: mova uma aplicação e modifique sua arquitetura aproveitando ao máximo os recursos nativos de nuvem para melhorar a agilidade, a performance e a escalabilidade. Isso normalmente envolve a portabilidade do sistema operacional e do banco de dados. Exemplo: migrar seu banco de dados Oracle on-premises para o Amazon Aurora Edição Compatível com PostgreSQL.
- Redefinir a plataforma (mover e redefinir [mover e redefinir (lift-and-reshape)]): mova uma aplicação para a nuvem e introduza algum nível de otimização a fim de aproveitar os recursos da nuvem. Exemplo: migrar seu banco de dados Oracle on-premises para o Amazon Relational Database Service (Amazon RDS) para Oracle na Nuvem AWS.
- Recomprar (drop and shop): mude para um produto diferente, normalmente migrando de uma licença tradicional para um modelo SaaS. Exemplo: migrar seu sistema de gerenciamento de relacionamento com o cliente (CRM) para o Salesforce.com.
- Redefinir a hospedagem (mover sem alterações [lift-and-shift]) mover uma aplicação para a nuvem sem fazer nenhuma alteração a fim de aproveitar os recursos da nuvem. Exemplo: migrar seu banco de dados Oracle on-premises para o Oracle em uma instância do EC2 na Nuvem AWS.
- Realocar (mover o hipervisor sem alterações [hypervisor-level lift-and-shift]): mover a infraestrutura para a nuvem sem comprar novo hardware, reescrever aplicações ou modificar suas operações existentes. Você migra servidores de uma plataforma on-premises para um serviço de nuvem para a mesma plataforma. Exemplo: Migrar um Microsoft Hyper-V aplicativo para o AWS
- Reter (revisitar): mantenha as aplicações em seu ambiente de origem. Isso pode incluir aplicações que exigem grande refatoração, e você deseja adiar esse trabalho para um

momento posterior, e aplicações antigas que você deseja manter porque não há justificativa comercial para migrá-las.

- Retirar: desative ou remova aplicações que não são mais necessárias em seu ambiente de origem.

## A

### ABAC

Consulte [controle de acesso baseado em atributo](#).

serviços abstraídos

Veja [serviços gerenciados](#).

### ACID

Veja [atomicidade, consistência, isolamento, durabilidade](#).

migração ativa-ativa

Um método de migração de banco de dados no qual os bancos de dados de origem e de destino são mantidos em sincronia (por meio de uma ferramenta de replicação bidirecional ou operações de gravação dupla), e ambos os bancos de dados lidam com transações de aplicações conectadas durante a migração. Esse método oferece suporte à migração em lotes pequenos e controlados, em vez de exigir uma substituição única. É mais flexível, mas exige mais trabalho do que a [migração ativa-passiva](#).

migração ativa-passiva

Um método de migração de banco de dados em que os bancos de dados de origem e de destino são mantidos em sincronia, mas somente o banco de dados de origem manipula as transações das aplicações conectadas, enquanto os dados são replicados no banco de dados de destino. O banco de dados de destino não aceita nenhuma transação durante a migração.

### AGGREGATE FUNCTION

Uma função SQL que opera em um grupo de linhas e calcula um único valor de retorno para o grupo. Exemplos de funções agregadas incluem SUM e MAX.

## AI

Veja [inteligência artificial](#).

---

## AIOps

Veja [operações de inteligência artificial](#).

### anonimização

O processo de excluir permanentemente informações pessoais em um conjunto de dados. A anonimização pode ajudar a proteger a privacidade pessoal. Dados anônimos não são mais considerados dados pessoais.

### antipadrões

Uma solução frequentemente usada para um problema recorrente em que a solução é contraproducente, ineficaz ou menos eficaz do que uma alternativa.

### controle de aplicações

Uma abordagem de segurança que permite o uso somente de aplicações aprovadas para ajudar a proteger um sistema contra malware.

### portfólio de aplicações

Uma coleção de informações detalhadas sobre cada aplicação usada por uma organização, incluindo o custo para criar e manter a aplicação e seu valor comercial. Essas informações são fundamentais para [o processo de descoberta e análise de portfólio](#) e ajudam a identificar e priorizar as aplicações a serem migradas, modernizadas e otimizadas.

### inteligência artificial (IA)

O campo da ciência da computação que se dedica ao uso de tecnologias de computação para desempenhar funções cognitivas normalmente associadas aos humanos, como aprender, resolver problemas e reconhecer padrões. Para obter mais informações, consulte [O que é inteligência artificial?](#)

### operações de inteligência artificial (AIOps)

O processo de usar técnicas de machine learning para resolver problemas operacionais, reduzir incidentes operacionais e intervenção humana e aumentar a qualidade do serviço. Para obter mais informações sobre como AIOps é usado na estratégia de AWS migração, consulte o [guia de integração de operações](#).

---

## criptografia assimétrica

Um algoritmo de criptografia que usa um par de chaves, uma chave pública para criptografia e uma chave privada para descryptografia. É possível compartilhar a chave pública porque ela não é usada na descryptografia, mas o acesso à chave privada deve ser altamente restrito.

## atomicidade, consistência, isolamento, durabilidade (ACID)

Um conjunto de propriedades de software que garantem a validade dos dados e a confiabilidade operacional de um banco de dados, mesmo no caso de erros, falhas de energia ou outros problemas.

## controle de acesso por atributo (ABAC)

A prática de criar permissões minuciosas com base nos atributos do usuário, como departamento, cargo e nome da equipe. Para obter mais informações, consulte [ABAC AWS](#) na documentação AWS Identity and Access Management (IAM).

## fonte de dados autorizada

Um local onde você armazena a versão principal dos dados, que é considerada a fonte de informações mais confiável. Você pode copiar dados da fonte de dados autorizada para outros locais com o objetivo de processar ou modificar os dados, como anonimizá-los, redigi-los ou pseudonimizá-los.

## Zona de disponibilidade

Um local distinto dentro de um Região da AWS que está isolado de falhas em outras zonas de disponibilidade e fornece conectividade de rede barata e de baixa latência a outras zonas de disponibilidade na mesma região.

## AWS Estrutura de adoção da nuvem (AWS CAF)

Uma estrutura de diretrizes e melhores práticas AWS para ajudar as organizações a desenvolver um plano eficiente e eficaz para migrar com sucesso para a nuvem. AWS O CAF organiza a orientação em seis áreas de foco chamadas perspectivas: negócios, pessoas, governança, plataforma, segurança e operações. As perspectivas de negócios, pessoas e governança têm como foco habilidades e processos de negócios; as perspectivas de plataforma, segurança e operações concentram-se em habilidades e processos técnicos. Por exemplo, a perspectiva das pessoas tem como alvo as partes interessadas que lidam com recursos humanos (RH), funções de pessoal e gerenciamento de pessoal. Nessa perspectiva, o AWS CAF fornece orientação para desenvolvimento, treinamento e comunicação de pessoas para ajudar a preparar a organização

---

para a adoção bem-sucedida da nuvem. Para obter mais informações, consulte o [site da AWS CAF](#) e o [whitepaper da AWS CAF](#).

## AWS Estrutura de qualificação da carga de trabalho (AWS WQF)

Uma ferramenta que avalia as cargas de trabalho de migração do banco de dados, recomenda estratégias de migração e fornece estimativas de trabalho. AWS O WQF está incluído com AWS Schema Conversion Tool (AWS SCT). Ela analisa esquemas de banco de dados e objetos de código, código de aplicações, dependências e características de performance, além de fornecer relatórios de avaliação.

## B

### bot malicioso

Um [bot](#) destinado a causar interrupção ou danos a indivíduos ou organizações.

### BCP

Veja [planejamento de continuidade de negócios](#)

### gráfico de comportamento

Uma visualização unificada e interativa do comportamento e das interações de recursos ao longo do tempo. É possível usar um gráfico de comportamento com o Amazon Detective para examinar tentativas de login malsucedidas, chamadas de API suspeitas e ações similares. Para obter mais informações, consulte [Dados em um gráfico de comportamento](#) na documentação do Detective.

### sistema big-endian

Um sistema que armazena o byte mais significativo antes. Veja também [endianness](#).

### classificação binária

Um processo que prevê um resultado binário (uma de duas classes possíveis). Por exemplo, seu modelo de ML pode precisar prever problemas como “Este e-mail é ou não é spam?” ou “Este produto é um livro ou um carro?”

### filtro de bloom

Uma estrutura de dados probabilística e eficiente em termos de memória que é usada para testar se um elemento é membro de um conjunto.

---

## blue/green deployment (implantação azul/verde)

Uma estratégia de implantação em que você cria dois ambientes separados, mas idênticos. Você executa a versão atual da aplicação em um ambiente (azul) e a nova versão da aplicação no outro ambiente (verde). Essa estratégia ajuda você a reverter rapidamente com o mínimo de impacto.

## bot

Uma aplicação de software que executa tarefas automatizadas na internet e simula a atividade ou interação humana. Alguns bots são úteis ou benéficos, como crawlers da web que indexam informações na internet. Outros bots, conhecidos como bots maliciosos, têm como objetivo causar interrupção ou danos a indivíduos ou organizações.

## botnet

Redes de [bots](#) infectadas por [malware](#) e sob o controle de uma única parte, conhecidas como bot herder ou operador de bots. Os botnets são o mecanismo mais conhecido para escalar bots e seu impacto.

## ramo

Uma área contida de um repositório de código. A primeira ramificação criada em um repositório é a ramificação principal. Você pode criar uma nova ramificação a partir de uma ramificação existente e, em seguida, desenvolver recursos ou corrigir bugs na nova ramificação. Uma ramificação que você cria para gerar um recurso é comumente chamada de ramificação de recurso. Quando o recurso estiver pronto para lançamento, você mesclará a ramificação do recurso de volta com a ramificação principal. Para obter mais informações, consulte [Sobre filiais](#) (GitHub documentação).

## Acesso de emergência

Em circunstâncias excepcionais e por meio de um processo aprovado, um meio rápido para um usuário obter acesso a um Conta da AWS que ele normalmente não tem permissão para acessar. Para obter mais informações, consulte o indicador [Implement break-glass procedures](#) nas orientações do AWS Well-Architected.

## estratégia brownfield

A infraestrutura existente em seu ambiente. Ao adotar uma estratégia brownfield para uma arquitetura de sistema, você desenvolve a arquitetura de acordo com as restrições dos sistemas e da infraestrutura atuais. Se estiver expandindo a infraestrutura existente, poderá combinar as estratégias brownfield e [greenfield](#).

---

## cache do buffer

A área da memória em que os dados acessados com mais frequência são armazenados.

## capacidade de negócios

O que uma empresa faz para gerar valor (por exemplo, vendas, atendimento ao cliente ou marketing). As arquiteturas de microsserviços e as decisões de desenvolvimento podem ser orientadas por recursos de negócios. Para obter mais informações, consulte a seção [Organizados de acordo com as capacidades de negócios](#) do whitepaper [Executar microsserviços containerizados na AWS](#).

## planejamento de continuidade de negócios (BCP)

Um plano que aborda o impacto potencial de um evento disruptivo, como uma migração em grande escala, nas operações e permite que uma empresa retome as operações rapidamente.

# C

## CAF

Veja [AWS Cloud Adoption Framework](#).

## implantação canário

O lançamento lento e incremental de uma versão para usuários finais. Quando estiver confiante, você implanta a nova versão e substitui a versão atual por completo.

## CCoE

Veja [Centro de Excelência da Nuvem](#).

## CDC

Veja [captura de dados de alteração](#).

## captura de dados de alterações (CDC)

O processo de rastrear alterações em uma fonte de dados, como uma tabela de banco de dados, e registrar metadados sobre a alteração. É possível usar o CDC para várias finalidades, como auditar ou replicar alterações em um sistema de destino para manter a sincronização.

---

## engenharia do caos

Introduzir intencionalmente falhas ou eventos disruptivos para testar a resiliência de um sistema. Você pode usar [AWS Fault Injection Service \(AWS FIS\)](#) para realizar experimentos que estressam suas AWS cargas de trabalho e avaliar sua resposta.

## CI/CD

Veja [integração e entrega contínuas](#).

## classificação

Um processo de categorização que ajuda a gerar previsões. Os modelos de ML para problemas de classificação predizem um valor discreto. Os valores discretos são sempre diferentes uns dos outros. Por exemplo, um modelo pode precisar avaliar se há ou não um carro em uma imagem.

## criptografia no lado do cliente

Criptografia de dados localmente, antes que o alvo os AWS service (Serviço da AWS) receba.

## Centro de excelência em nuvem (CCoE)

Uma equipe multidisciplinar que impulsiona os esforços de adoção da nuvem em toda a organização, incluindo o desenvolvimento de práticas recomendadas de nuvem, a mobilização de recursos, o estabelecimento de cronogramas de migração e a liderança da organização em transformações em grande escala. Para obter mais informações, consulte as [publicações CCoE](#) no blog de estratégia Nuvem AWS corporativa.

## computação em nuvem

A tecnologia de nuvem normalmente usada para armazenamento de dados remoto e gerenciamento de dispositivos de IoT. A computação em nuvem é normalmente conectada à tecnologia de [computação de borda](#).

## modelo operacional em nuvem

Em uma organização de TI, o modelo operacional usado para criar, amadurecer e otimizar um ou mais ambientes de nuvem. Para obter mais informações, consulte [Criar seu modelo operacional de nuvem](#).

## estágios de adoção da nuvem

As quatro fases pelas quais as organizações normalmente passam ao migrar para a Nuvem AWS:

- Projeto: executar alguns projetos relacionados à nuvem para fins de prova de conceito e aprendizado
- Fundação — Fazer investimentos fundamentais para escalar sua adoção da nuvem (por exemplo, criar uma landing zone, definir um CCo E, estabelecer um modelo de operações)
- Migração: migrar aplicações individuais
- Reinvenção: otimizar produtos e serviços e inovar na nuvem

Esses estágios foram definidos por Stephen Orban na postagem do blog [The Journey Toward Cloud-First & the Stages of Adoption](#) no blog de estratégia Nuvem AWS empresarial. Para obter informações sobre como eles se relacionam com a estratégia de AWS migração, consulte o [guia de preparação para migração](#).

## CMDB

Veja [banco de dados de gerenciamento de configuração](#).

## repositório de código

Um local onde o código-fonte e outros ativos, como documentação, amostras e scripts, são armazenados e atualizados por meio de processos de controle de versão. Os repositórios de nuvem comuns incluem o GitHub ou o Bitbucket Cloud. Cada versão do código é chamada de ramificação. Em uma estrutura de microsserviços, cada repositório é dedicado a uma única peça de funcionalidade. Um único pipeline de CI/CD pode usar vários repositórios.

## cache frio

Um cache de buffer que está vazio, não está bem preenchido ou contém dados obsoletos ou irrelevantes. Isso afeta a performance porque a instância do banco de dados deve ler da memória principal ou do disco, um processo que é mais lento do que a leitura do cache do buffer.

## dados frios

Dados que raramente são acessados e geralmente são históricos. Ao consultar esse tipo de dados, consultas lentas geralmente são aceitáveis. Mover esses dados para níveis ou classes de armazenamento de baixo desempenho e menos caros pode reduzir os custos.

## visão computacional (CV)

Um campo de [IA](#) que usa machine learning para analisar e extrair informações de formatos visuais, como vídeos e imagens digitais. Por exemplo, a Amazon SageMaker AI fornece algoritmos de processamento de imagem para CV.

---

## desvio de configuração

Em uma workload, uma alteração de configuração em relação ao estado esperado. Isso pode fazer com que a workload se torne incompatível e, normalmente, é gradual e não intencional.

## banco de dados de gerenciamento de configuração (CMDB)

Um repositório que armazena e gerencia informações sobre um banco de dados e seu ambiente de TI, incluindo componentes de hardware e software e suas configurações. Normalmente, os dados de um CMDB são usados no estágio de descoberta e análise do portfólio da migração.

## pacote de conformidade

Um conjunto de AWS Config regras e ações de remediação que você pode montar para personalizar suas verificações de conformidade e segurança. Você pode implantar um pacote de conformidade como uma entidade única em uma Conta da AWS região ou em uma organização usando um modelo YAML. Para obter mais informações, consulte [Pacotes de conformidade na documentação](#). AWS Config

## integração contínua e entrega contínua (CI/CD)

O processo de automatizar os estágios de origem, criação, teste, preparação e produção do processo de lançamento do software. CI/CD é comumente descrito como um pipeline. CI/CD pode ajudá-lo a automatizar processos, melhorar a produtividade, melhorar a qualidade do código e entregar com mais rapidez. Para obter mais informações, consulte [Benefícios da entrega contínua](#). CD também pode significar implantação contínua. Para obter mais informações, consulte [Entrega contínua versus implantação contínua](#).

## CV

Veja [visão computacional](#).

## D

### dados em repouso

Dados estacionários em sua rede, por exemplo, dados que estão em um armazenamento.

### classificação de dados

Um processo para identificar e categorizar os dados em sua rede com base em criticalidade e confidencialidade. É um componente crítico de qualquer estratégia de gerenciamento de riscos de

---

segurança cibernética, pois ajuda a determinar os controles adequados de proteção e retenção para os dados. A classificação de dados é um componente do pilar de segurança no AWS Well-Architected Framework. Para obter mais informações, consulte [Classificação de dados](#).

#### desvio de dados

Uma variação significativa entre os dados de produção e os dados usados para treinar um modelo de ML ou uma alteração significativa nos dados de entrada ao longo do tempo. O desvio de dados pode reduzir a qualidade geral, a precisão e a imparcialidade das previsões do modelo de ML.

#### dados em trânsito

Dados que estão se movendo ativamente pela sua rede, como entre os recursos da rede.

#### data mesh

Um framework de arquitetura que fornece propriedade de dados distribuída e descentralizada com gerenciamento e governança centralizados.

#### minimização de dados

O princípio de coletar e processar apenas os dados estritamente necessários. Praticar a minimização de dados no Nuvem AWS pode reduzir os riscos de privacidade, os custos e a pegada de carbono de sua análise.

#### perímetro de dados

Um conjunto de proteções preventivas em seu AWS ambiente que ajudam a garantir que somente identidades confiáveis acessem recursos confiáveis das redes esperadas. Para obter mais informações, consulte [Construindo um perímetro de dados em AWS](#)

#### pré-processamento de dados

A transformação de dados brutos em um formato que seja facilmente analisado por seu modelo de ML. O pré-processamento de dados pode significar a remoção de determinadas colunas ou linhas e o tratamento de valores ausentes, inconsistentes ou duplicados.

#### proveniência dos dados

O processo de rastrear a origem e o histórico dos dados ao longo de seu ciclo de vida, por exemplo, como os dados foram gerados, transmitidos e armazenados.

#### titular dos dados

Um indivíduo cujos dados estão sendo coletados e processados.

---

## data warehouse

Um sistema de gerenciamento de dados compatível com business intelligence, como analytics. Os data warehouses geralmente contêm grandes quantidades de dados históricos e geralmente são usados para consultas e análises.

## linguagem de definição de dados (DDL)

Instruções ou comandos para criar ou modificar a estrutura de tabelas e objetos em um banco de dados.

## linguagem de manipulação de dados (DML)

Instruções ou comandos para modificar (inserir, atualizar e excluir) informações em um banco de dados.

## DDL

Veja [linguagem de definição de banco de dados](#).

## deep ensemble

A combinação de vários modelos de aprendizado profundo para gerar previsões. Os deep ensembles podem ser usados para produzir uma previsão mais precisa ou para estimar a incerteza nas previsões.

## Aprendizado profundo

Um subcampo do ML que usa várias camadas de redes neurais artificiais para identificar o mapeamento entre os dados de entrada e as variáveis-alvo de interesse.

## defense-in-depth

Uma abordagem de segurança da informação na qual uma série de mecanismos e controles de segurança são cuidadosamente distribuídos por toda a rede de computadores para proteger a confidencialidade, a integridade e a disponibilidade da rede e dos dados nela contidos. Ao adotar essa estratégia AWS, você adiciona vários controles em diferentes camadas da AWS Organizations estrutura para ajudar a proteger os recursos. Por exemplo, uma defense-in-depth abordagem pode combinar autenticação multifatorial, segmentação de rede e criptografia.

## administrador delegado

Em AWS Organizations, um serviço compatível pode registrar uma conta de AWS membro para administrar as contas da organização e gerenciar as permissões desse serviço. Essa conta

é chamada de administrador delegado para esse serviço Para obter mais informações e uma lista de serviços compatíveis, consulte [Serviços que funcionam com o AWS Organizations](#) na documentação do AWS Organizations .

## implantação

O processo de criar uma aplicação, novos recursos ou correções de código disponíveis no ambiente de destino. A implantação envolve a implementação de mudanças em uma base de código e, em seguida, a criação e execução dessa base de código nos ambientes da aplicação

## ambiente de desenvolvimento

Veja [ambiente](#).

## controle detectivo

Um controle de segurança projetado para detectar, registrar e alertar após a ocorrência de um evento. Esses controles são uma segunda linha de defesa, alertando você sobre eventos de segurança que contornaram os controles preventivos em vigor. Para obter mais informações, consulte [Controles detectivos](#) em Como implementar controles de segurança na AWS.

## mapeamento do fluxo de valor de desenvolvimento (DVSM)

Um processo usado para identificar e priorizar restrições que afetam negativamente a velocidade e a qualidade em um ciclo de vida de desenvolvimento de software. O DVSM estende o processo de mapeamento do fluxo de valor originalmente projetado para práticas de manufatura enxuta. Ele se concentra nas etapas e equipes necessárias para criar e movimentar valor por meio do processo de desenvolvimento de software.

## gêmeo digital

Uma representação virtual de um sistema real, como um prédio, fábrica, equipamento industrial ou linha de produção. Os gêmeos digitais oferecem suporte à manutenção preditiva, ao monitoramento remoto e à otimização da produção.

## tabela de dimensões

Em um [esquema em estrela](#), uma tabela menor que contém atributos de dados sobre dados quantitativos em uma tabela de fatos. Os atributos da tabela de dimensões geralmente são campos de texto ou números discretos que se comportam como texto. Esses atributos normalmente são usados para restringir consultas, filtrar e rotular conjuntos de resultados.

---

## desastre

Um evento que impede que uma workload ou sistema cumpra seus objetivos de negócios em seu local principal de implantação. Esses eventos podem ser desastres naturais, falhas técnicas ou o resultado de ações humanas, como configuração incorreta não intencional ou ataque de malware.

### Recuperação de desastres (RD)

A estratégia e o processo que você usa para minimizar o tempo de inatividade e a perda de dados causados por um [desastre](#). Para obter mais informações, consulte [Recuperação de desastres de cargas de trabalho em AWS: Recuperação na nuvem no AWS Well-Architected Framework](#).

### DML

Veja [linguagem de manipulação de banco de dados](#).

### design orientado por domínio

Uma abordagem ao desenvolvimento de um sistema de software complexo conectando seus componentes aos domínios em evolução, ou principais metas de negócios, atendidos por cada componente. Esse conceito foi introduzido por Eric Evans em seu livro, Design orientado por domínio: lidando com a complexidade no coração do software (Boston: Addison-Wesley Professional, 2003). Para obter informações sobre como usar o design orientado por domínio com o padrão strangler fig, consulte [Modernizar incrementalmente os serviços web herdados do Microsoft ASP.NET \(ASMX\) usando contêineres e o Amazon API Gateway](#).

### DR

Veja [recuperação de desastres](#).

### Deteção da oscilação

Rastreamento de desvios de uma configuração de linha de base. Por exemplo, você pode usar AWS CloudFormation para [detectar desvios nos recursos do sistema](#) ou AWS Control Tower para [detectar mudanças em seu landing zone](#) que possam afetar a conformidade com os requisitos de governança.

### DVSM

Veja [mapeamento do fluxo de valor de desenvolvimento](#).

---

## E

### EDA

Veja [análise exploratória de dados](#).

### EDI

Veja [intercâmbio eletrônico de dados](#).

### computação de borda

A tecnologia que aumenta o poder computacional de dispositivos inteligentes nas bordas de uma rede de IoT. Quando comparada com a [computação em nuvem](#), a computação de borda pode reduzir a latência da comunicação e melhorar o tempo de resposta.

### intercâmbio eletrônico de dados (EDI)

A troca automatizada de documentos comerciais entre organizações. Para obter mais informações, consulte [O que é EDI \(Intercâmbio eletrônico de dados\)?](#).

### criptografia

Um processo de computação que transforma dados de texto simples, legíveis por humanos, em texto cifrado.

### chave de criptografia

Uma sequência criptográfica de bits aleatórios que é gerada por um algoritmo de criptografia. As chaves podem variar em tamanho, e cada chave foi projetada para ser imprevisível e exclusiva.

### endianismo

A ordem na qual os bytes são armazenados na memória do computador. Os sistemas big-endian armazenam o byte mais significativo antes. Os sistemas little-endian armazenam o byte menos significativo antes.

### endpoint

Veja [endpoint de serviço](#).

### serviço de endpoint

Um serviço que pode ser hospedado em uma nuvem privada virtual (VPC) para ser compartilhado com outros usuários. Você pode criar um serviço de endpoint com AWS PrivateLink e conceder permissões a outros diretores Contas da AWS ou a AWS Identity and Access Management (IAM).

Essas contas ou entidades principais podem se conectar ao serviço de endpoint de maneira privada criando endpoints da VPC de interface. Para obter mais informações, consulte [Criar um serviço de endpoint](#) na documentação do Amazon Virtual Private Cloud (Amazon VPC).

planejamento de recursos empresariais (ERP)

Um sistema que automatiza e gerencia os principais processos de negócios (como contabilidade, [MES](#) e gerenciamento de projetos) para uma empresa.

criptografia envelopada

O processo de criptografar uma chave de criptografia com outra chave de criptografia. Para obter mais informações, consulte [Criptografia de envelope](#) na documentação AWS Key Management Service (AWS KMS).

ambiente

Uma instância de uma aplicação em execução. Estes são tipos comuns de ambientes na computação em nuvem:

- ambiente de desenvolvimento: uma instância de uma aplicação em execução que está disponível somente para a equipe principal responsável pela manutenção da aplicação. Ambientes de desenvolvimento são usados para testar mudanças antes de promovê-las para ambientes superiores. Esse tipo de ambiente às vezes é chamado de ambiente de teste.
- ambientes inferiores: todos os ambientes de desenvolvimento para uma aplicação, como aqueles usados para compilações e testes iniciais.
- ambiente de produção: uma instância de uma aplicação em execução que os usuários finais podem acessar. Em um CI/CD pipeline, o ambiente de produção é o último ambiente de implantação.
- ambientes superiores: todos os ambientes que podem ser acessados por usuários que não sejam a equipe principal de desenvolvimento. Isso pode incluir um ambiente de produção, ambientes de pré-produção e ambientes para testes de aceitação do usuário.

epic

Em metodologias ágeis, categorias funcionais que ajudam a organizar e priorizar seu trabalho. Os epics fornecem uma descrição de alto nível dos requisitos e das tarefas de implementação. Por exemplo, os épicos de segurança AWS da CAF incluem gerenciamento de identidade e acesso, controles de detetive, segurança de infraestrutura, proteção de dados e resposta a incidentes. Para obter mais informações sobre epics na estratégia de migração da AWS, consulte o [guia de implementação do programa](#).

---

## ERP

Veja [planejamento de recursos empresariais](#).

### análise exploratória de dados (EDA)

O processo de analisar um conjunto de dados para entender suas principais características. Você coleta ou agrega dados e, em seguida, realiza investigações iniciais para encontrar padrões, detectar anomalias e verificar suposições. O EDA é realizado por meio do cálculo de estatísticas resumidas e da criação de visualizações de dados.

## F

### tabela de fatos

A tabela central em um [esquema em estrela](#). Ela armazena dados quantitativos sobre as operações comerciais. Normalmente, uma tabela de fatos contém dois tipos de colunas: as que contêm medidas e as que contêm uma chave externa para uma tabela de dimensões.

### Antecipar-se à falha

Uma filosofia que usa testes frequentes e incrementais para reduzir o ciclo de vida do desenvolvimento. É uma parte essencial de uma abordagem ágil.

### delimitação de isolamento contra falhas

No Nuvem AWS, um limite, como uma zona de disponibilidade, Região da AWS um plano de controle ou um plano de dados, que limita o efeito de uma falha e ajuda a melhorar a resiliência das cargas de trabalho. Para obter mais informações, consulte [AWS Fault Isolation Boundaries](#).

### ramificação de recursos

Veja [ramificação](#).

### recursos

Os dados de entrada usados para fazer uma previsão. Por exemplo, em um contexto de manufatura, os recursos podem ser imagens capturadas periodicamente na linha de fabricação.

### importância do recurso

O quanto um recurso é importante para as previsões de um modelo. Isso geralmente é expresso como uma pontuação numérica que pode ser calculada por meio de várias técnicas, como

---

Shapley Additive Explanations (SHAP) e gradientes integrados. Para obter mais informações, consulte [Interpretabilidade do modelo de aprendizado de máquina com AWS](#).

## transformação de recursos

O processo de otimizar dados para o processo de ML, incluindo enriquecer dados com fontes adicionais, escalar valores ou extrair vários conjuntos de informações de um único campo de dados. Isso permite que o modelo de ML se beneficie dos dados. Por exemplo, se a data “2021-05-27 00:15:37” for dividida em “2021”, “maio”, “quinta” e “15”, isso poderá ajudar o algoritmo de aprendizado a aprender padrões diferenciados associados a diferentes componentes de dados.

## prompt few shot

Fornecer a um [LLM](#) um pequeno número de exemplos que demonstram a tarefa e o resultado desejado antes de solicitar que ele execute uma tarefa semelhante. Essa técnica é uma aplicação do aprendizado em contexto, em que os modelos aprendem com exemplos (shots) incorporados aos prompts. Prompts few-shot podem ser eficazes para tarefas que exigem formatação, raciocínio ou conhecimento de domínio específicos. Veja também [prompts zero-shot](#).

## FGAC

Veja [controle de acesso refinado](#).

## Controle de acesso refinado (FGAC)

O uso de várias condições para permitir ou negar uma solicitação de acesso.

## migração flash-cut

Um método de migração de banco de dados que usa replicação contínua de dados via [captura de dados de alteração](#) para migrar os dados no menor tempo possível, em vez de usar uma abordagem em fases. O objetivo é reduzir ao mínimo o tempo de inatividade.

## FM

Veja [modelo de base](#).

## modelo de base (FM)

Uma grande rede neural de aprendizado profundo que vem treinando em grandes conjuntos de dados generalizados e não rotulados. FMs são capazes de realizar uma ampla variedade de tarefas gerais, como entender a linguagem, gerar texto e imagens e conversar em linguagem natural. Para obter mais informações, consulte [O que são modelos de base?](#)

---

# G

## IA generativa

Um subconjunto de modelos de [IA](#) que foram treinados em grandes quantidades de dados e que podem usar um simples prompt de texto para criar novos artefatos e conteúdo, como imagens, vídeos, texto e áudio. Para obter mais informações, consulte [O que é IA generativa?](#).

## bloqueio geográfico

Veja [restrições geográficas](#).

## restrições geográficas (bloqueio geográfico)

Na Amazon CloudFront, uma opção para impedir que usuários em países específicos acessem distribuições de conteúdo. É possível usar uma lista de permissões ou uma lista de bloqueios para especificar países aprovados e banidos. Para obter mais informações, consulte [Restringir a distribuição geográfica do seu conteúdo](#) na CloudFront documentação.

## Fluxo de trabalho do GitFlow

Uma abordagem na qual ambientes inferiores e superiores usam ramificações diferentes em um repositório de código-fonte. O fluxo de trabalho do Gitflow é considerado legado, e o [fluxo de trabalho trunk-based](#) é a abordagem moderna e preferencial.

## golden image

Um snapshot de um sistema ou software usado como modelo para implantar novas instâncias desse sistema ou software. Por exemplo, na manufatura, uma golden image pode ser usada para provisionar software em vários dispositivos e ajudar a melhorar a velocidade, a escalabilidade e a produtividade nas operações de fabricação de dispositivos.

## estratégia greenfield

A ausência de infraestrutura existente em um novo ambiente. Ao adotar uma estratégia greenfield para uma arquitetura de sistema, é possível selecionar todas as novas tecnologias sem a restrição da compatibilidade com a infraestrutura existente, também conhecida como [brownfield](#). Se estiver expandindo a infraestrutura existente, poderá combinar as estratégias brownfield e greenfield.

## barreira de proteção

Uma regra de alto nível que ajuda a governar recursos, políticas e conformidade em todas as unidades organizacionais (OUs). Barreiras de proteção preventivas impõem políticas para

garantir o alinhamento a padrões de conformidade. Elas são implementadas usando políticas de controle de serviço e limites de permissões do IAM. Barreiras de proteção detectivas detectam violações de políticas e problemas de conformidade e geram alertas para remediação. Eles são implementados usando AWS Config, AWS Security Hub CSPM, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector e verificações personalizadas AWS Lambda .

## H

### HA

Veja [alta disponibilidade](#).

#### migração heterogênea de bancos de dados

Migrar seu banco de dados de origem para um banco de dados de destino que usa um mecanismo de banco de dados diferente (por exemplo, Oracle para Amazon Aurora). A migração heterogênea geralmente faz parte de um esforço de redefinição da arquitetura, e converter o esquema pode ser uma tarefa complexa. [O AWS fornece o AWS SCT](#) para ajudar nas conversões de esquemas.

#### alta disponibilidade (HA)

A capacidade de uma workload operar continuamente, sem intervenção, em caso de desafios ou desastres. Os sistemas AH são projetados para realizar o failover automático, oferecer consistentemente desempenho de alta qualidade e lidar com diferentes cargas e falhas com impacto mínimo no desempenho.

#### modernização de historiador

Uma abordagem usada para modernizar e atualizar os sistemas de tecnologia operacional (OT) para melhor atender às necessidades do setor de manufatura. Um historiador é um tipo de banco de dados usado para coletar e armazenar dados de várias fontes em uma fábrica.

#### dados de hold-out

Uma parte dos dados históricos rotulados que são retidos de um conjunto de dados usado para treinar um modelo de [machine learning](#). Você pode usar dados de hold-out para avaliar a performance do modelo comparando as predições do modelo com os dados de retenção.

---

## migração homogênea de bancos de dados

Migrar seu banco de dados de origem para um banco de dados de destino que compartilha o mesmo mecanismo de banco de dados (por exemplo, Microsoft SQL Server para Amazon RDS para SQL Server). A migração homogênea geralmente faz parte de um esforço de redefinição da hospedagem ou da plataforma. É possível usar utilitários de banco de dados nativos para migrar o esquema.

### dados quentes

Dados acessados com frequência, como dados em tempo real ou dados translacionais recentes. Esses dados normalmente exigem uma camada ou classe de armazenamento de alto desempenho para fornecer respostas rápidas às consultas.

### hotfix

Uma correção urgente para um problema crítico em um ambiente de produção. Devido à sua urgência, um hotfix geralmente é feito fora do fluxo de trabalho normal de DevOps lançamento.

### período de hipercuidados

Imediatamente após a substituição, o período em que uma equipe de migração gerencia e monitora as aplicações migradas na nuvem para resolver quaisquer problemas. Normalmente, a duração desse período é de 1 a 4 dias. No final do período de hipercuidados, a equipe de migração normalmente transfere a responsabilidade pelas aplicações para a equipe de operações de nuvem.

## eu

### laC

Veja [infraestrutura como código](#).

### Política baseada em identidade

Uma política anexada a um ou mais diretores do IAM que define suas permissões no Nuvem AWS ambiente.

### aplicação ociosa

Uma aplicação que tem um uso médio de CPU e memória entre 5 e 20% em um período de 90 dias. Em um projeto de migração, é comum retirar essas aplicações ou retê-las on-premises.

---

## IloT

Veja [Internet das Coisas Industrial](#).

### infraestrutura imutável

Um modelo que implanta uma nova infraestrutura para workloads de produção em vez de atualizar, aplicar patches ou modificar a infraestrutura existente. Infraestruturas imutáveis são inerentemente mais consistentes, confiáveis e preditivas do que [infraestruturas mutáveis](#). Para obter mais informações, consulte a prática recomendada [Implantar usando infraestrutura imutável](#) no AWS Well-Architected Framework.

### VPC de entrada (admissão)

Em uma arquitetura de AWS várias contas, uma VPC que aceita, inspeciona e roteia conexões de rede de fora de um aplicativo. A [Arquitetura de Referência de AWS Segurança](#) recomenda configurar sua conta de rede com entrada, saída e inspeção VPCs para proteger a interface bidirecional entre seu aplicativo e a Internet em geral.

### migração incremental

Uma estratégia de substituição na qual você migra a aplicação em pequenas partes, em vez de realizar uma única substituição completa. Por exemplo, é possível mover inicialmente apenas alguns microsserviços ou usuários para o novo sistema. Depois de verificar se tudo está funcionando corretamente, mova os microsserviços ou usuários adicionais de forma incremental até poder descomissionar seu sistema herdado. Essa estratégia reduz os riscos associados a migrações de grande porte.

### Indústria 4.0

Um termo que foi introduzido por [Klaus Schwab](#) em 2016 para se referir à modernização dos processos de manufatura por meio de avanços em conectividade, dados em tempo real, automação, analytics e IA/ML.

### infraestrutura

Todos os recursos e ativos contidos no ambiente de uma aplicação.

### Infraestrutura como código (IaC)

O processo de provisionamento e gerenciamento da infraestrutura de uma aplicação por meio de um conjunto de arquivos de configuração. A IaC foi projetada para ajudar você a centralizar o gerenciamento da infraestrutura, padronizar recursos e escalar rapidamente para que novos ambientes sejam reproduzíveis, confiáveis e consistentes.

---

## Internet industrial das coisas (IIoT)

O uso de sensores e dispositivos conectados à Internet nos setores industriais, como manufatura, energia, automotivo, saúde, ciências biológicas e agricultura. Para obter mais informações, consulte [Criando uma estratégia de transformação digital industrial da Internet das Coisas \(IIoT\)](#).

## VPC de inspeção

Em uma arquitetura de AWS várias contas, uma VPC centralizada que gerencia as inspeções do tráfego de rede entre VPCs (na mesma ou em diferentes Regiões da AWS) a Internet e as redes locais. A [Arquitetura de Referência de AWS Segurança](#) recomenda configurar sua conta de rede com entrada, saída e inspeção VPCs para proteger a interface bidirecional entre seu aplicativo e a Internet em geral.

## Internet das coisas (IoT)

A rede de objetos físicos conectados com sensores ou processadores incorporados que se comunicam com outros dispositivos e sistemas pela Internet ou por uma rede de comunicação local. Para obter mais informações, consulte [O que é IoT?](#)

## interpretabilidade

Uma característica de um modelo de machine learning que descreve o grau em que um ser humano pode entender como as previsões do modelo dependem de suas entradas. Para obter mais informações, consulte [Interpretabilidade do modelo de aprendizado de máquina com AWS](#).

## IoT

Veja [Internet das Coisas](#).

## Biblioteca de informações de TI (ITIL)

Um conjunto de práticas recomendadas para fornecer serviços de TI e alinhar esses serviços a requisitos de negócios. A ITIL fornece a base para o ITSM.

## Gerenciamento de serviços de TI (ITSM)

Atividades associadas a design, implementação, gerenciamento e suporte de serviços de TI para uma organização. Para obter informações sobre a integração de operações em nuvem com ferramentas de ITSM, consulte o [guia de integração de operações](#).

## ITIL

Veja [biblioteca de informações de TI](#).

---

## ITSM

Veja [gerenciamento de serviços de TI](#).

## L

### controle de acesso baseado em etiqueta (LBAC)

Uma implementação do controle de acesso obrigatório (MAC) em que os usuários e os dados em si recebem explicitamente um valor de etiqueta de segurança. A interseção entre a etiqueta de segurança do usuário e a etiqueta de segurança dos dados determina quais linhas e colunas podem ser vistas pelo usuário.

### zona de pouso

Uma landing zone é um AWS ambiente bem arquitetado, com várias contas, escalável e seguro. Um ponto a partir do qual suas organizações podem iniciar e implantar rapidamente workloads e aplicações com confiança em seu ambiente de segurança e infraestrutura. Para obter mais informações sobre zonas de pouso, consulte [Configurar um ambiente da AWS com várias contas seguro e escalável](#).

### grande modelo de linguagem (LLM)

Um modelo de [IA](#) de aprendizado profundo pré-treinado em uma grande quantidade de dados. Um LLM pode realizar várias tarefas, como responder a perguntas, resumir documentos, traduzir texto para outros idiomas e completar frases. Para obter mais informações, consulte [O que são LLMs](#).

### migração de grande porte

Uma migração de 300 servidores ou mais.

### LBAC

Veja [controle de acesso baseado em rótulo](#).

### privilégio mínimo

A prática recomendada de segurança de conceder as permissões mínimas necessárias para executar uma tarefa. Para obter mais informações, consulte [Aplicar permissões de privilégios mínimos](#) na documentação do IAM.

---

mover sem alterações (lift-and-shift)

Veja [7 Rs](#).

sistema little-endian

Um sistema que armazena o byte menos significativo antes. Veja também [endianness](#).

LLM

Veja [grande modelo de linguagem](#).

ambientes inferiores

Veja [ambiente](#).

## M

machine learning (ML)

Um tipo de inteligência artificial que usa algoritmos e técnicas para reconhecimento e aprendizado de padrões. O ML analisa e aprende com dados gravados, por exemplo, dados da Internet das Coisas (IoT), para gerar um modelo estatístico baseado em padrões. Para obter mais informações, consulte [Machine learning](#).

ramificação principal

Veja [ramificação](#).

Malware

Software projetado para comprometer a segurança ou a privacidade do computador. O malware pode interromper os sistemas do computador, vazar informações sensíveis ou obter acesso não autorizado. Exemplos de malware incluem vírus, worms, ransomware, cavalos de Troia, spyware e keyloggers.

Serviços gerenciados

Serviços da AWS para o qual AWS opera a camada de infraestrutura, o sistema operacional e as plataformas, e você acessa os endpoints para armazenar e recuperar dados. O Amazon Simple Storage Service (Amazon S3) e o Amazon DynamoDB são exemplos de serviços gerenciados. Eles também são conhecidos como serviços abstraídos.

---

## sistema de execução de manufatura (MES)

Um sistema de software para rastrear, monitorar, documentar e controlar processos de produção que convertem matérias-primas em produtos acabados no chão de fábrica.

### MAP

Veja [Programa de Aceleração da Migração](#).

### mecanismo

Um processo completo em que você cria uma ferramenta, impulsiona a adoção da ferramenta e, em seguida, inspeciona os resultados para fazer ajustes. Um mecanismo é um ciclo que se reforça e se aprimora à medida que opera. Para obter mais informações, consulte [Construindo mecanismos](#) no AWS Well-Architected Framework.

### conta de membro

Todos, Contas da AWS exceto a conta de gerenciamento, que fazem parte de uma organização em AWS Organizations. Uma conta só pode ser membro de uma organização de cada vez.

### MES

Veja [sistema de execução de manufatura](#).

### Transporte de Telemetria de Enfileiramento de Mensagens (MQTT)

[Um protocolo de comunicação leve machine-to-machine \(M2M\), baseado no padrão de publicação/assinatura, para dispositivos de IoT com recursos limitados.](#)

### microsserviço

Um serviço pequeno e independente que se comunica de forma bem definida APIs e normalmente é de propriedade de equipes pequenas e independentes. Por exemplo, um sistema de seguradora pode incluir microsserviços que mapeiam as capacidades comerciais, como vendas ou marketing, ou subdomínios, como compras, reclamações ou análises. Os benefícios dos microsserviços incluem agilidade, escalabilidade flexível, fácil implantação, código reutilizável e resiliência. Para obter mais informações, consulte [Integração de microsserviços usando serviços sem AWS servidor](#).

### arquitetura de microsserviços

Uma abordagem à criação de aplicações com componentes independentes que executam cada processo de aplicação como um microsserviço. Esses microsserviços se comunicam por meio

de uma interface bem definida usando leveza. APIs Cada microserviço nessa arquitetura pode ser atualizado, implantado e escalado para atender à demanda por funções específicas de uma aplicação. Para obter mais informações, consulte [Implementação de microserviços em AWS](#)

## Programa de Aceleração da Migração (MAP)

Um AWS programa que fornece suporte de consultoria, treinamento e serviços para ajudar as organizações a criar uma base operacional sólida para migrar para a nuvem e ajudar a compensar o custo inicial das migrações. O MAP inclui uma metodologia de migração para executar migrações legadas de forma metódica e um conjunto de ferramentas para automatizar e acelerar cenários comuns de migração.

## migração em escala

O processo de mover a maior parte do portfólio de aplicações para a nuvem em ondas, com mais aplicações sendo movidas em um ritmo mais rápido a cada onda. Essa fase usa as práticas recomendadas e lições aprendidas nas fases anteriores para implementar uma fábrica de migração de equipes, ferramentas e processos para agilizar a migração de workloads por meio de automação e entrega ágeis. Esta é a terceira fase da [estratégia de migração para a AWS](#).

## fábrica de migração

Equipes multifuncionais que simplificam a migração de workloads por meio de abordagens automatizadas e ágeis. As equipes da fábrica de migração geralmente incluem operações, analistas e proprietários de negócios, engenheiros de migração, desenvolvedores e DevOps profissionais que trabalham em sprints. Entre 20 e 50% de um portfólio de aplicações corporativas consiste em padrões repetidos que podem ser otimizados por meio de uma abordagem de fábrica. Para obter mais informações, consulte [discussão sobre fábricas de migração](#) e o [guia do Cloud Migration Factory](#) neste conjunto de conteúdo.

## metadados de migração

As informações sobre a aplicação e o servidor necessárias para concluir a migração. Cada padrão de migração exige um conjunto de metadados de migração diferente. Exemplos de metadados de migração incluem a sub-rede, o grupo de segurança e AWS a conta de destino.

## padrão de migração

Uma tarefa de migração repetível que detalha a estratégia de migração, o destino da migração e a aplicação ou o serviço de migração usado. Exemplo: rehoste a migração para o Amazon EC2 AWS com o Application Migration Service.

## Avaliação de Portfólio para Migração (MPA)

Uma ferramenta on-line que fornece informações para validar o caso de negócios para migrar para a Nuvem AWS. O MPA fornece avaliação detalhada do portfólio (dimensionamento correto do servidor, preços, comparações de TCO, análise de custos de migração), bem como planejamento de migração (análise e coleta de dados de aplicações, agrupamento de aplicações, priorização de migração e planejamento de ondas). A [ferramenta MPA](#) (requer login) está disponível gratuitamente para todos os AWS consultores e consultores parceiros da APN.

## Avaliação de Preparação para Migração (MRA)

O processo de obter insights sobre o status de prontidão de uma organização para a nuvem, identificar pontos fortes e fracos e criar um plano de ação para fechar as lacunas identificadas, usando o CAF. AWS Para mais informações, consulte o [guia de preparação para migração](#). A MRA é a primeira fase da [estratégia de migração para a AWS](#).

### estratégia de migração

A abordagem usada para migrar uma workload para a Nuvem AWS. Para obter mais informações, veja a entrada [7 Rs](#) neste glossário e consulte [Mobilize sua organização para acelerar migrações em grande escala](#).

### ML

Veja [machine learning](#).

### modernização

Transformar uma aplicação desatualizada (herdada ou monolítica) e sua infraestrutura em um sistema ágil, elástico e altamente disponível na nuvem para reduzir custos, ganhar eficiência e aproveitar as inovações. Para obter mais informações, consulte [Strategy for modernizing applications in the Nuvem AWS](#).

### avaliação de preparação para modernização

Uma avaliação que ajuda a determinar a preparação para modernização das aplicações de uma organização. Ela identifica benefícios, riscos e dependências e determina o quão bem a organização pode acomodar o estado futuro dessas aplicações. O resultado da avaliação é um esquema da arquitetura de destino, um roteiro que detalha as fases de desenvolvimento e os marcos do processo de modernização e um plano de ação para abordar as lacunas identificadas. Para obter mais informações, consulte [Evaluating modernization readiness for applications in the Nuvem AWS](#).

---

## aplicações monolíticas (monólitos)

Aplicações que são executadas como um único serviço com processos fortemente acoplados. As aplicações monolíticas apresentam várias desvantagens. Se um recurso da aplicação apresentar um aumento na demanda, toda a arquitetura deverá ser escalada. Adicionar ou melhorar os recursos de uma aplicação monolítica também se torna mais complexo quando a base de código cresce. Para resolver esses problemas, é possível criar uma arquitetura de microsserviços. Para obter mais informações, consulte [Decompor monólitos em microsserviços](#).

### MPA

Veja [Avaliação do Portfólio para Migração](#).

### MQTT

Veja [Transporte de Telemetria de Enfileiramento de Mensagens](#).

### classificação multiclasse

Um processo que ajuda a gerar previsões para várias classes (prevendo um ou mais de dois resultados). Por exemplo, um modelo de ML pode perguntar “Este produto é um livro, um carro ou um telefone?” ou “Qual categoria de produtos é mais interessante para este cliente?”

### infraestrutura mutável

Um modelo que atualiza e modifica a infraestrutura existente para workloads de produção. Para melhorar a consistência, confiabilidade e previsibilidade, o AWS Well-Architected Framework recomenda o uso de infraestrutura [imutável](#) como uma prática recomendada.

## O

### OAC

Veja [controle de acesso de origem](#).

### OAI

Veja [identidade de acesso de origem](#).

### OCM

Veja [gerenciamento de alterações organizacionais](#).

---

## migração offline

Um método de migração no qual a workload de origem é desativada durante o processo de migração. Esse método envolve tempo de inatividade prolongado e geralmente é usado para workloads pequenas e não críticas.

OI

Veja [integração de operações](#).

Ola

Veja [acordo de nível operacional](#).

## migração online

Um método de migração no qual a workload de origem é copiada para o sistema de destino sem ser colocada offline. As aplicações conectadas à workload podem continuar funcionando durante a migração. Esse método envolve um tempo de inatividade nulo ou mínimo e normalmente é usado para workloads essenciais para a produção.

OPC-UA

Veja [Open Process Communications - Unified Architecture](#).

Open Process Communications - Unified Architecture (OPC-UA)

Um protocolo de comunicação machine-to-machine (M2M) para automação industrial. O OPC-UA fornece um padrão de interoperabilidade com esquemas de criptografia, autenticação e autorização de dados.

acordo de nível operacional (OLA)

Um acordo que esclarece o que os grupos funcionais de TI prometem oferecer uns aos outros para apoiar um acordo de serviço (SLA).

análise de prontidão operacional (ORR)

Uma lista de verificação de perguntas e práticas recomendadas associadas que ajudam você a entender, avaliar, prevenir ou reduzir o escopo de incidentes e possíveis falhas. Para obter mais informações, consulte [Operational Readiness Reviews \(ORR\)](#) no AWS Well-Architected Framework.

tecnologia operacional (TO)

Sistemas de hardware e software que trabalham com o ambiente físico para controlar operações, equipamentos e infraestrutura industriais. Na manufatura, a integração dos sistemas de

---

tecnologia da informação (TI) e tecnologia operacional (TO) é o foco principal das transformações da [Indústria 4.0](#).

#### integração de operações (OI)

O processo de modernização das operações na nuvem, que envolve planejamento de preparação, automação e integração. Para obter mais informações, consulte o [guia de integração de operações](#).

#### trilha organizacional

Uma trilha criada por ela AWS CloudTrail registra todos os eventos de todas as Contas da AWS em uma organização em AWS Organizations. Essa trilha é criada em cada Conta da AWS que faz parte da organização e monitora a atividade em cada conta. Para obter mais informações, consulte [Criação de uma trilha para uma organização](#) na CloudTrail documentação.

#### gerenciamento de alterações organizacionais (OCM)

Uma estrutura para gerenciar grandes transformações de negócios disruptivas de uma perspectiva de pessoas, cultura e liderança. O OCM ajuda as organizações a se prepararem e fazerem a transição para novos sistemas e estratégias, acelerando a adoção de alterações, abordando questões de transição e promovendo mudanças culturais e organizacionais. Na estratégia de AWS migração, essa estrutura é chamada de aceleração de pessoas, devido à velocidade de mudança exigida nos projetos de adoção da nuvem. Para obter mais informações, consulte o [guia do OCM](#).

#### controle de acesso de origem (OAC)

Em CloudFront, uma opção aprimorada para restringir o acesso para proteger seu conteúdo do Amazon Simple Storage Service (Amazon S3). O OAC oferece suporte a todos os buckets S3 Regiões da AWS, criptografia do lado do servidor com AWS KMS (SSE-KMS) e solicitações dinâmicas ao bucket S3. PUT DELETE

#### Identidade do acesso de origem (OAI)

Em CloudFront, uma opção para restringir o acesso para proteger seu conteúdo do Amazon S3. Quando você usa o OAI, CloudFront cria um principal com o qual o Amazon S3 pode se autenticar. Os diretores autenticados podem acessar o conteúdo em um bucket do S3 somente por meio de uma distribuição específica. CloudFront Veja também [OAC](#), que fornece um controle de acesso mais granular e aprimorado.

#### ORR

Veja [análise de prontidão operacional](#).

---

## OT

Veja [tecnologia operacional](#).

### VPC de saída (egresso)

Em uma arquitetura de AWS várias contas, uma VPC que gerencia conexões de rede que são iniciadas de dentro de um aplicativo. A [Arquitetura de Referência de AWS Segurança](#) recomenda configurar sua conta de rede com entrada, saída e inspeção VPCs para proteger a interface bidirecional entre seu aplicativo e a Internet em geral.

## P

### limite de permissões

Uma política de gerenciamento do IAM anexada a entidades principais do IAM para definir as permissões máximas que o usuário ou perfil podem ter. Para obter mais informações, consulte [Limites de permissões](#) na documentação do IAM.

### Informações de identificação pessoal (PII)

Informações que, quando visualizadas diretamente ou combinadas com outros dados relacionados, podem ser usadas para inferir razoavelmente a identidade de um indivíduo. Exemplos de PII incluem nomes, endereços e informações de contato.

## PII

Veja [informações de identificação pessoal](#).

### manual

Um conjunto de etapas predefinidas que capturam o trabalho associado às migrações, como a entrega das principais funções operacionais na nuvem. Um manual pode assumir a forma de scripts, runbooks automatizados ou um resumo dos processos ou etapas necessários para operar seu ambiente modernizado.

## PLC

Veja [controlador lógico programável](#).

## PLM

Veja [gerenciamento do ciclo de vida do produto](#).

---

## política

Um objeto que pode definir permissões (veja [política baseada em identidade](#)), especificar condições de acesso (veja [política baseada em recurso](#)) ou definir as permissões máximas para todas as contas em uma organização no AWS Organizations (veja [política de controle de serviços](#)).

## persistência poliglota

Escolher de forma independente a tecnologia de armazenamento de dados de um microsserviço com base em padrões de acesso a dados e outros requisitos. Se seus microsserviços tiverem a mesma tecnologia de armazenamento de dados, eles poderão enfrentar desafios de implementação ou apresentar baixa performance. Os microsserviços serão implementados com mais facilidade e alcançarão performance e escalabilidade melhores se usarem o armazenamento de dados mais bem adaptado às suas necessidades.

## avaliação do portfólio

Um processo de descobrir, analisar e priorizar o portfólio de aplicações para planejar a migração. Para obter mais informações, consulte [Avaliar a preparação para a migração](#).

## predicado

Uma condição de consulta que retorna `true` ou `false`, normalmente localizada em uma cláusula `WHERE`.

## pushdown de predicados

Uma técnica de otimização de consultas de banco de dados que filtra os dados na consulta antes da transferência. Isso reduz a quantidade de dados que devem ser recuperados e processados do banco de dados relacional e melhora a performance das consultas.

## controle preventivo

Um controle de segurança projetado para evitar que um evento ocorra. Esses controles são a primeira linha de defesa para ajudar a evitar acesso não autorizado ou alterações indesejadas em sua rede. Para obter mais informações, consulte [Controles preventivos](#) em Como implementar controles de segurança na AWS.

## principal (entidade principal)

Uma entidade AWS que pode realizar ações e acessar recursos. Essa entidade geralmente é um usuário raiz para um Conta da AWS, uma função do IAM ou um usuário. Para obter mais

---

informações, consulte Entidade principal em [Termos e conceitos de perfis](#) na documentação do IAM.

## Privacidade por design

Uma abordagem em engenharia de sistemas que leva em consideração a privacidade em todo o processo de desenvolvimento.

## zonas hospedadas privadas

Um contêiner que contém informações sobre como você deseja que o Amazon Route 53 responda às consultas de DNS para um domínio e seus subdomínios em um ou mais VPCs. Para obter mais informações, consulte [Como trabalhar com zonas hospedadas privadas](#) na documentação do Route 53.

## controle proativo

Um [controle de segurança](#) desenvolvido para evitar a implantação de recursos não conformes. Esses controles verificam os recursos antes de serem provisionados. Se o recurso não estiver em conformidade com o controle, ele não será provisionado. Para obter mais informações, consulte o [guia de referência de controles](#) na AWS Control Tower documentação e consulte [Controles proativos](#) em Implementação de controles de segurança em AWS.

## gerenciamento do ciclo de vida do produto (PLM)

O gerenciamento de dados e processos de um produto em todo o seu ciclo de vida, desde a concepção, o desenvolvimento e o lançamento, passando pelo crescimento e maturidade, até o declínio e a remoção.

## ambiente de produção

Veja [ambiente](#).

## controlador lógico programável (PLC)

Na manufatura, um computador altamente confiável e adaptável que monitora as máquinas e automatiza os processos de fabricação.

## encadeamento de prompts

Uso da saída de um prompt do [LLM](#) como entrada para o próximo prompt para gerar respostas melhores. Essa técnica é usada para dividir uma tarefa complexa em subtarefas, ou para refinar ou expandir iterativamente uma resposta preliminar. Isso ajuda a melhorar a precisão e a relevância das respostas de um modelo e permite resultados mais granulares e personalizados.

---

## pseudonimização

O processo de substituir identificadores pessoais em um conjunto de dados por valores de espaço reservado. A pseudonimização pode ajudar a proteger a privacidade pessoal. Os dados pseudonimizados ainda são considerados dados pessoais.

## publish/subscribe (pub/sub)

Um padrão que permite comunicações assíncronas entre microsserviços para melhorar a escalabilidade e a capacidade de resposta. Por exemplo, em um [MES](#) baseado em microsserviços, um microsserviço pode publicar mensagens de eventos em um canal em que outros microsserviços possam assinar. O sistema pode adicionar novos microsserviços sem alterar o serviço de publicação.

## Q

### plano de consulta

Uma série de etapas, como instruções, usadas para acessar os dados em um sistema de banco de dados relacional SQL.

### regressão de planos de consultas

Quando um otimizador de serviço de banco de dados escolhe um plano menos adequado do que escolhia antes de uma determinada alteração no ambiente de banco de dados ocorrer. Isso pode ser causado por alterações em estatísticas, restrições, configurações do ambiente, associações de parâmetros de consulta e atualizações do mecanismo de banco de dados.

## R

### Matriz RACI

Veja [responsável, aprovador, consultado, informado \(RACI\)](#).

### RAG

Veja [geração aumentada via recuperação](#).

### ransomware

Um software mal-intencionado desenvolvido para bloquear o acesso a um sistema ou dados de computador até que um pagamento seja feito.

---

## Matriz RASCI

Veja [responsável, aprovador, consultado, informado \(RACI\)](#).

## RCAC

Veja [controle de acesso por linha e coluna](#).

## réplica de leitura

Uma cópia de um banco de dados usada somente para leitura. É possível encaminhar consultas para a réplica de leitura e reduzir a carga no banco de dados principal.

## Redefinir arquitetura

Veja [7 Rs](#).

## objetivo de ponto de recuperação (RPO).

O máximo período de tempo aceitável desde o último ponto de recuperação de dados. Isso determina o que é considerado uma perda aceitável de dados entre o último ponto de recuperação e a interrupção do serviço.

## objetivo de tempo de recuperação (RTO)

O máximo atraso aceitável entre a interrupção e a restauração do serviço.

## refatorar

Veja [7 Rs](#).

## Região

Uma coleção de AWS recursos em uma área geográfica. Cada um Região da AWS é isolado e independente dos outros para fornecer tolerância a falhas, estabilidade e resiliência. Para obter informações, consulte [Specify which Regiões da AWS your account can use](#).

## regressão

Uma técnica de ML que prevê um valor numérico. Por exemplo, para resolver o problema de “Por qual preço esta casa será vendida?” um modelo de ML pode usar um modelo de regressão linear para prever o preço de venda de uma casa com base em fatos conhecidos sobre a casa (por exemplo, a metragem quadrada).

## redefinir a hospedagem

Veja [7 Rs](#).

---

## versão

Em um processo de implantação, o ato de promover mudanças em um ambiente de produção.

realocar

Veja [7 Rs](#).

redefinir a plataforma

Veja [7 Rs](#).

recomprar

Veja [7 Rs](#).

resiliência

A capacidade de uma aplicação de resistir ou se recuperar de interrupções. [Alta disponibilidade](#) e [recuperação de desastres](#) são considerações comuns ao planejar a resiliência na Nuvem AWS. Para obter mais informações, consulte [Nuvem AWS Resilience](#).

política baseada em recurso

Uma política associada a um recurso, como um bucket do Amazon S3, um endpoint ou uma chave de criptografia. Esse tipo de política especifica quais entidades principais têm acesso permitido, ações válidas e quaisquer outras condições que devem ser atendidas.

matriz responsável, accountable, consultada, informada (RACI)

Uma matriz que define as funções e responsabilidades de todas as partes envolvidas nas atividades de migração e nas operações de nuvem. O nome da matriz é derivado dos tipos de responsabilidade definidos na matriz: responsável (R), responsabilizável (A), consultado (C) e informado (I). O tipo de suporte (S) é opcional. Se você incluir suporte, a matriz será chamada de matriz RASCI e, se excluir, será chamada de matriz RACI.

controle responsivo

Um controle de segurança desenvolvido para conduzir a remediação de eventos adversos ou desvios em relação à linha de base de segurança. Para obter mais informações, consulte [Controles responsivos](#) em Como implementar controles de segurança na AWS.

reter

Veja [7 Rs](#).

---

## Retirada

Veja [7 Rs](#).

### Geração Aumentada de Recuperação (RAG)

Uma tecnologia de [IA generativa](#) em que um [LLM](#) faz referência a uma fonte de dados autorizada que está fora de suas fontes de dados de treinamento antes de gerar uma resposta. Por exemplo, um modelo RAG pode realizar uma pesquisa semântica na base de conhecimento ou nos dados personalizados de uma organização. Para obter mais informações, consulte [O que é RAG \(geração aumentada via recuperação\)?](#).

### alternância

O processo de atualizar periodicamente um [segredo](#) para dificultar o acesso de um invasor às credenciais.

### controle de acesso por linha e coluna (RCAC)

O uso de expressões SQL básicas e flexíveis que tenham regras de acesso definidas. O RCAC consiste em permissões de linha e máscaras de coluna.

### RPO

Veja [objetivo de ponto de recuperação](#).

### RTO

Veja [objetivo de tempo de recuperação](#).

### runbook

Um conjunto de procedimentos manuais ou automatizados necessários para realizar uma tarefa específica. Eles são normalmente criados para agilizar operações ou procedimentos repetitivos com altas taxas de erro.

## S

### SAML 2.0

Um padrão aberto que muitos provedores de identidade (IdPs) usam. Esse recurso permite o login único federado (SSO), para que os usuários possam fazer login Console de gerenciamento da AWS ou chamar as operações da AWS API sem que você precise criar um usuário no IAM

---

para todos em sua organização. Para obter mais informações sobre a federação baseada em SAML 2.0, consulte [Sobre a federação baseada em SAML 2.0](#) na documentação do IAM.

## SCADA

Veja [controle de supervisão e aquisição de dados](#).

## SCP

Veja [política de controle de serviço](#).

## secret

Em AWS Secrets Manager, informações confidenciais ou restritas, como uma senha ou credenciais de usuário, que você armazena de forma criptografada. Consiste no valor secreto e em seus metadados. O valor secreto pode ser binário, uma única string ou várias strings. Para obter mais informações, consulte [What's in a Secrets Manager secret?](#) na documentação do Secrets Manager.

## segurança desde a concepção

Uma abordagem em engenharia de sistemas que leva em consideração a segurança em todo o processo de desenvolvimento.

## controle de segurança

Uma barreira de proteção técnica ou administrativa que impede, detecta ou reduz a capacidade de uma ameaça explorar uma vulnerabilidade de segurança. Existem quatro tipos primários de controles de segurança: [preventivos](#), [detectivos](#), [responsivos](#) e [proativos](#).

## hardening da segurança

O processo de reduzir a superfície de ataque para torná-la mais resistente a ataques. Isso pode incluir ações como remover recursos que não são mais necessários, implementar a prática recomendada de segurança de conceder privilégios mínimos ou desativar recursos desnecessários em arquivos de configuração.

## sistema de gerenciamento de eventos e informações de segurança (SIEM)

Ferramentas e serviços que combinam sistemas de gerenciamento de informações de segurança (SIM) e gerenciamento de eventos de segurança (SEM). Um sistema SIEM coleta, monitora e analisa dados de servidores, redes, dispositivos e outras fontes para detectar ameaças e violações de segurança e gerar alertas.

---

## automação de resposta de segurança

Uma ação predefinida e programada projetada para responder ou remediar automaticamente um evento de segurança. Essas automações servem como controles de segurança [responsivos](#) ou [detectivos](#) que ajudam você a implementar as melhores práticas AWS de segurança. Exemplos de ações de resposta automatizada incluem a modificação de um grupo de segurança da VPC, a aplicação de patches em uma instância do Amazon EC2 ou a alternância de credenciais.

## Criptografia do lado do servidor

Criptografia dos dados em seu destino, por AWS service (Serviço da AWS) quem os recebe.

## política de controle de serviços (SCP)

Uma política que fornece controle centralizado sobre as permissões de todas as contas em uma organização em AWS Organizations. SCPs defina barreiras ou estabeleça limites nas ações que um administrador pode delegar a usuários ou funções. Você pode usar SCPs como listas de permissão ou listas de negação para especificar quais serviços ou ações são permitidos ou proibidos. Para obter mais informações, consulte [Políticas de controle de serviço](#) na AWS Organizations documentação.

## service endpoint (endpoint de serviço)

O URL do ponto de entrada para um AWS service (Serviço da AWS). Você pode usar o endpoint para se conectar programaticamente ao serviço de destino. Para obter mais informações, consulte [Endpoints do AWS service \(Serviço da AWS\)](#) na Referência geral da AWS.

## acordo de serviço (SLA)

Um acordo que esclarece o que uma equipe de TI promete fornecer aos clientes, como tempo de atividade e performance do serviço.

## indicador de nível de serviço (SLI)

Uma avaliação de um aspecto de performance de um serviço, como taxa de erro, disponibilidade ou throughput.

## objetivo de nível de serviço (SLO)

Uma métrica alvo que representa a integridade de um serviço, conforme avaliado por um [indicador de nível de serviço](#).

---

## modelo de responsabilidade compartilhada

Um modelo que descreve a responsabilidade com a qual você compartilha AWS pela segurança e conformidade na nuvem. AWS é responsável pela segurança da nuvem, enquanto você é responsável pela segurança na nuvem. Para obter mais informações, consulte o [Modelo de responsabilidade compartilhada](#).

## SIEM

Veja [sistema de gerenciamento de eventos e informações de segurança](#).

## ponto único de falha (SPOF)

Uma falha em um único componente crítico de uma aplicação que pode interromper o sistema.

## SLA

Veja [acordo de serviço](#).

## SLI

Veja [indicador de nível de serviço](#).

## SLO

Veja [objetivo de nível de serviço](#).

## split-and-seed modelo

Um padrão para escalar e acelerar projetos de modernização. À medida que novos recursos e lançamentos de produtos são definidos, a equipe principal se divide para criar novas equipes de produtos. Isso ajuda a escalar os recursos e os serviços da sua organização, melhora a produtividade do desenvolvedor e possibilita inovações rápidas. Para obter mais informações, consulte [Phased approach to modernizing applications in the Nuvem AWS](#).

## SPOF

Veja [ponto único de falha](#).

## esquema em estrela

Uma estrutura organizacional de banco de dados que usa uma grande tabela de fatos para armazenar dados transacionais ou medidos e usa uma ou mais tabelas dimensionais menores para armazenar atributos de dados. Essa estrutura foi projetada para ser usada em um [data warehouse](#) ou para fins de inteligência comercial.

---

## padrão strangler fig

Uma abordagem à modernização de sistemas monolíticos que consiste em reescrever e substituir incrementalmente a funcionalidade do sistema até que o sistema herdado possa ser desativado. Esse padrão usa a analogia de uma videira que cresce e se torna uma árvore estabelecida e, eventualmente, supera e substitui sua hospedeira. O padrão foi [apresentado por Martin Fowler](#) como forma de gerenciar riscos ao reescrever sistemas monolíticos. Para ver um exemplo de como aplicar esse padrão, consulte [Modernizar incrementalmente os serviços Web herdados do Microsoft ASP.NET \(ASMX\) usando contêineres e o Amazon API Gateway](#).

## sub-rede

Um intervalo de endereços IP na VPC. Cada sub-rede fica alocada em uma única zona de disponibilidade.

## controle supervisor e aquisição de dados (SCADA)

Na manufatura, um sistema que usa hardware e software para monitorar ativos físicos e operações de produção.

## symmetric encryption (criptografia simétrica)

Um algoritmo de criptografia que usa a mesma chave para criptografar e descriptografar dados.

## testes sintéticos

Testar um sistema de forma que simule as interações do usuário para detectar possíveis problemas ou monitorar a performance. Você pode usar o [Amazon CloudWatch Synthetics](#) para criar esses testes.

## prompt do sistema

Uma técnica para fornecer contexto, instruções ou orientações a um [LLM](#) a fim de direcionar seu comportamento. Os prompts do sistema ajudam a definir o contexto e a estabelecer regras para interações com os usuários.

# T

## tags

Pares de valores-chave que atuam como metadados para organizar seus recursos. AWS As tags podem ajudar você a gerenciar, identificar, organizar, pesquisar e filtrar recursos da . Para obter mais informações, consulte [Marcar seus recursos do AWS](#).

---

## variável-alvo

O valor que você está tentando prever no ML supervisionado. Ela também é conhecida como variável de resultado. Por exemplo, em uma configuração de fabricação, a variável-alvo pode ser um defeito do produto.

## lista de tarefas

Uma ferramenta usada para monitorar o progresso por meio de um runbook. Uma lista de tarefas contém uma visão geral do runbook e uma lista de tarefas gerais a serem concluídas. Para cada tarefa geral, ela inclui o tempo estimado necessário, o proprietário e o progresso.

## ambiente de teste

Veja [ambiente](#).

## treinamento

O processo de fornecer dados para que seu modelo de ML aprenda. Os dados de treinamento devem conter a resposta correta. O algoritmo de aprendizado descobre padrões nos dados de treinamento que mapeiam os atributos dos dados de entrada no destino (a resposta que você deseja prever). Ele gera um modelo de ML que captura esses padrões. Você pode usar o modelo de ML para obter previsões de novos dados cujo destino você não conhece.

## gateway de trânsito

Um hub de trânsito de rede que você pode usar para interconectar sua rede com VPCs a rede local. Para obter mais informações, consulte [O que é um gateway de trânsito](#) na AWS Transit Gateway documentação.

## fluxo de trabalho baseado em troncos

Uma abordagem na qual os desenvolvedores criam e testam recursos localmente em uma ramificação de recursos e, em seguida, mesclam essas alterações na ramificação principal. A ramificação principal é então criada para os ambientes de desenvolvimento, pré-produção e produção, sequencialmente.

## Acesso confiável

Conceder permissões a um serviço que você especifica para realizar tarefas em sua organização AWS Organizations e em suas contas em seu nome. O serviço confiável cria um perfil vinculado ao serviço em cada conta, quando esse perfil é necessário, para realizar tarefas de gerenciamento para você. Para obter mais informações, consulte [Usando AWS Organizations com outros AWS serviços](#) na AWS Organizations documentação.

---

## tuning (ajustar)

Alterar aspectos do processo de treinamento para melhorar a precisão do modelo de ML. Por exemplo, você pode treinar o modelo de ML gerando um conjunto de rótulos, adicionando rótulos e repetindo essas etapas várias vezes em configurações diferentes para otimizar o modelo.

## equipe de duas pizzas

Uma pequena DevOps equipe que você pode alimentar com duas pizzas. Uma equipe de duas pizzas garante a melhor oportunidade possível de colaboração no desenvolvimento de software.

## U

### incerteza

Um conceito que se refere a informações imprecisas, incompletas ou desconhecidas que podem minar a confiabilidade dos modelos preditivos de ML. Há dois tipos de incertezas: a incerteza epistêmica é causada por dados limitados e incompletos, enquanto a incerteza aleatória é causada pelo ruído e pela aleatoriedade inerentes aos dados.

### tarefas indiferenciadas

Também conhecido como trabalho pesado, trabalho necessário para criar e operar um aplicativo, mas que não fornece valor direto ao usuário final nem oferece vantagem competitiva. Exemplos de tarefas indiferenciadas incluem aquisição, manutenção e planejamento de capacidade.

### ambientes superiores

Veja [ambiente](#).

## V

### aspiração

Uma operação de manutenção de banco de dados que envolve limpeza após atualizações incrementais para recuperar armazenamento e melhorar a performance.

### controle de versões

Processos e ferramentas que rastreiam mudanças, como alterações no código-fonte em um repositório.

---

## emparelhamento da VPC

Uma conexão entre duas VPCs que permite rotear o tráfego usando endereços IP privados. Para ter mais informações, consulte [O que é emparelhamento de VPC?](#) na documentação da Amazon VPC.

## Vulnerabilidade

Uma falha de software ou hardware que compromete a segurança do sistema.

## W

### cache quente

Um cache de buffer que contém dados atuais e relevantes que são acessados com frequência. A instância do banco de dados pode ler do cache do buffer, o que é mais rápido do que ler da memória principal ou do disco.

### dados mornos

Dados acessados raramente. Ao consultar esse tipo de dados, consultas moderadamente lentas geralmente são aceitáveis.

### função de janela

Uma função SQL que executa um cálculo em um grupo de linhas que se relacionam de alguma forma com o registro atual. As funções de janela são úteis para processar tarefas, como calcular uma média móvel ou acessar o valor das linhas com base na posição relativa da linha atual.

### workload

Uma coleção de códigos e recursos que geram valor empresarial, como uma aplicação voltada para o cliente ou um processo de backend.

### workstreams

Grupos funcionais em um projeto de migração que são responsáveis por um conjunto específico de tarefas. Cada workstream é independente, mas oferece suporte aos outros workstreams do projeto. Por exemplo, o workstream de portfólio é responsável por priorizar aplicações, planejar ondas e coletar metadados de migração. O workstream de portfólio entrega esses ativos ao workstream de migração, que então migra os servidores e as aplicações.

---

## WORM

Veja [gravação única e várias leituras](#).

## WQF

Veja [AWS Workload Qualification Framework](#).

### gravação única e várias leituras (WORM)

Um modelo de armazenamento que grava dados uma única vez e evita que os dados sejam excluídos ou modificados. Os usuários autorizados podem ler os dados quantas vezes forem necessárias, mas não podem alterá-los. Essa infraestrutura de armazenamento de dados é considerada [imutável](#).

## Z

### exploração de dia zero

Um ataque, normalmente malware, que tira proveito de uma [vulnerabilidade zero-day](#).

### vulnerabilidade de dia zero

Uma falha ou vulnerabilidade não mitigada em um sistema de produção. Os agentes de ameaças podem usar esse tipo de vulnerabilidade para atacar o sistema. Os desenvolvedores frequentemente ficam cientes da vulnerabilidade como resultado do ataque.

### prompt zero shot

Fornecer a um [LLM](#) instruções para realizar uma tarefa, mas sem exemplos (shots) que possam ajudar a orientá-lo. O LLM deve usar seu conhecimento pré-treinado para lidar com a tarefa. A eficácia dos prompts zero-shot depende da complexidade da tarefa e da qualidade do prompt.

Veja também [prompts few-shot](#).

### aplicação zumbi

Uma aplicação que tem um uso médio de CPU e memória inferior a 5%. Em um projeto de migração, é comum retirar essas aplicações.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.