



Reformulando a plataforma de seus aplicativos de mainframe usando um banco de dados IBM Db2 for z/OS compartilhado para migração gradual

AWS Orientação prescritiva



AWS Orientação prescritiva: Reformulando a plataforma de seus aplicativos de mainframe usando um banco de dados IBM Db2 for z/OS compartilhado para migração gradual

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens de marcas da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

Table of Contents

Introdução	1
Resultados de negócios	2
AWS Mainframe Modernization	4
Redefinir plataformas	4
Refatoração automatizada	5
Benefícios da replataforma	5
O processo de transformação	7
Planejamento	8
Descoberta de aplicativos	8
Dependências de dados	8
Referência de capacidade	9
Planejamento de ondas	12
Desenvolvimento	12
Consistência do aplicativo	13
Arquitetura	14
Em execução	16
Confirmação em duas fases (2PC)	17
infraestrutura de tempo de execução	18
Teste	19
Ambiente de origem	19
Ambiente de destino	20
Análise	20
Testando seu aplicativo em AWS Mainframe Modernization	21
substituição	23
Provisão	23
Transmita ao vivo	23
Reversão	24
Concluir	24
Arquitetura	24
Práticas recomendadas	26
Latência de rede	26
Segurança	27
Governança de aplicativos	27
Elasticidade	28

Próximas etapas	29
Recursos	30
AWS documentação	30
Referências da Rocket Software	30
Referências da IBM	30
Ferramentas	30
AWS Padrões e guias de orientação prescritiva	30
Histórico do documento	31
Glossário	32
#	32
A	33
B	36
C	38
D	41
E	46
F	48
G	50
H	51
eu	52
L	55
M	56
O	60
P	63
Q	66
R	66
S	70
T	74
U	75
V	76
W	76
Z	77
.....	lxxix

Reformatando seus aplicativos de mainframe usando um banco de dados IBM Db2 for compartilhado para migração gradual z/OS

Luis Gustavo Dantas e André Botura, da Amazon Web Services (AWS)

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

No cenário em constante mudança da tecnologia corporativa, a modernização do mainframe tornou-se um requisito essencial para organizações que precisam permanecer competitivas e ágeis. Essa transformação não se trata apenas de substituir sistemas antigos por novos; é uma evolução estratégica que preenche a lacuna entre as bases robustas e confiáveis do passado e as possibilidades dinâmicas e inovadoras do futuro.

O mainframe, que já foi o líder indiscutível da computação corporativa, agora está em um momento decisivo. Seu poder de processamento e recursos de segurança inigualáveis o mantiveram relevante por décadas, mas as empresas atuais precisam de sistemas que possam se integrar perfeitamente aos serviços em nuvem, oferecer suporte a aplicativos móveis e aproveitar o poder da inteligência artificial e da análise de big data.

A modernização nem sempre exige uma migração completa dos mainframes. Algumas organizações estão optando por abordagens híbridas que aproveitam os pontos fortes dos ambientes de mainframe e nuvem. Essa estratégia permite que eles mantenham aplicativos legados críticos enquanto fazem a transição gradual para plataformas mais modernas. Essa transição tecnológica envolve mais do que apenas atualizações do sistema; requer a transformação da cultura e das habilidades organizacionais. À medida que as empresas se modernizam, elas investem em novas tecnologias e em sua força de trabalho, preenchendo as lacunas geracionais e promovendo o aprendizado e a inovação contínuos.

Este guia discute uma estratégia de migração gradual que equilibra os benefícios dos sistemas de mainframe com as vantagens das modernas tecnologias de nuvem. Essa abordagem de replataforma em fases migra primeiro a camada do aplicativo, mantendo a conectividade com seu z/OS banco de dados IBM Db2 for existente para simplificar o processo de transição e minimizar as interrupções em suas operações comerciais críticas enquanto você adota novos recursos na nuvem.

Este guia foi desenvolvido para tomadores de decisões técnicas e equipes de implementação envolvidas em iniciativas de modernização do mainframe. O público principal inclui arquitetos corporativos e de soluções, gerentes técnicos de projetos e líderes de programas de modernização que precisam entender os aspectos estratégicos e técnicos da replataforma do mainframe. O conteúdo é igualmente valioso para as equipes de implementação, incluindo desenvolvedores de aplicativos de mainframe AWS ou engenheiros de nuvem, administradores de banco de dados e DevOps engenheiros, responsáveis pela execução da implementação da modernização.

Resultados de negócios

As empresas têm muitos motivos convincentes para atualizar seus aplicativos antigos. Esse processo cria um senso de urgência em todos os setores. Quando os especialistas mais velhos se aposentam, eles deixam uma lacuna significativa de conhecimento, o que torna crucial modernizar os sistemas antes que essa experiência seja perdida. Além disso, as empresas são motivadas pela necessidade de reduzir custos, aumentar a agilidade e responder rapidamente às rápidas mudanças nas condições do mercado.

O impulso pela transformação digital é ainda mais intensificado pelas tecnologias emergentes e pela demanda por experiências aprimoradas do cliente. Esses fatores, combinados com os riscos associados à manutenção de sistemas complexos, estão levando as organizações a agir rapidamente na modernização de sua infraestrutura de TI.

A modernização do mainframe, em particular, apresenta um equilíbrio delicado. As empresas devem preservar a estabilidade e a segurança pelas quais os mainframes são conhecidos, ao mesmo tempo em que adotam a flexibilidade e a escalabilidade oferecidas pelas arquiteturas modernas. Esse processo envolve decisões complexas sobre quais aplicativos migrar, quais reescrever e quais manter no mainframe.

Os principais fatores para a modernização incluem agilidade e redução de custos:

- Agilidade e tempo de lançamento no mercado. Os sistemas modernos permitem processos de aquisição mais rápidos e respostas mais rápidas às mudanças nas demandas do mercado. A adoção DevOps e as SysOps práticas podem melhorar significativamente a produtividade e as velocidades de implantação.
- Redução de custos. A modernização geralmente leva à redução dos custos de infraestrutura por meio de:
 - Pay-as-you-go modelos, que alinham os custos com o uso real.

- Taxas de licenciamento reduzidas associadas a sistemas legados.
- Elasticidade aprimorada, que proporciona melhor alocação de recursos.
- Configurações ativas e de alta disponibilidade, que aprimoram a resiliência do sistema e otimizam a utilização de recursos.

Com base nesses fatores de negócios, a replataforma de aplicativos COBOL é considerada uma abordagem estratégica para a modernização. Você pode usar um banco de dados compartilhado para seguir um caminho de migração gradual que equilibre a necessidade de modernização com a necessidade de manter a continuidade dos negócios. Esse método permite que você aproveite os benefícios das arquiteturas modernas e, ao mesmo tempo, preserve a confiabilidade de seus aplicativos COBOL. Como resultado, você pode obter agilidade, eficiência de custos e inovação e, ao mesmo tempo, mitigar os riscos associados a transições abruptas em grande escala. A abordagem de banco de dados Db2 compartilhado descrita neste guia fornece uma ponte entre sistemas legados e plataformas modernas e permite um processo de modernização mais suave e controlado.

Neste guia:

- [AWS Mainframe Modernization](#)
- [O processo de transformação](#)
- [Práticas recomendadas](#)
- [Próximas etapas](#)
- [Recursos](#)
- [Histórico de documentos](#)

AWS Mainframe Modernization

Note

AWS Mainframe Modernization O serviço (experiência do Managed Runtime Environment) não está mais aberto a novos clientes. Para recursos semelhantes ao AWS Mainframe Modernization Serviço (experiência do Managed Runtime Environment), explore o AWS Mainframe Modernization Service (Experiência autogerenciada). Os clientes atuais podem continuar usando o serviço normalmente. Para obter mais informações, consulte [Mudança de disponibilidade do AWS Mainframe Modernization](#).

O [AWS Mainframe Modernization serviço](#) permite que você migre seus aplicativos de mainframe legados para um ambiente nativo em nuvem, preserve a lógica de negócios e os investimentos existentes, use ferramentas automatizadas e serviços gerenciados de tempo de execução, otimize o desempenho dos aplicativos e reduza os custos operacionais. Esse serviço simplifica o processo de modernização, para que você possa aproveitar o poder da nuvem e, ao mesmo tempo, manter o valor de seus principais sistemas de mainframe. AWS fornece duas abordagens principais para a modernização do mainframe: replataforma e refatoração automatizada.

Redefinir plataformas

AWS Mainframe Modernization A [replataforma com a Rocket Software](#) (antiga Micro Focus) fornece uma opção poderosa de replataforma para empresas que desejam migrar seus aplicativos de mainframe para a nuvem com o mínimo de interrupção. Essa solução permite que você recompile e execute seus PL/I aplicativos e COBOL existentes AWS sem exigir alterações significativas no código.

Os principais benefícios da solução AWS Replatform with Rocket Software incluem:

- Preservação da lógica de negócios e dos investimentos existentes
- Risco reduzido e menor tempo de lançamento no mercado
- Escalabilidade e desempenho aprimorados na infraestrutura AWS
- Acesso a ferramentas e práticas modernas de desenvolvimento

Você pode usar essa solução para manter suas linguagens de programação de mainframe conhecidas e, ao mesmo tempo, aproveitar a flexibilidade, a economia e a inovação do. Nuvem AWS

Refatoração automatizada

Para uma abordagem mais transformadora do que a replataforma, você pode usar o [AWS Blu Age](#), que oferece refatoração automatizada de aplicativos de mainframe em aplicativos nativos da nuvem baseados em Java. Essa solução ajuda você a modernizar seus sistemas legados de forma mais abrangente e a convertê-los em aplicativos que podem aproveitar ao máximo as tecnologias nativas da nuvem.

As principais vantagens do AWS Blu Age incluem:

- Conversão de código legado em aplicativos Java modernos e de fácil manutenção
- Transformação automatizada que reduz o esforço manual e possíveis erros
- Criação de aplicativos nativos da nuvem que são otimizados para Serviços da AWS
- Maior agilidade e integração mais fácil com tecnologias modernas

AWS O Blu Age ajuda você a migrar seus aplicativos e prepará-los para a nuvem, abrindo novas possibilidades de inovação e crescimento. Para obter mais informações sobre essa abordagem, consulte [Refatoração automática de aplicativos com o AWS Blu Age](#) na documentação. AWS Mainframe Modernization

Benefícios da replataforma

Este guia discute uma abordagem para reformatar aplicativos COBOL de mainframe em. AWS Essa abordagem visa modernizar os sistemas legados e, ao mesmo tempo, reter temporariamente o IBM Db2 para z/OS agilizar o processo de transição. Ao manter inicialmente a estrutura de banco de dados existente, você pode reduzir a complexidade e o risco durante a migração. Essa abordagem em fases ajuda você a se beneficiar da escalabilidade e da economia do, ao mesmo tempo em que Nuvem AWS preserva a integridade crítica dos dados. As vantagens da replataforma em fases incluem o seguinte:

- Modernização acelerada: a reformulação de plataformas e a refatoração normalmente exigem menos tempo e recursos em comparação com a recriação de um aplicativo legado na nuvem, porque não envolvem a reescrita de todo o aplicativo. Essa abordagem também oferece suporte a

uma transição mais gradual que permite que as organizações se modernizem em seu próprio ritmo e, ao mesmo tempo, se beneficiem imediatamente da escalabilidade e da relação custo-benefício do. Nuvem AWS

- **Mitigação de riscos:** a replataforma oferece várias vantagens em relação à refatoração para muitas organizações. As empresas podem manter suas PL/I bases de código e COBOL existentes, preservar anos de lógica de negócios e minimizar o risco associado a alterações de código em grande escala.
- **Continuidade de dados e migração em fases:** um benefício significativo da replataforma é a opção de inicialmente manter os dados no Db2 em seu formato de dados original z/OS . Essa estratégia evita a necessidade de processos de migração de dados imediatos, complexos e potencialmente arriscados. Ao manter os dados em seu ambiente original durante a fase inicial, você pode preservar a integridade dos dados, reduzir o tempo de inatividade e minimizar o risco de perda ou corrupção de dados durante o processo de modernização. Como segunda etapa, você pode planejar uma migração de dados controlada e em fases para bancos de dados nativos em nuvem que envolva testes e validação completos enquanto o aplicativo continua sendo executado no ambiente reformulado.
- **Flexibilidade e preparação para o futuro:** para empresas que têm investimentos significativos em habilidades e aplicativos de mainframe, a replataforma fornece um caminho pragmático para a modernização que equilibra inovação com continuidade. Ele oferece a flexibilidade de reter inicialmente estruturas de dados e métodos de acesso críticos, além de preparar o terreno para futuros esforços de modernização, incluindo eventual migração de dados para soluções totalmente nativas em nuvem.

As organizações podem seguir a abordagem de replataforma para se modernizar em seu próprio ritmo e atender às necessidades imediatas enquanto planejam metas de transformação digital de longo prazo. Essa abordagem também oferece às empresas a oportunidade de treinar sua equipe em serviços nativos da nuvem.

O processo de transformação

A modernização do mainframe é uma etapa fundamental para organizações que desejam aproveitar os benefícios da computação em nuvem e, ao mesmo tempo, preservar seus valiosos aplicativos legados. Essa transformação apresenta desafios significativos. Os aplicativos de mainframe geralmente são altamente acoplados e têm interdependências complexas que evoluíram ao longo de décadas de operação. Essa complexidade exige uma abordagem cuidadosa e metódica da modernização.

As organizações precisam passar pelas seguintes fases principais para uma transição bem-sucedida:

- **Planejamento:** essa fase envolve uma descoberta abrangente dos sistemas existentes e a priorização dos esforços de modernização. As organizações avaliam sua infraestrutura atual, identificam aplicativos essenciais e determinam quais sistemas precisam ser modernizados primeiro.
- **Construção:** Durante esse estágio, as organizações criam processos para migrar aplicativos e desenvolver novos sistemas e infraestrutura. Isso envolve projetar e implementar a arquitetura modernizada e compilar o código-fonte.
- **Em execução:** essa etapa consiste em criar os ambientes de tempo de execução para hospedar os aplicativos reformulados. Envolve a configuração do hardware, software e infraestrutura de nuvem necessários para dar suporte aos sistemas modernizados e garantir que eles possam operar com eficiência no novo ambiente.
- **Teste:** Essa fase inclui uma validação rigorosa dos sistemas modernizados para verificar se todos os requisitos funcionais e de desempenho foram atendidos. Testes extensivos são conduzidos para verificar a integridade dos dados, a compatibilidade do sistema e o desempenho geral do novo ambiente.
- **Transição:** a fase final se concentra na implementação de estratégias para uma transição suave e no controle da mudança do mainframe legado para o ambiente modernizado. Isso inclui um planejamento cuidadoso do cronograma de migração e dos planos de contingência para minimizar a interrupção das operações comerciais.

As seções a seguir discutem essas fases em detalhes:

- [Planejamento](#)

- [Edifício](#)
- [Em execução](#)
- [Teste](#)
- [Transição](#)

Planejamento

Para lidar com os requisitos do aplicativo antigo de mainframe de forma eficaz, as organizações geralmente começam com uma avaliação abrangente de seu ambiente de mainframe.

Descoberta de aplicativos

Uma ferramenta poderosa nessa fase inicial é o [Rocket Enterprise Analyzer](#), que fornece insights profundos sobre a estrutura, dependências e complexidade dos aplicativos de mainframe. Essa ferramenta ajuda você a determinar o escopo de seu esforço de modernização, os riscos potenciais e as oportunidades de otimização.

Um aspecto crucial a ser descoberto é a intrincada rede de dependências de dados nos sistemas de mainframe. Essas dependências geralmente estão ocultas sob camadas de código legado e podem impactar significativamente os esforços de modernização. Ao mapear como diferentes aplicativos e módulos interagem com várias fontes de dados, você pode entender melhor os efeitos potenciais de qualquer mudança que planeja implementar.

Dependências de dados

Uma avaliação completa das dependências de dados pode revelar informações críticas sobre fluxo de dados, qualidade e governança de dados em seu ambiente de mainframe. Esse conhecimento é inestimável ao planejar estratégias de migração de dados, garantir a integridade dos dados durante a modernização e identificar oportunidades de otimização de dados. Ao obter uma visão clara de seus dados, você pode tomar decisões mais informadas sobre quais abordagens de modernização serão mais eficazes e menos disruptivas para suas operações existentes.

Uma análise de cima para baixo que identifica o uso de tabelas por transações ou tarefas de linguagem de controle de tarefas (JCL) é fundamental para criar planejamento e priorização de ondas. Essa abordagem esclarece as relações entre os diferentes componentes de seus sistemas de mainframe e ajuda você a desenvolver uma abordagem estratégica e em fases para

a modernização. Ao identificar quais tabelas são acessadas com mais frequência e por quais processos, você pode priorizar seus esforços de modernização: você pode se concentrar primeiro nas áreas de alto impacto e garantir uma transição mais suave com o mínimo de interrupção nas operações comerciais críticas.

Além de usar o Rocket Enterprise Analyzer para descobrir dependências de dados, muitas organizações também usam suas próprias soluções personalizadas para obter insights mais profundos sobre seus ambientes de mainframe. Essas ferramentas internas geralmente exploram a riqueza de informações disponíveis no catálogo do IBM Db2 e nos registros do System Management Facility (SMF).

Referência de capacidade

Uma etapa no planejamento de seu projeto de replataforma de mainframe é reunir informações detalhadas sobre o consumo atual da carga de trabalho. Esses dados ajudarão você a prever e provisionar com precisão a capacidade inicial necessária em seu ambiente de nuvem de destino. Por exemplo, recomendamos que você colete dados de consumo de milhões de instruções por segundo (MIPS) por hora para transações on-line e transações em lote das tarefas IBM Customer Information Control System (CICS) ou Information Management System (IMS) e Job Control Language (JCL).

A IBM oferece uma variedade diversificada de [modelos de preços](#) para MIPS na computação de mainframe, e muitos desses modelos se concentram no pico de uso. Entre esses modelos baseados em picos, o mais comum é o pico contínuo de quatro horas.

Os custos do mainframe incluem cinco áreas principais que afetam significativamente as despesas gerais:

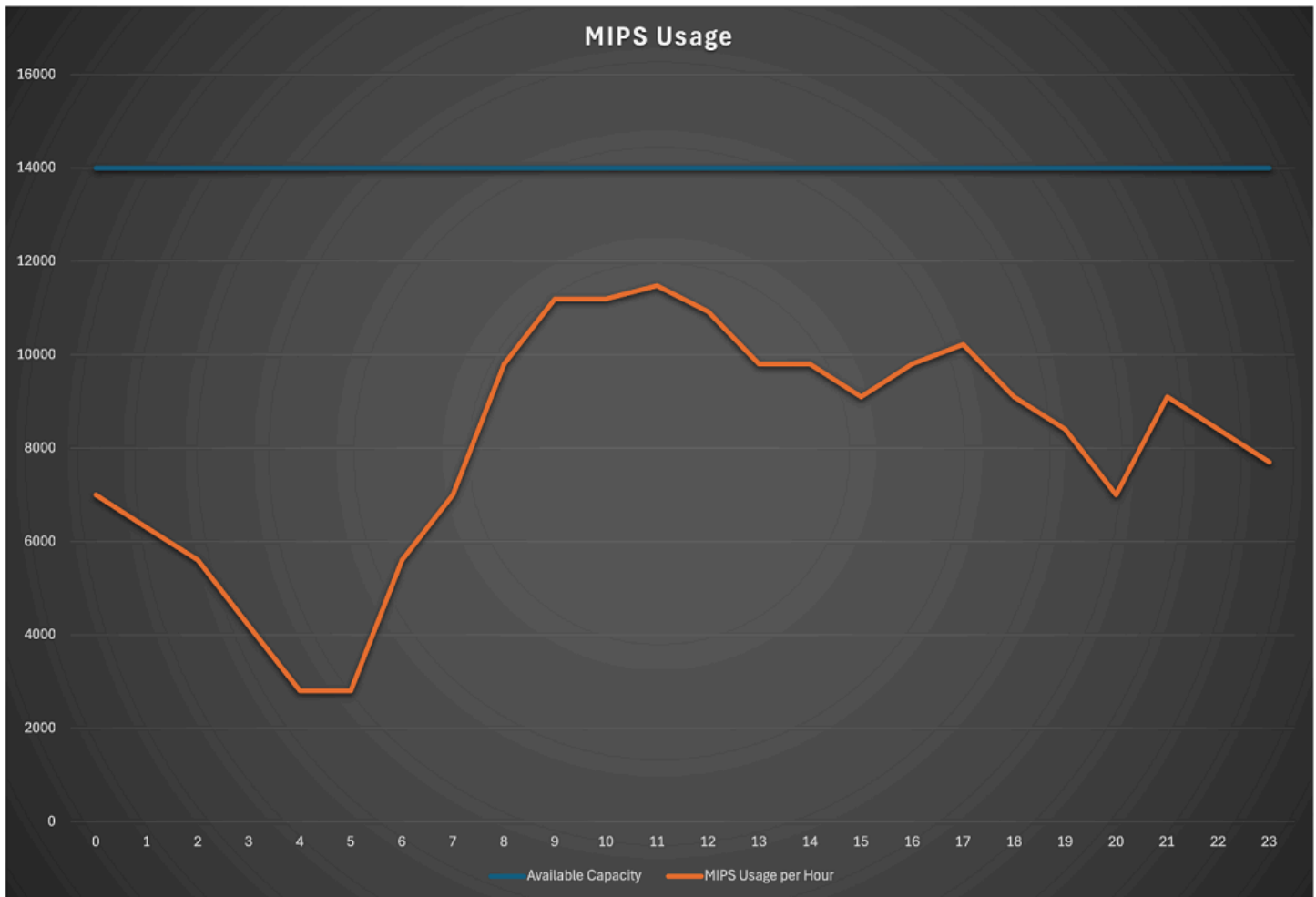
- O licenciamento de software geralmente é um componente importante. Ele abrange sistemas operacionais, middleware, bancos de dados e vários aplicativos, e os custos às vezes estão vinculados à capacidade ou ao uso da máquina.
- As despesas de hardware incluem a compra ou locação inicial de equipamentos de mainframe, manutenção contínua e atualizações.
- Os custos de armazenamento podem ser substanciais devido às grandes quantidades de dados gerenciados e envolvem sistemas de disco, bibliotecas de fitas e software de gerenciamento associado.
- As despesas com pessoal cobrem salários de profissionais especializados em mainframe, como programadores de sistemas e administradores de banco de dados.

- As medidas de recuperação de desastres e continuidade dos negócios, incluindo sistemas de backup, hardware redundante e instalações de recuperação externas, representam um investimento significativo para garantir alta disponibilidade e recuperação rápida.

Essas cinco categorias de custo, combinadas com cobranças baseadas em MIPS, formam o núcleo da maioria dos orçamentos de mainframe. No entanto, suas proporções relativas podem variar muito, dependendo do tamanho, do setor e dos padrões específicos de utilização do mainframe de sua organização.

Os dados MIPS de hora em hora são cruciais para obter uma compreensão abrangente dos padrões e do desempenho da carga de trabalho do seu mainframe. Diferentemente das médias diárias ou mensais, os dados por hora fornecem informações granulares que revelam as flutuações diferenciadas na utilização de recursos do sistema ao longo do dia. Esse nível de detalhe é inestimável para avaliar com precisão as necessidades de desempenho e capacidade do seu aplicativo na nuvem.

Ao analisar os dados MIPS de hora em hora, você pode identificar períodos de pico de uso, identificar tendências e identificar possíveis gargalos que podem estar ocultos nos dados agregados, conforme mostrado no diagrama a seguir. Essa granularidade permite um planejamento de capacidade mais preciso, ajuda a otimizar a alocação de recursos e pode potencialmente levar à economia de custos e à melhoria da eficiência do sistema.



Os dados MIPS de hora em hora também servem como uma ferramenta essencial de referência de desempenho. Ele estabelece uma linha de base detalhada do desempenho do seu sistema, o que é particularmente valioso quando você planeja ou avalia mudanças no sistema, como migrações ou atualizações. Ao comparar os dados MIPS horários antes e depois da alteração, você pode medir com precisão o impacto dessas modificações no desempenho do seu sistema e garantir que seu mainframe continue atendendo às necessidades da sua organização.

Para coletar dados MIPS de hora em hora, você tem várias opções. Uma abordagem é usar registros SMF diretamente. Esses registros fornecem uma grande quantidade de informações sobre a atividade do sistema e o uso de recursos. Como alternativa, você pode usar ferramentas especializadas, como a IBM Sub-Capacity Reporting Tool (SCRT), que pode simplificar o processo de coleta e análise de dados MIPS.

Independentemente do método escolhido, é importante coletar dados por um longo período — idealmente, vários meses. Esse período de coleta estendido permite que você contabilize as

variações cíclicas em sua carga de trabalho, como picos de end-of-month processamento ou flutuações sazonais. Ao capturar esses padrões de longo prazo, você pode desenvolver uma imagem mais precisa e abrangente das características de desempenho do seu mainframe, o que permite uma tomada de decisão mais bem informada e um gerenciamento de capacidade mais eficaz.

Planejamento de ondas

Você pode usar as informações coletadas para priorizar estrategicamente suas iniciativas de replataforma de mainframe. Uma abordagem prudente é começar com cargas de trabalho menos críticas, como transações comerciais não essenciais ou trabalhos em lotes, para permitir que as equipes ganhem experiência e refinem processos com risco mínimo para as operações essenciais. Além disso, considerar cargas de trabalho somente para leitura como candidatas iniciais à migração pode ser vantajoso, pois essas cargas de trabalho geralmente envolvem menos complexidade e menor risco de inconsistências de dados. Essa abordagem permite que você crie confiança e impulso em seus esforços de reformulação de plataforma.

Além disso, agrupar cargas de trabalho que compartilham tabelas do Db2 para operações de gravação ou atualização pode simplificar o processo de migração. Ao identificar essas cargas de trabalho interconectadas, você pode planejar ondas de migração coesas que mantenham a integridade dos dados e minimizem a necessidade de soluções provisórias complexas. Essa estratégia não apenas reduz o risco de conflitos de dados, mas também otimiza o cronograma geral de replataforma, abordando os componentes relacionados simultaneamente. Em última análise, essa abordagem de priorização baseada em dados garante uma consideração equilibrada de criticidade, complexidade e interdependência e leva a um processo de modernização do mainframe mais eficiente e bem-sucedido.

Desenvolvimento

O uso de um banco de dados Db2 compartilhado permite a execução simultânea de aplicativos idênticos ou consistentes em ambientes de mainframe e nuvem. Essa abordagem oferece várias vantagens quando você mantém a mesma versão do aplicativo em ambas as plataformas e fornece maior flexibilidade e confiabilidade em suas operações.

Uma das principais vantagens dessa estratégia é a capacidade de implementar um plano de reversão eficaz. Se surgirem problemas durante a migração ou a implantação, ter a mesma versão

do aplicativo permite uma reversão perfeita ao estado anterior e minimiza o tempo de inatividade e possíveis inconsistências de dados.

Consistência do aplicativo

Espelhar componentes do aplicativo de um gerenciador de controle de origem distribuído para o mainframe é uma abordagem estratégica durante o processo de replataforma. Esse método suporta o uso de ferramentas modernas de gerenciamento de código-fonte, mantendo a sincronização com o ambiente de mainframe. Esse processo de espelhamento é temporário e dura somente até que a carga de trabalho esteja totalmente funcional na produção na plataforma distribuída.

Ao migrar o código-fonte do seu aplicativo reformulado para uma ferramenta distribuída de gerenciamento de alterações, você pode aproveitar vários benefícios oferecidos pelos gerenciadores de código-fonte modernos. Isso inclui:

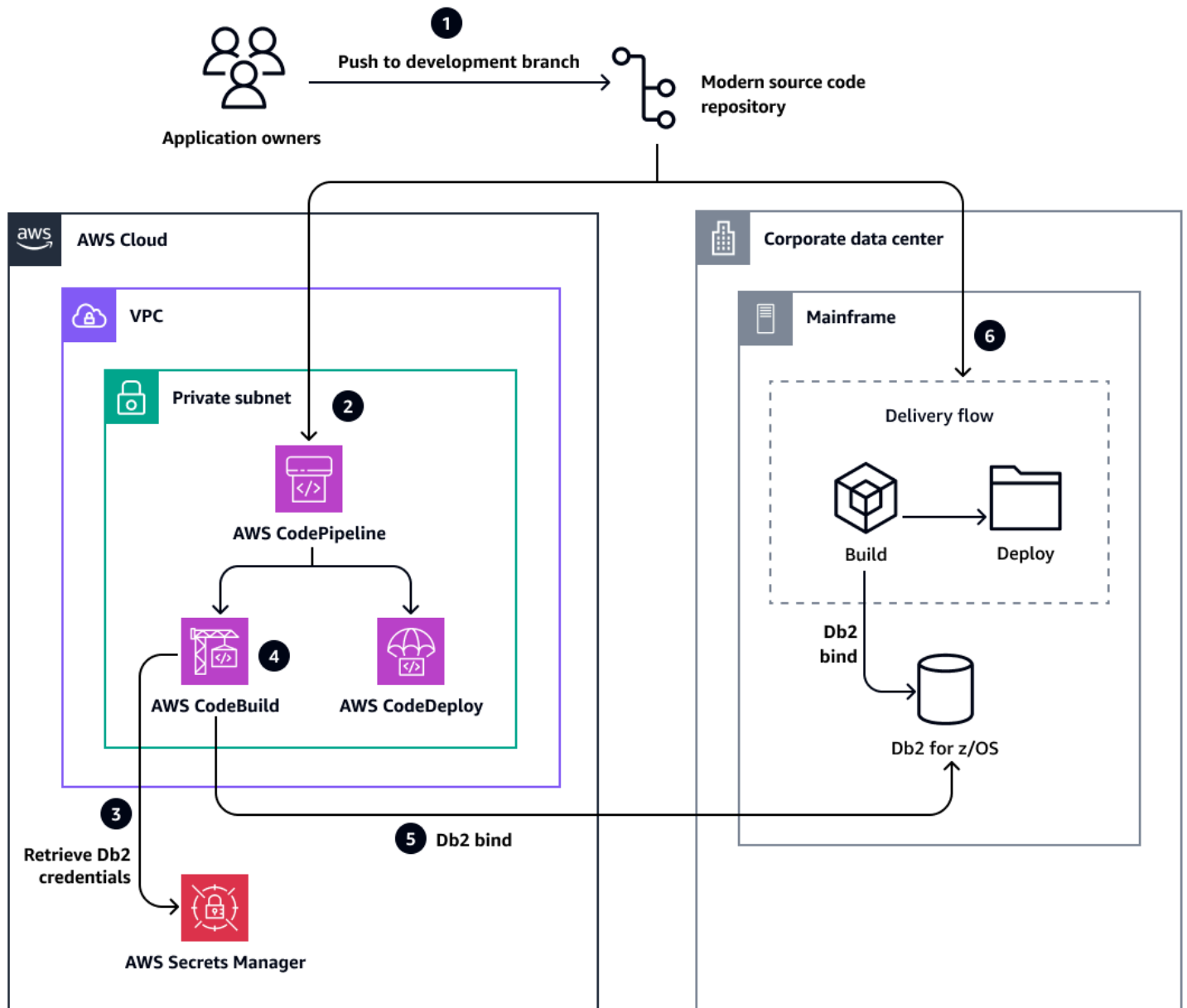
- Colaboração aprimorada: as ferramentas distribuídas geralmente oferecem melhor suporte à colaboração em equipe, incluindo recursos como pull requests, análises de código e estratégias de ramificação.
- Controle de versão aprimorado: os sistemas modernos oferecem um controle de versão mais granular e facilitam o rastreamento de alterações e o gerenciamento de diferentes versões do código.
- Integração com CI/CD pipelines: muitas ferramentas distribuídas se integram perfeitamente aos pipelines de integração contínua e implantação contínua (CI/CD), o que simplifica o processo de desenvolvimento.
- Melhor visibilidade e rastreabilidade: essas ferramentas geralmente fornecem painéis e recursos de geração de relatórios superiores e oferecem maior visão do processo de desenvolvimento.
- Support para práticas modernas de desenvolvimento: os sistemas distribuídos geralmente são mais adequados para metodologias e DevOps práticas ágeis.

O processo de espelhamento envolve a sincronização do código do gerenciador de controle de origem distribuído de volta ao mainframe. Isso garante que os dois ambientes permaneçam consistentes durante o período de transição. No entanto, você deve implementar o espelhamento como uma sincronização unidirecional, na qual as atualizações fluem do sistema distribuído para o mainframe, em vez de bidirecionalmente. Essa abordagem mantém a consistência e evita possíveis conflitos que possam surgir de atualizações simultâneas em ambos os ambientes.

Ao adotar essa estratégia de espelhamento, você pode gradualmente transferir seus esforços de desenvolvimento para a plataforma distribuída e, ao mesmo tempo, garantir que o ambiente de mainframe permaneça up-to-date. Isso proporciona uma transição mais suave e uma rede de segurança durante o processo de replataforma. Quando a carga de trabalho estiver totalmente funcional e estável no ambiente de produção distribuído, você poderá eliminar gradualmente o processo de espelhamento e concluir a migração para o sistema moderno de gerenciamento de código-fonte.

Arquitetura

O diagrama a seguir mostra como um sistema distribuído de gerenciamento de código-fonte pode espelhar os componentes do aplicativo e manter a sincronização entre os ambientes Nuvem AWS e o mainframe. O Nuvem AWS ambiente usa CI/CD serviços como [AWS CodeBuild](#), [AWS CodePipeline](#), e [AWS CodeDeploy](#) para criar e implantar o aplicativo.



Nesse fluxo de trabalho:

1. Os proprietários do aplicativo entregam uma nova versão do aplicativo na ramificação de desenvolvimento do repositório de código-fonte.
2. A nova versão é AWS CodePipeline acionada.
3. AWS CodeBuild recupera as credenciais do Db2 de [AWS Secrets Manager](#)
4. CodeBuild compila o aplicativo.
5. CodeBuild usa o Db2 z/OS para vincular o aplicativo.

6. O fluxo de entrega do mainframe também cria e implanta o aplicativo.

Em execução

Para garantir o desempenho ideal e a baixa latência entre seu aplicativo baseado em nuvem e seu banco de dados local, recomendamos que você implemente. [AWS Direct Connect](#) Esse serviço fornece uma conexão de rede dedicada entre AWS o data center da sua organização e oferece desempenho de rede mais consistente e latência reduzida em comparação com conexões baseadas na Internet. Isso é particularmente crucial para operações de banco de dados que exigem tempos de resposta rápidos.

Para obter alta disponibilidade (HA) e elasticidade para o aplicativo em execução AWS, você pode implementar uma arquitetura robusta usando os seguintes componentes:

- Elastic Load Balancing (ELB): você pode implantar um balanceador de carga para distribuir o tráfego de entrada em várias instâncias do Amazon Elastic Compute Cloud (Amazon EC2) nas quais seu aplicativo é executado. Isso garante uma distribuição uniforme da carga de trabalho e fornece um único ponto de entrada para as solicitações do cliente.
- Grupo de Auto Scaling: as instâncias do EC2 que hospedam o aplicativo podem ser organizadas em um grupo de Auto Scaling. Isso permite que a infraestrutura ajuste automaticamente o número de instâncias com base em métricas predefinidas, como utilização da CPU ou tráfego de rede. Durante os horários de pico, instâncias adicionais podem ser lançadas para lidar com o aumento da carga, enquanto em períodos mais silenciosos, instâncias desnecessárias podem ser encerradas para otimizar os custos.
- Instâncias EC2: o aplicativo pode ser implantado em instâncias EC2 dentro do grupo Auto Scaling. Essas instâncias devem ser distribuídas em várias zonas de disponibilidade para aumentar a tolerância a falhas e garantir alta disponibilidade.
- Implantação Multi-AZ: ao distribuir as instâncias do aplicativo em várias zonas de disponibilidade, o sistema pode resistir à falha de uma única zona de disponibilidade sem impacto significativo na disponibilidade geral.

Essa arquitetura permite que o aplicativo seja escalado perfeitamente com base na demanda, mantendo a alta disponibilidade. O balanceador de carga garante que o tráfego seja distribuído uniformemente entre instâncias íntegras, e o grupo Auto Scaling gerencia o número de instâncias com base na carga de trabalho real.

Para aumentar ainda mais a confiabilidade, você pode implementar um sistema robusto de monitoramento e alerta usando CloudWatch a [Amazon](#) para ajudar a detectar e responder prontamente a quaisquer problemas de desempenho ou falhas. Além disso, testes regulares dos recursos de escalonamento automático e dos cenários de failover garantirão que o sistema se comporte conforme o esperado durante várias condições de carga e possíveis falhas.

Ao adotar essa abordagem, você pode se beneficiar da escalabilidade e flexibilidade do e, Nuvem AWS ao mesmo tempo, manter uma conexão segura com seu banco de dados Db2 local. Essa configuração híbrida serve como um excelente caminho para uma migração completa para a nuvem e fornece transição gradual e mitigação de riscos durante todo o processo.

Confirmação em duas fases (2PC)

AWS Mainframe Modernization O [Replatform with Rocket Software](#) oferece suporte para transações de confirmação em duas fases (2PC) por meio da implementação de arquitetura estendida (XA). Esse recurso é crucial para manter a integridade dos dados em sistemas distribuídos, especialmente em ambientes de mainframe em que transações complexas geralmente abrangem vários recursos.

A arquitetura XA, que é integrada ao AWS Replatform with Rocket Software, permite a coordenação de transações em diversos recursos, como bancos de dados e filas de mensagens. Essa integração garante que todas as partes de uma transação distribuída sejam confirmadas ou revertidas em uníssono, para manter a consistência em todo o sistema.

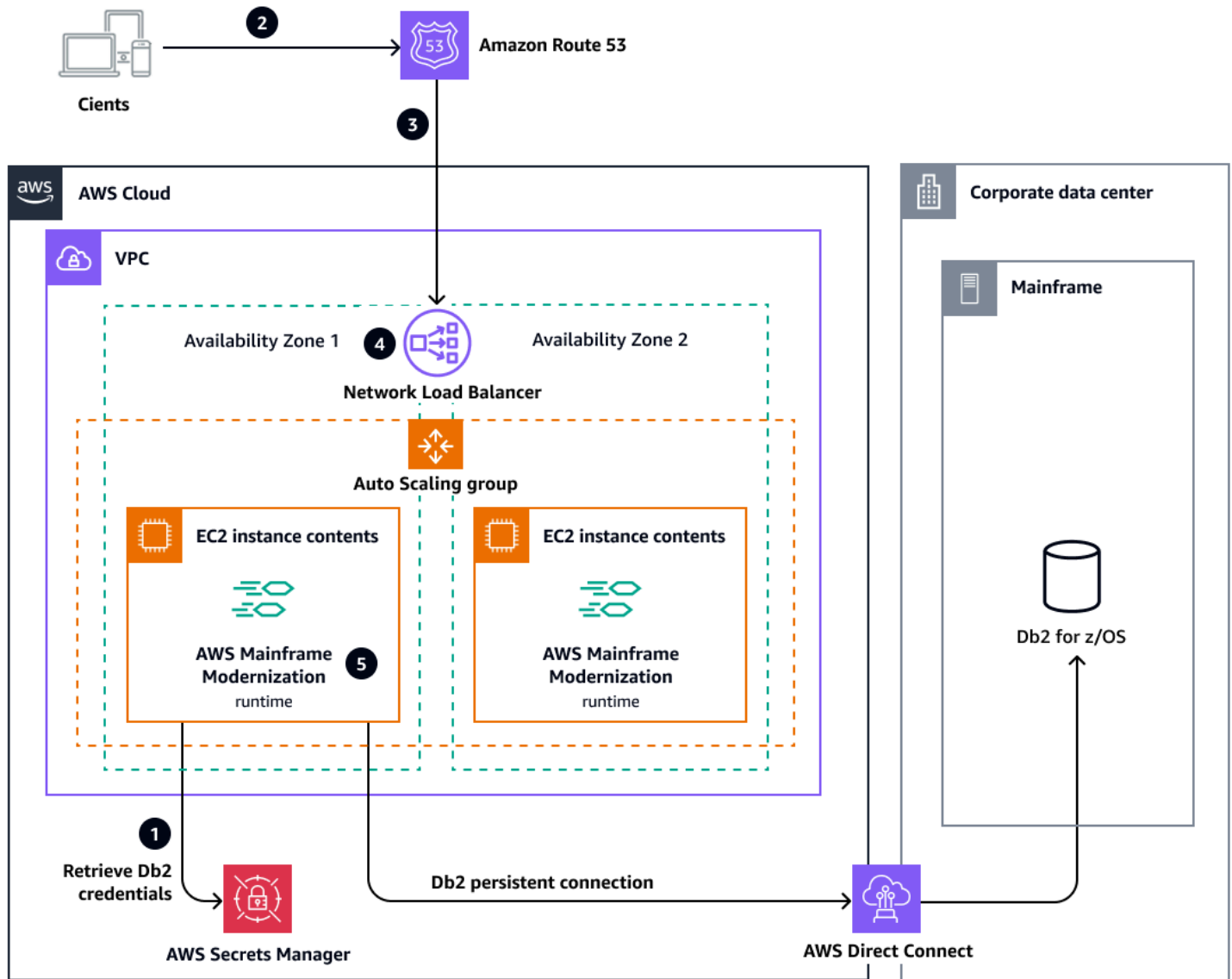
O processo 2PC consiste em duas fases:

- Fase de preparação: o gerente de transações consulta todos os gerentes de recursos envolvidos na transação para garantir que eles estejam prontos para se comprometer.
- Fase de confirmação: se todos os gerentes de recursos responderem positivamente, o gerente de transações os instruirá a confirmar as alterações. Se algum dos gerentes de recursos não puder se comprometer, todos os gerentes serão instruídos a reverter as alterações.

Ao usar o XA, o AWS Replatform with Rocket Software fornece uma solução confiável e escalável para gerenciar transações complexas e distribuídas em ambientes de mainframe modernizados. Esse recurso é essencial para organizações que desejam migrar seus aplicativos de mainframe para a nuvem sem comprometer a integridade ou o desempenho transacional.

infraestrutura de tempo de execução

O diagrama a seguir mostra um ambiente altamente disponível e elástico no Nuvem AWS que inclui duas zonas de disponibilidade, instâncias EC2 em um grupo de Auto Scaling, um Network Load Balancer e uma conexão dedicada AWS entre os ambientes e de mainframe por meio de. AWS Direct Connect



Nesta arquitetura:

1. Quando o AWS Mainframe Modernization tempo de execução é iniciado, ele recupera as credenciais do Db2 [AWS Secrets Manager](#) e abre uma conexão persistente com o Db2 for z/OS.

Note

AWS Mainframe Modernization O serviço (experiência do Managed Runtime Environment) não está mais aberto a novos clientes. Para recursos semelhantes ao AWS Mainframe Modernization Serviço (experiência do Managed Runtime Environment), explore o AWS Mainframe Modernization Service (Experiência autogerenciada). Os clientes atuais podem continuar usando o serviço normalmente. Para obter mais informações, consulte [Mudança de disponibilidade do AWS Mainframe Modernization](#).

2. Os clientes vinculam o endereço do Network Load Balancer no [Amazon Route 53](#).
3. O Route 53 redireciona as transações para o Network Load Balancer.
4. O Network Load Balancer distribui transações em várias instâncias do EC2.
5. A carga de trabalho em execução AWS Mainframe Modernization interage com o Db2 usando uma conexão persistente z/OS por meio de. AWS Direct Connect

Teste

Quando você replataforma um aplicativo COBOL enquanto mantém o Db2 z/OS como banco de dados compartilhado, é crucial garantir que o novo sistema funcione de forma equivalente ao original. Esse ambiente híbrido apresenta desafios e oportunidades únicas para testes. A estratégia a seguir descreve uma abordagem abrangente para testes de equivalência funcional e foi projetada para validar o desempenho, a integridade dos dados e a integração perfeita do aplicativo reformulado com o banco de dados Db2 for z/OS existente.

Comece identificando os processos e transações comerciais essenciais que precisam ser comparados entre os sistemas. Em seguida, crie um plano de teste detalhado com cenários específicos que avaliarão efetivamente a equivalência funcional dessas transações. Por fim, desenvolva conjuntos de dados de teste abrangentes que cubram todos os cenários identificados e certifique-se de que sejam idênticos nos dois sistemas para permitir uma comparação precisa.

Ambiente de origem

- Instantâneo inicial (primeiro instantâneo):
 - Certifique-se de que a tabela de dados não esteja sendo usada por outros aplicativos durante o teste, pois isso pode afetar o teste de equivalência.

- Faça um snapshot do Db2 para z/OS tabelas que são usadas pela transação antes de executar qualquer teste.
- Teste do sistema de origem:
 - Execute o conjunto completo de testes no aplicativo COBOL original.
 - Registre todas as transações, entradas e saídas.
 - Monitore o desempenho do sistema e a utilização de recursos.
- Instantâneo do teste pós-fonte (segundo instantâneo):
 - Faça outro snapshot do z/OS banco de dados Db2 após concluir os testes do sistema de origem.

Ambiente de destino

- Redefinição do banco de dados:
 - Restaure o banco de dados ao estado inicial usando o primeiro instantâneo.
- Teste do sistema alvo (ambiente reformulado):
 - Execute o mesmo conjunto de testes no aplicativo reformulado.
 - Certifique-se de que todos os testes do sistema de destino usem as mesmas entradas dos testes do sistema de origem.
 - Monitore o desempenho do sistema e a utilização de recursos.
- Instantâneo do teste pós-destino (terceiro instantâneo):
 - Faça um instantâneo final do z/OS banco de dados Db2 após concluir os testes do sistema de destino.

Análise

- Comparação e análise:
 - Compare o segundo e o terceiro instantâneos para identificar quaisquer discrepâncias nos dados.
 - Analise os resultados dos testes e compare as saídas dos sistemas de origem e de destino.
 - Avalie as métricas de desempenho entre os dois ambientes.
- Teste de integração:

- Execute testes que envolvam o aplicativo reformulado e quaisquer componentes COBOL restantes.
- Verifique a interação perfeita entre os dois ambientes.
- Teste de failover e recuperação:
 - Cenários de teste em que um ambiente falha e o outro assume o controle.
 - Garanta a consistência e a integridade dos dados durante situações de failover.
- Teste de carga e estresse:
 - Realize testes com cargas variadas para avaliar o desempenho do sistema híbrido sob estresse.
 - Identifique quaisquer gargalos ou problemas de desempenho em qualquer um dos ambientes.
- Documentação e relatórios:
 - Documente todos os resultados, discrepâncias e métricas de desempenho dos testes.
 - Prepare um relatório abrangente que compare os sistemas de origem e de destino.

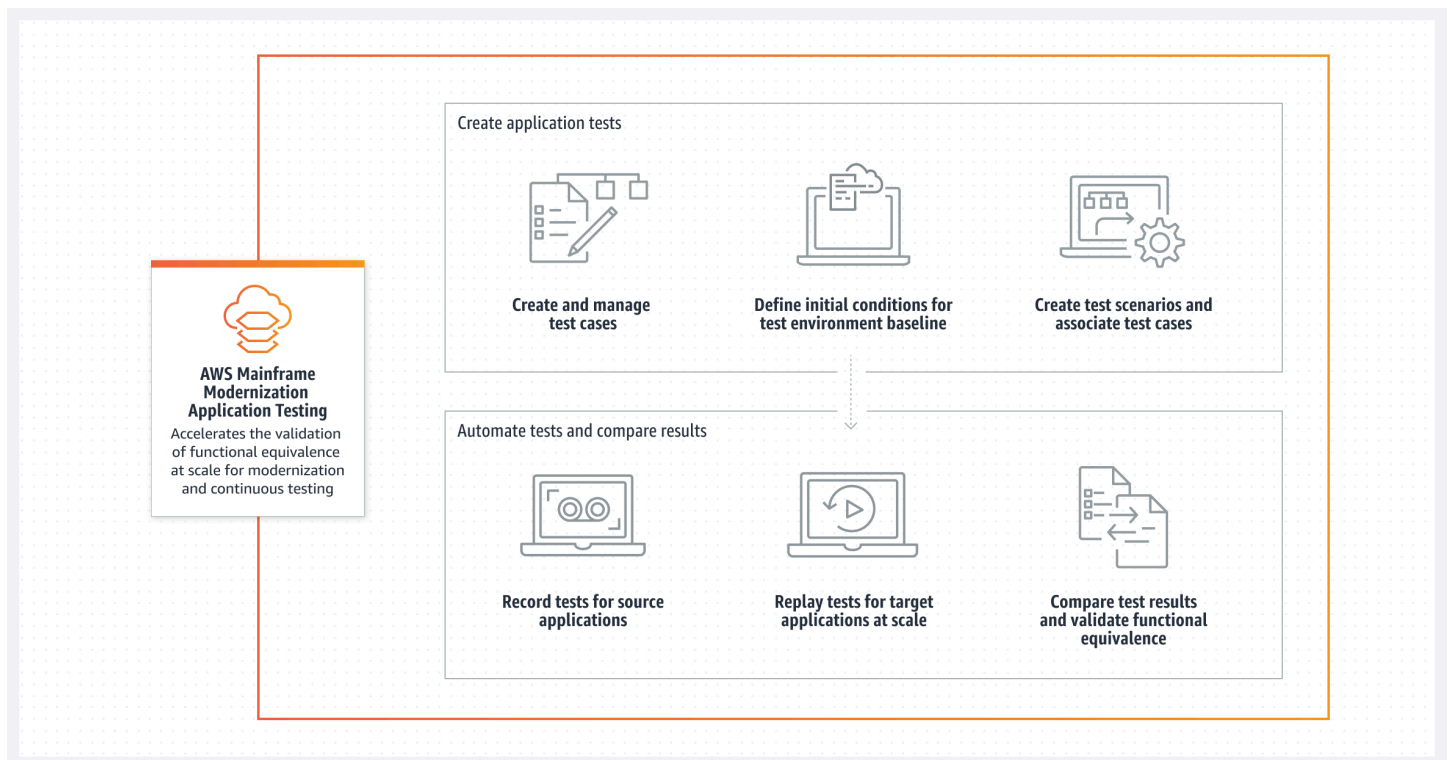
Testando seu aplicativo em AWS Mainframe Modernization

O [Testes de aplicação do AWS Mainframe Modernization](#) serviço automatiza a execução de testes de aplicativos em grande escala. Testes de aplicação da AWS ajuda a otimizar e reduzir os custos do projeto de modernização e teste de aplicativos de mainframe.

Note

AWS Mainframe Modernization O serviço (experiência do Managed Runtime Environment) não está mais aberto a novos clientes. Para recursos semelhantes ao AWS Mainframe Modernization Serviço (experiência do Managed Runtime Environment), explore o AWS Mainframe Modernization Service (Experiência autogerenciada). Os clientes atuais podem continuar usando o serviço normalmente. Para obter mais informações, consulte [Mudança de disponibilidade do AWS Mainframe Modernization](#).

O diagrama a seguir mostra como Testes de aplicação da AWS funciona em alto nível.



O processo consiste nestas etapas:

1. Crie e gerencie casos de teste, que são a menor unidade de ações de teste. Identifique os tipos de dados que melhor representam a equivalência funcional entre os sistemas de origem e de destino.
2. Defina a configuração do ambiente de teste especificando CloudFormation modelos e atributos adicionais.
3. Crie suítes de teste, que são coleções de casos de teste.
4. Carregar e reproduzir conjuntos de dados: capture os conjuntos de dados de entrada e saída no mainframe, carregue-os e AWS, em seguida, reproduza o cenário de teste no sistema de destino.
5. Compare conjuntos de dados de origem e destino. Testes de aplicação da AWS compara automaticamente os conjuntos de dados de saída dos sistemas de origem e de destino. Revise-os e avalie-os para identificar discrepâncias.

Para obter mais informações, consulte a documentação do [AWS Mainframe Modernization](#).

substituição

Na modernização do mainframe, um dos desafios mais importantes é minimizar o tempo de inatividade e o risco durante a transição para uma nova plataforma. A estratégia blue/green de implantação oferece uma abordagem poderosa e flexível para a migração do sistema.

A implantação azul/verde é uma técnica que reduz o tempo de inatividade e o risco ao executar dois ambientes de produção idênticos, chamados de azul e verde. Veja como isso funciona no contexto da modernização do mainframe:

- Ambiente azul: este é o seu sistema de mainframe atual que está lidando com todo o tráfego de produção.
- Ambiente verde: esta é sua plataforma nova e modernizada AWS que está pronta para assumir o controle.

A estratégia de blue/green transição inclui as seguintes etapas: provisionar, entrar em operação, reverter se surgirem problemas e concluir.

Provisão

Nesse estágio, você provisiona o novo ambiente (verde) AWS seguindo estas etapas:

1. Reformule a plataforma do ambiente: a zona hospedada do [Route 53](#) deve conter um [registro DNS que aponte](#) para o ambiente do mainframe (azul).
2. Verifique a conectividade: garanta a conexão adequada entre seus gerenciadores de transações locais Conta da AWS e o Db2 for z/OS Database.
3. Execute testes de fumaça: use o endereço do balanceador de AWS carga para acessar o ambiente reformado e realizar testes de fumaça abrangentes para verificar o seguinte:
 - Todas as cargas de trabalho esperadas estão disponíveis.
 - 3270 transações estão sendo processadas corretamente.
 - As interações de dados com o Db2 for z/OS estão funcionando conforme o esperado.

Transmita ao vivo

Nesse estágio, você transfere o tráfego para o ambiente verde e monitora as mudanças.

1. Use as políticas de roteamento de tráfego no Route 53 para mudar o tráfego:
 - Opção A: Você pode mudar o tráfego de uma só vez.
 - Opção B: Como alternativa, você pode usar uma distribuição ponderada gradual.
2. Monitore e valide:
 - Observe atentamente o AWS ambiente à medida que o tráfego muda.
 - Verifique o processamento da transação 3270.
 - Verifique a z/OS comunicação do Db2.
 - Monitore problemas de desempenho.
 - Faça com que os usuários validem os resultados da transação.

Reversão

Se surgirem problemas, você pode atualizar rapidamente o Route 53 para redirecionar o tráfego de volta para o ambiente de mainframe local (azul).

Você deve investigar e resolver problemas antes de tentar outra transição.

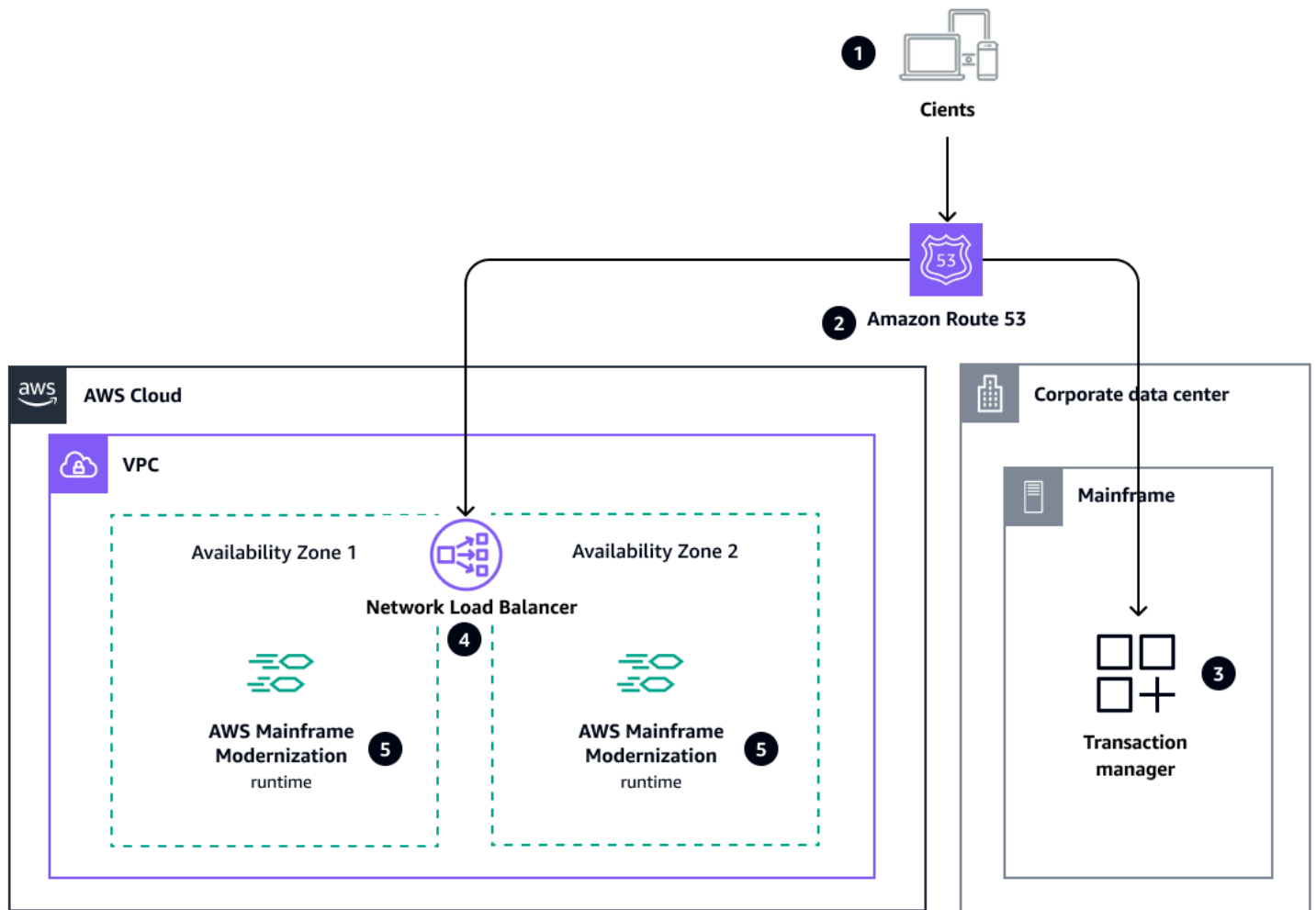
Concluir

Depois de monitorar o tráfego e validar que seu ambiente verde está funcionando corretamente, você pode aumentar gradualmente o tráfego do aplicativo para AWS.

Após um período estável, você pode desativar o ambiente de transação do mainframe (azul) e manter o z/OS banco de dados Db2 for no local.

Arquitetura

O diagrama a seguir ilustra o fluxo de transição.



O processo de transição consiste no seguinte:

1. Os aplicativos cliente, front-ends e back-ends para frontends (BFFs) enviam transações para o nome de domínio do Route 53.
2. O Route 53 roteia a conexão para o gerenciador de transações do mainframe ou para o Network Load Balancer, dependendo da política de roteamento definida.
3. O gerenciador de transações processa as transações que são enviadas para o mainframe.
4. O Network Load Balancer distribui transações para os ambientes de replataforma disponíveis para processamento.
5. Os ambientes AWS Mainframe Modernization de replataforma processam as solicitações.

Práticas recomendadas

Esta seção descreve um conjunto de melhores práticas para enfrentar os principais desafios na replataforma de cargas de trabalho de mainframe em ambientes de nuvem e, ao mesmo tempo, manter o banco de dados no Db2 for z/OS.

Latência de rede

Para prever com precisão o impacto da latência da separação do aplicativo do banco de dados Db2 durante um esforço de reformulação de plataforma, recomendamos que você realize uma avaliação completa do número de chamadas do Db2 para transações e processos em lote. Essa avaliação deve ser feita usando dados de rastreamento e deve incluir as seguintes etapas:

- **Colete dados de rastreamento:** reúna rastreamentos detalhados de transações representativas e trabalhos em lote e certifique-se de que os rastreamentos capturem todas as interações do Db2, incluindo entradas e saídas.
- **Analise os dados de rastreamento:** conte o número de entradas e saídas do Db2 para cada transação e trabalho em lote e calcule o número médio de interações do Db2 por transação e processo em lote.
- **Meça os tempos de resposta atuais:** verifique se o acesso ao Db2 está alinhado com o contrato de nível de serviço (SLA) do seu aplicativo.
- **Estime a latência da rede:** determine a latência de rede esperada entre o aplicativo reformulado e o banco de dados Db2. Considere fatores como distância física, infraestrutura de rede e possíveis gargalos.
- **Calcule o impacto potencial:** para cada transação e processo em lote, multiplique o número de entradas e saídas do Db2 pela latência estimada da rede. Adicione esse tempo calculado aos tempos de resposta atuais para prever o novo tempo total de processamento.
- **Avalie os resultados:** avalie se o aumento previsto da latência é aceitável para seus requisitos de negócios e identifique quaisquer transações ou processos que possam exigir otimização ou redesenho para mitigar os problemas de latência.
- **Considere estratégias de mitigação:** explore opções como pool de conexões, armazenamento em cache ou recuperação de dados em lote para reduzir o número de interações individuais do Db2. Avalie a possibilidade de mover dados acessados com frequência para mais perto do nível do aplicativo.

Seguindo essas etapas, você poderá tomar decisões baseadas em dados sobre a viabilidade de sua estratégia de replataforma e identificar possíveis problemas de desempenho antes que eles afetem seu ambiente de produção. Essa abordagem ajudará a garantir uma transição suave, mantendo níveis de desempenho aceitáveis para seus aplicativos dependentes de banco de dados.

Segurança

- Proteja a criação do seu aplicativo: use uma sub-rede privada na nuvem privada virtual (VPC) para AWS CodeBuild executar e ajudar a garantir o isolamento e a segurança aprimorada. Implemente um contexto confiável do Db2 a partir do CIDR da CodeBuild sub-rede para acesso seguro ao banco de dados durante o processo de criação.
- Proteja seu ambiente de tempo de execução: use um contexto confiável do Db2 da sub-rede CIDR de tempo de execução para conexões seguras de banco de dados.
- Gerencie as credenciais do banco de dados com segurança: implemente um cronograma regular de rotação de credenciais para minimizar o risco de acesso não autorizado. Armazene as credenciais do Db2 com segurança no AWS Secrets Manager
- Estabeleça a segurança da rede: implemente regras sólidas de segmentação de rede e firewall para proteger os ambientes de construção e de tempo de execução. Use a combinação correta de AWS Direct Connect e AWS Site-to-Site VPN para alcançar o nível de segurança necessário.
- Aplique criptografia: aplique criptografia para dados em trânsito entre seu aplicativo e o Db2 for z/OS.

Governança de aplicativos

- Estabeleça uma fonte confiável: estabeleça o novo gerenciamento de configuração de software (SCM) — por exemplo, GitHub — como a única fonte confiável para o código do aplicativo migrado. Isso garante a consistência e elimina as discrepâncias de versão entre os ambientes de nuvem e mainframe durante o período de transição.
- Atualize o processo de gerenciamento de mudanças: atualize o processo de gerenciamento de mudanças para considerar as modificações no código nesse novo paradigma de ambiente duplo. Esse processo deve incluir:
 - Fluxos de trabalho de aprovação claros para alterações de código.
 - Revisões obrigatórias do código antes de mesclar o código na ramificação principal.

- Procedimentos de implantação sincronizados para garantir que os dois ambientes recebam atualizações simultaneamente.
- Mecanismos de reversão em caso de problemas em qualquer ambiente.

Elasticidade

A elasticidade da computação em nuvem introduz uma mudança de paradigma que altera significativamente a estrutura de custos e o gerenciamento de recursos do mainframe.

Diferentemente do ambiente tradicional de mainframe, que tem capacidade fixa e modelos de preços baseados em picos, as plataformas de nuvem oferecem escalabilidade dinâmica e uma pay-as-you-go abordagem que pode potencialmente levar a economias substanciais de custos e maior eficiência operacional.

Em um ambiente de nuvem, as organizações podem aumentar ou diminuir seus recursos de computação em tempo real com base na demanda real, o que elimina a necessidade de provisionamento excessivo para acomodar cargas de pico. Essa elasticidade permite que as empresas paguem apenas pelos recursos que consomem, em vez de investir em licenças caras de hardware e software para lidar com picos ocasionais no uso.

Para obter detalhes sobre como os preços funcionam AWS, consulte [AWS Preços](#).

Próximas etapas

A modernização do mainframe é uma iniciativa complexa e crítica que exige conhecimento especializado e soluções avançadas. Você pode acelerar seu processo de modernização e alcançar resultados comerciais mais rápidos por meio de [parcerias estratégicas](#) que ajudarão você nas seguintes tarefas:

- **Avalie e priorize:** analise seus aplicativos de mainframe e identifique quais são adequados para reformulação de plataforma, mantendo o banco de dados no Db2 for z/OS. Considere fatores como complexidade, importância comercial e potencial retorno sobre o investimento (ROI).
- **Desenvolva uma estratégia de migração:** crie um plano detalhado para reformular a plataforma de seus aplicativos selecionados, incluindo cronogramas, alocação de recursos e estratégias de mitigação de riscos.
- **Avalie ferramentas e tecnologias:** pesquise e selecione ferramentas e tecnologias apropriadas para facilitar o processo de replataforma, como plataformas de modernização de aplicativos ou ferramentas de conversão de código.
- **Interaja com especialistas:** considere fazer parcerias com especialistas em modernização de mainframe ou empresas de consultoria que tenham experiência em projetos de replataforma.
- **Prova de conceito:** comece com uma prova de conceito em pequena escala para validar sua abordagem e identificar possíveis desafios antes de expandir para aplicativos maiores.
- **Teste e validação:** desenvolva uma estratégia de teste abrangente para garantir que seus aplicativos reformulados funcionem corretamente e mantenham a integridade dos dados com seu banco de dados Db2 for existente. z/OS
- **Treinamento e transferência de conhecimento:** prepare sua equipe para o novo ambiente fornecendo treinamento sobre os aplicativos reformulados e quaisquer novas ferramentas ou tecnologias introduzidas.
- **Implementação em fases:** considere uma abordagem em fases para a replataforma, na qual você migra gradualmente os aplicativos enquanto monitora o desempenho e resolve quaisquer problemas que surjam.
- **Otimização contínua:** após a reformulação da plataforma, monitore e otimize continuamente o desempenho de seus aplicativos e suas interações com o z/OS banco de dados Db2 for para garantir o sucesso a longo prazo.
- **Modernize no seu ritmo:** agora que a carga de trabalho está sendo executada AWS e já está aproveitando a nuvem, comece a planejar a fase de reimaginação de sua modernização.

Recursos

Para obter mais informações sobre migração e modernização do mainframe, consulte os recursos a seguir.

AWS documentação

- [Configurar o Amazon Route 53 como serviço DNS](#)
- [Roteamento de tráfego para um balanceador de carga ELB](#)
- [Roteamento ponderado](#)
- [Reestruturando aplicativos com o Rocket Software](#)

Referências da Rocket Software

- [Interface de chamada externa \(ECI\) da Micro Focus](#)
- [Serviços Web do CICS](#)

Referências da IBM

- [Contextos confiáveis](#) (IBM Db2 para z/OS documentação)

Ferramentas

- [Servidor Rocket Enterprise](#)

AWS Padrões e guias de orientação prescritiva

- [Crie programas COBOL Db2 usando e AWS Mainframe ModernizationAWS CodeBuild](#)
- [DevOps para AWS Mainframe Modernization](#)
- [Modernização do mainframe: padrões de desacoplamento para migrar o código do aplicativo](#)
- [Proteja e simplifique o acesso do usuário em um banco de dados de federação Db2 AWS usando contextos confiáveis](#)

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
Publicação inicial	—	7 de maio de 2025

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 disrupção 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

Uma coleção 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 [disastre](#). 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](#).

Detecção 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 (OU)s. 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 previsões 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, vaziar 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 microsserviço 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 microsserviços 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 no 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. Para obter mais informações, consulte o guia [Como quantificar a incerteza em sistemas de aprendizado profundo](#).

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.