



Memobilisasi organisasi Anda untuk mempercepat migrasi skala besar

# AWS Panduan Preskriptif



# AWS Panduan Preskriptif: Memobilisasi organisasi Anda untuk mempercepat migrasi skala besar

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

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau mungkin tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

---

# Table of Contents

Pengantar .....	1
Gambaran umum .....	3
Menilai .....	3
Memobilisasi .....	4
Migrasi .....	5
Tahap Menilai .....	6
Penilaian Kesiapan Migrasi (MRA) .....	6
Tahap Memobilisasi .....	7
Kasus bisnis terperinci .....	7
Penemuan portofolio terperinci .....	8
Migrasi aplikasi .....	9
Tata kelola migrasi .....	11
Zona pendaratan .....	12
Keamanan, risiko, dan kepatuhan .....	13
Operasi .....	14
Orang: keterampilan, budaya, perubahan, dan kepemimpinan .....	15
Fase migrasi .....	18
Pabrik migrasi .....	18
Langkah selanjutnya .....	20
Sumber Daya .....	21
Panduan .....	21
Pola .....	21
Sumber daya tambahan .....	21
Riwayat dokumen .....	23
Glosarium .....	24
# .....	24
A .....	25
B .....	28
C .....	30
D .....	33
E .....	38
F .....	40
G .....	42
H .....	43

---

I .....	44
L .....	47
M .....	48
O .....	53
P .....	55
Q .....	58
R .....	59
D .....	62
T .....	66
U .....	67
V .....	68
W .....	68
Z .....	69
.....	lxxi

# Memobilisasi organisasi Anda untuk mempercepat migrasi skala besar

Addie Cox, Adam Spicer, Christine Screnci, dan Sukhvinder Gulati, Amazon Web Services

Oktober 2025 ([sejarah dokumen](#))

Migrasi ratusan atau ribuan beban kerja membutuhkan koordinasi dan implementasi di berbagai disiplin ilmu dan tim. AWS mendekati migrasi skala besar dalam tiga fase: menilai, memobilisasi, dan bermigrasi. Setiap fase dibangun di atas yang sebelumnya. Strategi Bimbingan AWS Preskriptif ini mencakup fase penilaian dan fase mobilisasi. Fase-fase ini menetapkan dasar untuk percepatan migrasi dalam skala besar selama fase migrasi.

Strategi migrasi ini mencakup metode dan praktik terbaik untuk hal-hal berikut:

- Memobilisasi tenaga kerja dan sumber daya Anda
- Membangun otomatisasi untuk memigrasikan aplikasi dengan pendekatan tangkas
- Migrasi aset ini ke AWS Cloud

Pendekatan ini didasarkan pada pengalaman bertahun-tahun tim Layanan AWS Profesional membantu AWS pelanggan perusahaan memobilisasi untuk adopsi dan migrasi cloud.

Strategi ini untuk siapa saja yang berencana untuk memigrasikan beban kerja lokal mereka ke AWS Cloud: eksekutif TI dan bisnis, manajer program dan proyek, pemilik produk, dan manajer operasi dan infrastruktur. Ini membahas bagaimana Anda dapat mempercepat adopsi cloud Anda, membuat rencana migrasi, menyiapkan fondasi melalui pengalaman langsung, dan memigrasikan set aplikasi pertama Anda ke minggu berikutnya. AWS Cloud

Dalam strategi ini:

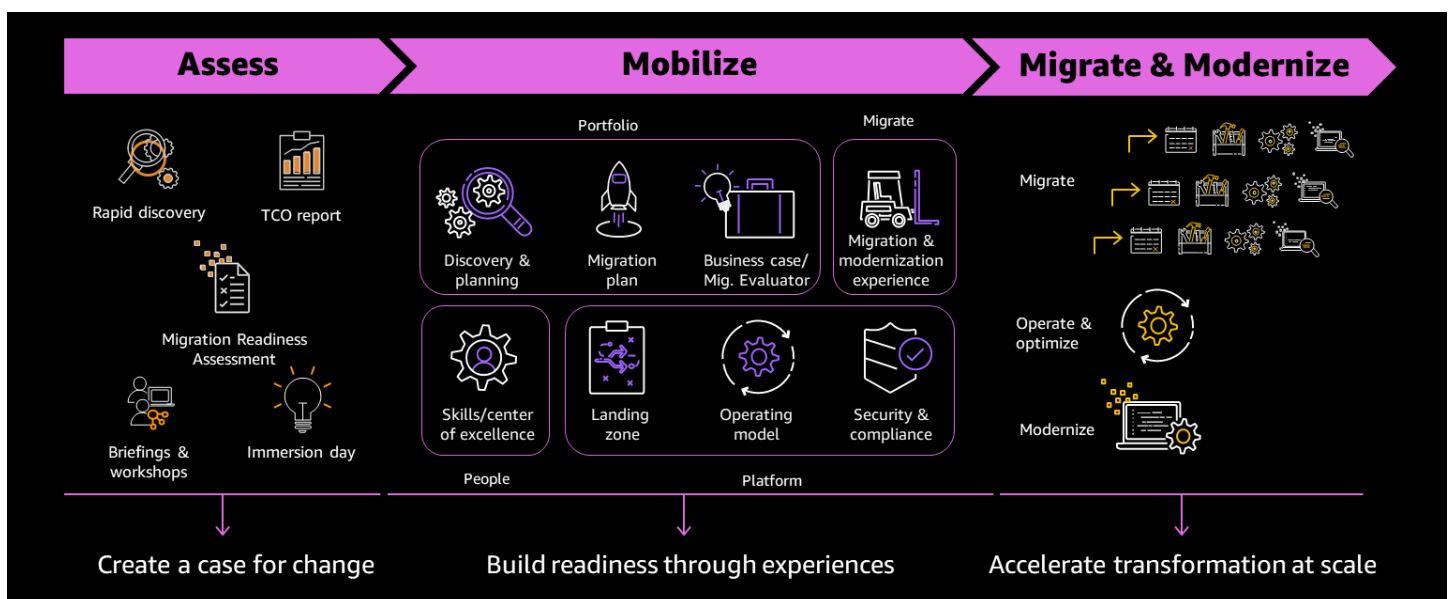
- [Ikhtisar](#)
- [Menilai fase](#)
- [Memobilisasi fase](#)
- [Fase migrasi](#)
- [Langkah selanjutnya](#)
- [Sumber Daya](#)

- [Riwayat dokumen](#)

## Gambaran umum

Untuk mengoptimalkan adopsi dan migrasi cloud Anda, Anda harus terlebih dahulu menilai dan mengevaluasi kesiapan perusahaan Anda. Setelah Anda menilai kesiapan organisasi Anda, Anda dapat mempercepat adopsi cloud Anda dan membangun fondasi untuk migrasi dalam skala besar melalui pengalaman langsung migrasi gelombang pertama aplikasi hanya dalam beberapa minggu. Kami membahas fase-fase ini secara lebih rinci dalam topik-topik berikut.

Pendekatan berulang untuk adopsi cloud yang dibahas dalam panduan ini dapat dipecah menjadi tiga fase tingkat tinggi yaitu menilai, memobilisasi, dan bermigrasi. Fase-fase ini ditunjukkan pada diagram berikut.



## Menilai

Tahap pertama migrasi cloud dimulai dengan penilaian kesiapan cloud terhadap keadaan perusahaan Anda saat ini. Tugas penilaian ini memberi Anda gambaran yang jelas tentang apa yang akan Anda migrasi, serta kasus bisnis untuk analisis migrasi dan biaya kepemilikan total (TCO) untuk migrasi.

Penilaian Kesiapan Migrasi (MRA) adalah proses yang memberi Anda hal-hal berikut:

- Pemahaman tentang seberapa jauh bisnis Anda dalam perjalanan cloud
- Pemahaman tentang kekuatan dan kelemahan Anda saat ini dalam kaitannya dengan perusahaan yang siap cloud

- Rencana aksi untuk menutup kesenjangan yang teridentifikasi

Penilaian ini didasarkan pada [AWS Cloud Adoption Framework \(AWS CAF\)](#) dan enam perspektifnya: bisnis, orang, tata kelola, platform, keamanan, dan operasi. Menggunakan kerangka kerja ini selama fase penilaian membantu memastikan bahwa Anda memiliki pandangan holistik tentang inisiatif transformasi yang akan mendukung langkah efektif ke cloud.

Fase penilaian mencakup proses penemuan cepat dengan penilaian TCO tingkat tinggi selain MRA. Langkah-langkah ini akan berguna saat Anda maju untuk mendapatkan komitmen pemangku kepentingan dan pendanaan untuk inisiatif yang lebih besar berikutnya.

Untuk informasi lebih lanjut, lihat [Fase menilai](#).

## Memobilisasi

Dengan pemahaman yang kuat tentang keadaan Anda saat ini dan portofolio Anda bersama dengan kasus bisnis yang solid, Anda berkembang dalam fase penilaian, Anda dapat menunjukkan nilai cloud ke organisasi Anda.

Tujuan dari fase mobilisasi adalah untuk membangun kemampuan dasar baik dalam organisasi maupun dengan AWS lingkungan dengan pengalaman migrasi langsung yang berfokus pada otomatisasi keamanan dan operasi. Proses ini menyatukan portofolio alat dan praktik Anda dalam landing zone yang terukur dan aman, yang merupakan kerangka orkestrasi untuk lingkungan dasar Anda. AWS Pada fase ini, Anda memigrasikan sekumpulan kecil aplikasi bisnis ke cloud, sambil menerapkan budaya pengiriman yang gesit dan terukur, struktur tim, dan proses manajemen perubahan.

Pendekatan AWS mobilisasi mencakup serangkaian aktivitas yang ditentukan di delapan alur kerja yang berbeda:

- Kasus bisnis terperinci
- Penemuan portofolio terperinci
- Migrasi aplikasi
- Tata kelola migrasi
- AWS landing zone
- Keamanan, risiko, dan kepatuhan

- Operasi
- Orang: keterampilan, budaya, perubahan, dan kepemimpinan

Dengan pendekatan ini, kegiatan di bawah alur kerja ini disampaikan melalui delapan sprint dua minggu.

Memobilisasi memberikan hasil sebagai berikut:

- Tingkatkan keterampilan dan kompetensi staf TI Anda untuk migrasi dan transformasi organisasi dengan membangun kepercayaan diri dengan metodologi, proses, dan perkakas.
- Menentukan dan mengotomatiskan kebijakan keamanan, risiko, dan kepatuhan untuk mengakomodasi kontrol operasional.
- Tentukan dan terapkan AWS landing zone yang dapat diskalakan saat Anda memigrasikan aplikasi tambahan.
- Buat model operasi cloud Anda.
- Jalankan aplikasi dalam kapasitas produksi.

Migrasi selama fase mobilisasi adalah proses berulang yang berkembang saat organisasi Anda mengembangkan keterampilan, alat, dan kemampuan baru. Keterampilan dan kemampuan ini membangun momentum dan mempercepat upaya migrasi Anda dari waktu ke waktu. Membangun fondasi yang tepat untuk dibangun adalah kunci keberhasilan migrasi. Kerangka kerja AWS migrasi menyeimbangkan bisnis dan upaya teknis yang diperlukan untuk menyelesaikan migrasi cloud. Model ini membantu Anda mengidentifikasi driver bisnis utama untuk migrasi dan strategi terbaik untuk merencanakan dan menerapkan migrasi cloud yang sukses.

Untuk informasi lebih lanjut, lihat [fase Mobilisasi](#).

## Migrasi

Fase migrasi menggunakan pola, proses, alat, sumber daya, dan metodologi yang ditentukan dan diuji selama fase mobilisasi untuk memigrasikan aplikasi dalam skala besar. Setelah menggunakan praktik dan pelajaran terbaik dari fase sebelumnya, Anda dapat menerapkan pabrik migrasi, yang merupakan cetak biru implementasi dan operasi penskalaan, melalui otomatisasi dan pengiriman tangkas.

Untuk informasi selengkapnya, lihat [Fase Migrasi](#).

# Tahap Menilai

Untuk memobilisasi tenaga kerja dan sumber daya Anda untuk memigrasikan perusahaan Anda dalam skala besar, Anda harus terlebih dahulu menilai kesiapan dan inventaris portofolio Anda. Langkah-langkah ini memungkinkan Anda untuk membangun kasus bisnis dengan indikator kinerja kunci yang tepat (KPIs) dan metrik, dan untuk menyelaraskan pemangku kepentingan utama dan kepemimpinan pada tujuan bisnis. Anda akan menjalankan alur kerja Penilaian Kesiapan Migrasi (MRA) selama fase ini.

## Penilaian Kesiapan Migrasi (MRA)

MRA adalah AWS proses untuk mendapatkan wawasan tentang kesiapan cloud perusahaan Anda saat ini dan membangun rencana tindakan untuk menutup celah yang teridentifikasi, menggunakan AWS Cloud Adoption Framework (AWS CAF).

### Objektif

- Menilai dan mengevaluasi kesiapan perusahaan Anda dengan menjawab pertanyaan berdasarkan enam perspektif AWS CAF.

### Hasil

- Memahami di mana perusahaan Anda berada dalam perjalanan cloud Anda
- Identifikasi area kekuatan dan kelemahan
- Rencanakan untuk mengisi celah yang diidentifikasi

### Panduan cara

[Mengevaluasi kesiapan migrasi](#)

# Tahap Memobilisasi

Langkah selanjutnya dalam mempersiapkan tenaga kerja dan sumber daya Anda untuk memigrasikan perusahaan Anda dalam skala besar adalah memecah aktivitas mobilisasi ke dalam alur kerja yang berbeda. Meskipun tujuan dari fase mobilisasi adalah migrasi aplikasi bisnis, fase ini juga memberikan kesempatan untuk meletakkan dasar bagi perkakas, proses, dan budaya yang akan mempercepat migrasi Anda dalam skala besar. Sebagian besar workstream ini dapat berjalan secara paralel setelah fase penilaian selesai. Workstream berikut harus dijalankan selama fase ini:

- [Kasus bisnis terperinci](#)
- [Penemuan portofolio terperinci](#)
- [Migrasi aplikasi](#)
- [Tata kelola migrasi](#)
- [Zona pendaratan](#)
- [Keamanan, risiko, dan kepatuhan](#)
- [Operasi](#)
- [Orang: keterampilan, budaya, perubahan, dan kepemimpinan](#)

Ini dibahas secara rinci di bagian berikut.

## Kasus bisnis terperinci

Kasus migrasi bisnis multi-tahun yang terperinci yang mencakup biaya lokal saat ini, biaya baru, dan AWS biaya migrasi membantu menyelaraskan pemangku kepentingan dan eksekutif.

### Tujuan

- Tentukan biaya migrasi.
- Perkirakan berapa banyak yang dapat Anda hemat dengan bermigrasi ke AWS.
- Perkirakan manfaat bisnis lain dari migrasi.
- Tentukan panjang migrasi target.
- Tentukan beban kerja mana yang akan Anda migrasikan dan tahun berapa.
- Masukkan inventaris terperinci untuk setiap beban kerja.

## Hasil

- Kasus bisnis migrasi multi-tahun
- Biaya migrasi

## AWS Mitra dan alat

AWS Mitra seperti RISC Networks, Deloitte, Cloudamize, TSO Logic, dan Apptio memiliki alat dan pengalaman di bidang ini.

## Penemuan portofolio terperinci

Di sinilah Anda mulai menyatukan potongan-potongan itu dan mengembangkan strategi untuk migrasi. Pada tahap ini, Anda ingin mempertimbangkan di mana perjalanan cloud Anda cocok dengan strategi bisnis organisasi Anda yang lebih besar dan menemukan peluang untuk penyelarasan visi. Strategi migrasi yang selaras dengan kasus bisnis pendukung dan rencana migrasi aplikasi menyeluruh menetapkan dasar yang tepat untuk keberhasilan adopsi cloud.

Aspek penting dari strategi migrasi Anda adalah pengumpulan data portofolio aplikasi dan rasionalisasi data ini terhadap [tujuh Rs migrasi](#): rehost, replatform, refactor/re-architect, pembelian kembali, relokasi, pensiun, dan pertahankan. Anda membangun rencana gelombang migrasi dengan menggunakan tujuh Rs untuk mengkategorikan apa yang ada di lingkungan Anda. Selanjutnya, gabungkan kategori-kategori tersebut dengan informasi tentang saling ketergantungan, kompleksitas teknis untuk bermigrasi, dan bagaimana Anda akan memigrasikan setiap aplikasi atau kumpulan aplikasi. Setelah Anda menganalisis aplikasi Anda sesuai dengan tujuh Rs, Anda dapat menguraikan rencana migrasi untuk setiap aplikasi dalam portofolio Anda. Ini adalah rencana berulang yang akan matang saat Anda maju melalui migrasi, membangun kepercayaan diri, mempelajari kemampuan baru, dan lebih memahami perkebunan Anda yang ada.

## Tujuan

- Kembangkan daftar grup migrasi yang diprioritaskan, termasuk aplikasi dan infrastruktur terkait, untuk semua aplikasi dalam lingkup.
- Tentukan elemen data bisnis dan infrastruktur yang diperlukan, dan rekomendasikan alat pengumpulan data.
- Bekerja dengan tim kepemimpinan bisnis dan TI untuk menentukan driver migrasi dan akhirnya rencana portofolio.

- Buat rencana migrasi dengan ketelitian tinggi untuk portofolio aplikasi yang mencakup aktivitas berikut:
  - Temukan lingkungan keadaan saat ini, termasuk semua aplikasi dan infrastruktur pendukung.
  - Tentukan dependensi aplikasi dan infrastruktur.
  - Dokumentasikan kekritisian aplikasi, siklus hidup, dan siklus bisnis.
  - Kelompokkan aplikasi dan infrastruktur ke dalam grup dan pola migrasi.
  - Tentukan kesiapan dan kesesuaian migrasi, desain status target, dan pola migrasi.
  - Kembangkan jadwal migrasi yang diprioritaskan.

## Hasil

- Jadwal migrasi dengan kesetiaan tinggi dan diprioritaskan untuk empat sprint migrasi awal
- Data aplikasi dan infrastruktur yang cukup untuk mengelompokkan dan menjadwalkan seluruh portofolio aplikasi

## AWS Mitra dan alat

Jika Anda memerlukan bantuan untuk memahami portofolio TI Anda, Anda dapat bekerja dengan AWS Mitra seperti RISC Networks, Cloudamize, dan Deloitte.

## Panduan cara

- [Penemuan dan analisis portofolio](#)

# Migrasi aplikasi

Aliran kerja migrasi aplikasi mengintegrasikan output dari alur kerja lain dengan migrasi aplikasi produksi ke file. AWS Cloud Alur kerja ini memandu sumber daya Anda dan mengarahkan Anda melalui tantangan migrasi aplikasi, praktik terbaik, kerangka kerja tangkas, alat, dan proses yang dapat berhasil diterapkan pada upaya migrasi skala besar apa pun.

## Tujuan

Migrasikan aplikasi bisnis dari lokasi ke AWS Cloud:

- Tentukan cara memilih dan memprioritaskan aplikasi untuk migrasi.

- Pahami praktik terbaik yang telah terbukti untuk memigrasikan aplikasi ke AWS.
- Validasi AWS landing zone, runbook operasi, dan buku pedoman keamanan Anda dengan menguji aplikasi. AWS
- Latih staf internal Layanan AWS dan AWS Partner alat-alat melalui pengalaman langsung.
- Pelajari alat dan teknik migrasi yang diterima industri untuk berbagai jenis aplikasi.
- Mulailah dengan menggunakan epos (backlog) yang ada dari pola [migrasi](#) yang berbeda.

## Hasil

- Validasi berbagai komponen AWS landing zone melalui penerapan dan pengujian aplikasi
- Validasi model operasi yang diuraikan (output dari aliran kerja operasi) dengan menerapkan, memantau, dan melaporkan aplikasi yang berjalan di AWS Cloud
- Memahami proses tangkas yang dapat diskalakan dan pola migrasi untuk memigrasi aplikasi yang berbeda
- Mempelajari cara merancang arsitektur target dan cara menggunakan migrasi data, migrasi server, dan alat vendor untuk migrasi otomatis
- Sumber daya terlatih Layanan AWS, dan berikan pengalaman migrasi langsung
- Validasi buku pedoman keamanan Anda melalui audit real-time dari aplikasi yang berjalan AWS
- Pembelajaran, implementasi, dan validasi proses pengiriman migrasi, termasuk jumlah sumber daya, kecepatan (kecepatan migrasi), jaminan kualitas, manajemen rilis, dan integrasi dengan penyedia layanan terkelola () MSPs

## AWS Mitra dan alat

Bila Anda memiliki pengalaman mendasar dalam memigrasikan beberapa aplikasi dan rencana yang didukung organisasi, inilah saatnya untuk mempercepat migrasi dan mencapai skala. Mitra Pengiriman Migrasi, seperti 2nd Watch dan Accenture, dapat membantu Anda melalui setiap tahap migrasi. Mitra Marketplace Migration seperti RiverMeadow Software, dan Attunity juga dapat membantu, dan Anda dapat menggunakan alat dan layanan seperti [AWS Transform MGN](#) dan [AWS Database Migration Service \(AWS DMS\)](#). Untuk daftar lengkap semua Mitra AWS Migrasi dan solusi, lihat situs web Mitra [Kompetensi AWS Migrasi dan Modernisasi](#).

# Tata kelola migrasi

Alur kerja tata kelola migrasi mencakup pengelolaan ruang lingkup migrasi, jadwal, rencana sumber daya, masalah dan risiko, dan komunikasi kepada semua pemangku kepentingan. Beberapa aplikasi dimigrasikan dalam beberapa aliran, berdampak pada beberapa tim, sehingga fokus awal pada perencanaan dapat membantu mengatur proyek. Rencana migrasi mempertimbangkan faktor-faktor penting seperti urutan di mana beban kerja akan bermigrasi, kapan sumber daya diperlukan, dan bagaimana kemajuan migrasi akan dilacak. Kami merekomendasikan metodologi pengiriman tangkas, praktik terbaik pengendalian proyek, rencana komunikasi bisnis yang kuat, dan pendekatan pengiriman yang terdefinisi dengan baik.

## Tujuan

Mengelola ruang lingkup, jadwal, rencana sumber daya, masalah dan risiko, koordinasi, dan komunikasi kepada semua pemangku kepentingan:

- Siapkan tim scrum yang terdiri dari sumber daya internal dari alur kerja yang ditentukan dalam fase ini.
- Identifikasi 10 hingga 30 aplikasi untuk bermigrasi dari tempat ke AWS lokasi.
- Tinjau metode dan kemampuan manajemen proyek keadaan saat ini.
- Mendefinisikan manajemen proyek metode tangkas dan alat-alat yang akan digunakan selama proyek.
- Identifikasi tim tingkat tinggi untuk setiap alur kerja.
- Tentukan piagam proyek, pelaporan, dan prosedur eskalasi.
- Memfasilitasi koordinasi dan kegiatan kelompok di seluruh proyek.

## Hasil

- Siapkan program tangkas dengan epos untuk semua alur kerja dalam tahap kesiapan dan perencanaan.
- Memberikan pertimbangan dan rencana implementasi migrasi untuk penemuan, transformasi, dan penyebaran set komponen beban kerja yang kompleks.

## AWS Mitra dan alat

AWS Mitra Kompetensi Migrasi dapat membantu Anda melalui setiap tahap migrasi, mempercepat hasil dengan menyediakan personel, alat, dan pendidikan dalam bentuk layanan profesional. Mitra ini adalah [penyedia layanan terkelola \(MSPs\)](#) atau memiliki hubungan dengan MSP AWS yang diaudit untuk membantu pelanggan dengan dukungan beban kerja yang berkelanjutan. AWS Untuk mempelajari lebih lanjut, lihat Mitra [Kompetensi AWS Migrasi dan Modernisasi](#).

Panduan cara

- [Menyiapkan program tangkas untuk mempercepat migrasi cloud](#)

## Zona pendaratan

Landing zone adalah kerangka orkestrasi untuk lingkungan dasar Anda. AWS Ini memberikan dasar untuk memulai dengan arsitektur multi-akun, identitas dan manajemen akses, tata kelola, keamanan data, desain jaringan, dan logging.

AWS memiliki dua opsi untuk membuat landing zone Anda: menggunakan landing zone berbasis layanan [AWS Control Tower](#) dan landing zone khusus yang Anda bangun. Setiap opsi membutuhkan tingkat AWS pengetahuan yang berbeda.

AWS Control Tower membantu Anda menghemat waktu dengan mengotomatiskan pengaturan landing zone sehingga Anda dapat menjalankan beban kerja yang aman dan terukur. AWS Control Tower dikelola oleh AWS dan menggunakan praktik dan pedoman terbaik untuk membantu Anda menciptakan lingkungan dasar Anda. AWS Control Tower menggunakan layanan terintegrasi seperti [AWS Service Catalog](#) dan [AWS Organizations](#) untuk menyediakan akun di landing zone Anda dan mengelola akses ke akun tersebut.

Tujuan

Buat landing zone dengan konfigurasi awal untuk hal-hal berikut:

- Struktur akun
- Struktur jaringan
- Kerangka kerja identitas dan penagihan yang telah ditentukan sebelumnya
- Paket yang dapat dipilih pengguna yang telah ditentukan sebelumnya
- Kemampuan untuk menyesuaikan dan mengkonfigurasi

Hasil

- Sebuah landing zone yang ditetapkan dan aman siap untuk migrasi dan kustomisasi lebih lanjut

#### Panduan cara

- [Menyiapkan lingkungan AWS multi-akun yang aman dan dapat diskalakan](#)

#### Sumber daya terkait

- [AWS Control Tower](#)
- [AWS Service Catalog](#)
- [AWS Organizations](#)
- [Merancang keamanan & tata kelola di seluruh landing zone Anda](#) (AWS re:Invent 2019 presentation)

## Keamanan, risiko, dan kepatuhan

Alur kerja keamanan, risiko, dan kepatuhan mendefinisikan pendekatan terstruktur untuk membantu Anda membangun kepercayaan diri. AWS Ini juga memungkinkan kemampuan keamanan, risiko, dan kepatuhan dasar yang dapat mempercepat kesiapan dan perencanaan Anda untuk proyek migrasi. Pendekatan pengiriman dibangun di atas perspektif keamanan AWS CAF dan memberikan panduan yang lebih rinci untuk tim keamanan yang sedang mempersiapkan migrasi beban kerja bisnis ke AWS. Alur kerja ini memanfaatkan konsep pusat data virtual untuk mengatasi keamanan dasar minimum dan kontrol kepatuhan. Pusat data virtual dimaksudkan untuk dibangun melalui proses pengembangan tangkas menggunakan satu atau lebih tim pengiriman keamanan cloud.

#### Tujuan

Perspektif keamanan menyediakan konfigurasi awal yang direkomendasikan untuk hal-hal berikut:

- Model manajemen identitas dan akses
- Model pencatatan dan pemantauan
- Keamanan infrastruktur
- Perlindungan data
- Respons insiden

## Hasil

Buku pedoman yang dapat direferensikan yang dapat direferensikan, didukung oleh contoh kode yang relevan, dan mencakup lima topik inti berikut untuk tugas keamanan dan audit menggunakan Layanan AWS

- Manajemen identitas dan akses
- Kontrol detektif
- Keamanan infrastruktur
- Perlindungan data
- Respons insiden

## Operasi

Tujuan dari alur kerja operasi adalah untuk meninjau model operasional Anda saat ini dan mengembangkan pendekatan integrasi operasi untuk mendukung model operasi keadaan masa depan saat Anda bermigrasi. AWS Model operasi harus mencakup hubungan antara orang, proses, dan alat untuk mendukung penyampaian tujuan organisasi. Pemilik aliran kerja mengidentifikasi dan mendokumentasikan kesenjangan tingkat tinggi berdasarkan model operasional keadaan akhir untuk alat, proses, dan orang. Prioritas peta jalan kemudian dibuat untuk implementasi. Peta jalan ini dipengaruhi oleh, dan akan mempengaruhi, alur kerja migrasi lainnya karena banyaknya saling ketergantungan antara keamanan, orang, AWS landing zone, dan workstream proyek lainnya.

### Tujuan

Buat peta jalan untuk konstruksi operasional untuk diskalakan: AWS

- Identifikasi status manajemen layanan TI (ITSM) yang diinginkan dan model dukungan.
- Tinjau praktik operasional saat ini (alat, orang, proses) di tempat dan di cloud.
- Identifikasi potensi kerentanan seputar operasi penskalaan.
- Tinjau perencanaan kontinuitas bisnis (BCP), dan buat rencana untuk mengatasi dampak potensial pada operasi.
- Identifikasi bagaimana menjalankan migrasi akan berdampak pada operasi normal.
- Identifikasi organisasi pendukung operasional dan AWS Mitra yang akan berinteraksi dengan lingkungan cloud.

## Hasil

- Peningkatan postur operasional, dan peningkatan perjanjian tingkat layanan (SLAs) dan perjanjian tingkat operasional () OLAs
- Runbook dan panduan desain untuk silo operasional seperti backup, monitoring, dan deployment
- Buku pedoman operasional di AWS
- Buku pedoman planning/disaster pemulihan kelangsungan bisnis (BCP/DR)
- ITSM pada AWS didokumentasikan dan didefinisikan

## AWS Mitra dan alat

Saat aplikasi dimigrasikan dan sistem lama dihentikan, model operasi Anda menjadi seperangkat orang, proses, dan teknologi yang terus-menerus beralih ke model operasi modern. AWS Mitra seperti AppDynamics, New Relic, dan Dynatrace dapat membantu Anda melanjutkan iterasi pada model operasi Anda saat Anda memindahkan lebih banyak operasi ke cloud.

## Panduan cara

- [Modernisasi operasi di AWS Cloud](#)

## Orang: keterampilan, budaya, perubahan, dan kepemimpinan

Alur kerja ini sangat penting untuk membangun kesiapan migrasi dan menjalankan migrasi dalam skala besar. Dampak migrasi cloud akan dirasakan di seluruh organisasi dan secara signifikan akan mempengaruhi budaya organisasi. Selain itu, budaya organisasi Anda akan memengaruhi perjalanan Anda ke cloud. Implikasi budaya ini, penerimaan organisasi Anda terhadap perubahan, keberhasilan dan kegagalan perubahan sebelumnya, pola komunikasi organisasi, struktur organisasi, dan strategi pelatihan dan pemberdayaan karyawan yang ada adalah elemen penting dalam membangun pendekatan migrasi yang sukses. Agar siap menghadapi migrasi perusahaan, Anda harus memiliki banyak orang yang kritis dengan AWS pengalaman produksi dan proses operasional yang mapan. Anda juga harus memiliki Cloud Center of Excellence (CCoE) yang didedikasikan untuk memobilisasi sumber daya yang sesuai dan memimpin organisasi melalui banyak tantangan transformasi organisasi dan bisnis yang disajikan selama upaya migrasi skala besar.

## Tujuan

- Rancang tim yang bertanggung jawab untuk memobilisasi sumber daya cloud yang penting.

- Tentukan bagaimana organisasi membangun dan mengimplementasikan strategi cloud mereka dengan merancang tim untuk keadaan operasi masa depan.
- Membangun tim yang berdedikasi dengan kepemilikan single-threaded dan sponsor eksekutif yang kuat, terlihat, dan terlibat.
- Tetapkan area fungsional yang akan dikelola selama proses migrasi.
- Mulailah membuat model tata kelola cloud, seperangkat standar, praktik terbaik, dan prinsip atau prinsip panduan.
- Gunakan [AWS Change Acceleration 6-Point Framework dan Organizational Change Management Toolkit](#), yang menyediakan kit lengkap alat pemberdayaan untuk mendukung proyek adopsi cloud Anda.

## Hasil

- Ubah dokumen risiko manajemen
- Identifikasi dampak perubahan tingkat tinggi (berdasarkan peran dan proses utama)
- Pemetaan pemangku kepentingan utama
- Strategi dan platform pesan komunikasi
- Rencana komunikasi awal dan matriks pesan
- Ubah rencana kerja manajemen (awal)
- Piagam akselerasi organisasi
- Struktur adoption/acceleration tim orang (didokumentasikan dan dionboard)
- Definisi tujuan dan sasaran akselerasi organisasi
- Model kepegawaian negara masa depan (struktur organisasi sasaran)
- Ubah kartu skor risiko (manajemen risiko)
- Dokumen penyelarasan kepemimpinan
- Irama laporan pemangku kepentingan (penilaian pemangku kepentingan)
- Analisis dampak perubahan area, penilaian berbasis pemangku kepentingan, dan temuan dampak perubahan dan rekomendasi mitigasi
- Laporan penilaian kesiapan organisasi
- Ubah strategi
- Strategi komunikasi
- Strategi keterlibatan

- Strategi pelatihan
- Strategi mitigasi risiko
- Ubah peta jalan sponsor akselerasi

#### Panduan cara

- [Mempercepat adopsi cloud melalui budaya, perubahan, dan kepemimpinan](#)

## Fase migrasi

Migrasi pada skala terdiri dari blok bangunan, proses, alat, sumber daya, dan metodologi yang didefinisikan dan diuji selama tahap kesiapan dan perencanaan. Setelah menggunakan praktik terbaik dan pelajaran yang dipetik dari fase sebelumnya, Anda dapat menerapkan pabrik migrasi, yang merupakan cetak biru implementasi dan operasi penskalaan, melalui otomatisasi dan pengiriman tangkas.

## Pabrik migrasi

Pada fase scale-out proyek migrasi, Anda akan memiliki beberapa tim yang beroperasi secara bersamaan. Beberapa akan mendukung volume besar migrasi dalam rehost dan pola replatform kecil. Tim-tim ini disebut sebagai pabrik migrasi. Pabrik migrasi Anda akan meningkatkan kecepatan rencana migrasi Anda, dengan beberapa tim sprint bekerja secara paralel. 20-50 persen portofolio aplikasi perusahaan terdiri dari pola berulang yang dapat dioptimalkan dengan pendekatan pabrik. Ini adalah model pengiriman yang gesit, dan penting untuk membuat rencana manajemen rilis. Rencana Anda harus didasarkan pada beban kerja saat ini dan informasi yang dihasilkan selama tahap kesiapan dan perencanaan. Ini harus terus dioptimalkan untuk gelombang migrasi masa depan dan tim migrasi masa depan. Kami menyarankan Anda memiliki backlog aplikasi yang mendukung tiga sprint untuk setiap tim. Ini memungkinkan Anda untuk memprioritaskan kembali aplikasi jika Anda memiliki masalah yang memengaruhi jadwal.

Aplikasi yang lebih besar dan lebih kompleks sering mengikuti pola refactor/re-architect. Mereka umumnya dilakukan dalam siklus rilis yang direncanakan oleh pemilik aplikasi. Tim pabrik mandiri dan mencakup lima hingga enam peran lintas fungsi. Mereka termasuk operasi, analis bisnis dan pemilik, insinyur migrasi, pengembang, dan DevOps profesional. Berikut ini adalah contoh tim pabrik migrasi yang berfokus secara khusus:

- Tim migrasi rehost memigrasikan aplikasi bervolume tinggi dan kompleksitas rendah yang tidak memerlukan perubahan material. Tim-tim ini memanfaatkan alat otomatisasi migrasi. Pendekatan ini diintegrasikan ke dalam proses patch-and-release manajemen.
- Tim migrasi replatform merancang dan memigrasikan aplikasi yang memerlukan perubahan platform atau perubahan berulang dalam arsitektur aplikasi.
- Tim migrasi refactor/arsitek ulang merancang dan memigrasikan aplikasi bisnis kompleks atau inti yang memiliki banyak dependensi. Dalam kebanyakan kasus, tim pengembangan dan operasi teknis mendukung kemampuan bisnis ini. Migrasi menjadi siklus rilis atau beberapa siklus rilis

dalam rencana untuk tim itu. Banyak di antaranya dapat digunakan dalam penerbangan, dan Cloud Business Office (CBO) bertanggung jawab untuk melacak waktu, risiko, dan masalah melalui penyelesaian migrasi. Tim ini memiliki proses migrasi aplikasi.

Item yang perlu dipertimbangkan:

- Lakukan analisis portofolio untuk memahami pola umum di semua aplikasi untuk membantu membangun pekerjaan berulang agar tim pabrik dapat diterapkan secara efisien.
- Gunakan AWS Partner untuk membantu dengan kendala sumber daya karena tim Anda mendukung aktivitas bisnis reguler. AWS dan AWS Partner komunitas dapat membawa sumber daya khusus untuk topik tertentu seperti database, pengembangan aplikasi, dan perkakas migrasi.

Panduan cara

- [Mengotomatiskan migrasi server skala besar dengan Cloud Migration Factory](#)

## Langkah selanjutnya

Dalam dokumen strategi ini, kami memperkenalkan langkah-langkah persiapan dan implementasi yang diperlukan untuk migrasi besar ke AWS Cloud. Menganalisis keadaan Anda saat ini, membangun rencana, dan mengulangi pekerjaan memecah migrasi besar ke dalam aktivitas yang dapat dikelola untuk implementasi yang efisien. Melihat migrasi sebagai proyek perubahan organisasi membantu Anda membangun buy-in dan memelihara komunikasi melalui setiap tahap proses. Bangun kasus bisnis dan perbaiki laba atas investasi saat proyek berlangsung. Gunakan [AWS Cloud Adoption Framework \(AWS CAF\)](#) untuk menganalisis lingkungan Anda melalui berbagai perspektif bisnis, orang, tata kelola, platform, keamanan, dan operasi. Analisis ini memberi Anda gambaran lengkap tentang area mana yang perlu diperbaiki sebelum Anda bergerak maju dengan upaya migrasi yang besar.

Gunakan pabrik migrasi untuk membangun dan mengulangi pola migrasi untuk membuat pemindahan optimal ke AWS Cloud. Saat ini, AWS pelanggan telah beralih dari bertanya, “Mengapa bermigrasi ke cloud?” untuk bertanya, “Kapan?” Membangun strategi dan rencana migrasi yang efektif akan memungkinkan Anda untuk merespons, “Sekarang!”

Migrasi hanyalah awal dari apa yang mungkin. Setelah memigrasikan aplikasi, Anda dapat mempertimbangkan pengalaman migrasi sebagai kemampuan yang dapat Anda gunakan untuk fase pengoptimalan aplikasi ini. Anda akan memiliki arsitektur saat ini dan desain future. Anda akan menerapkan, menguji, dan memvalidasi perubahan. Anda akan memotong dan hidup. Anda sekarang memiliki kemampuan TI baru yang dapat mendorong kecepatan, kelincahan, dan nilai bisnis untuk organisasi dan perusahaan Anda.

# Sumber Daya

## Panduan

Panduan berikut di situs web AWS Prescriptive Guidance memberikan informasi lebih lanjut tentang aspek teknis dan non-teknis dalam mempercepat migrasi Anda ke cloud dalam skala besar.

- [Mengevaluasi kesiapan migrasi](#)
- [Panduan penilaian portofolio aplikasi untuk AWS Cloud migrasi](#)
- [Alat untuk bermigrasi ke AWS Cloud](#)
- [Menyiapkan program tangkas untuk mempercepat migrasi cloud](#)
- [Menyiapkan lingkungan AWS multi-akun yang aman dan terukur](#)
- [Modernisasi operasi di cloud](#)
- [Mempercepat adopsi cloud melalui budaya, perubahan, dan kepemimpinan](#)

## Pola

AWS Pola Panduan Preskriptif menyediakan alat, arsitektur, dan step-by-step panduan untuk menerapkan metodologi untuk strategi migrasi. Anda juga akan menemukan pola yang mendukung inisiatif lain, seperti proyek cloud-native atau keamanan. Anda dapat [melihat semua pola](#) atau menggunakan opsi pemfilteran dan bidang kueri pencarian di [situs web AWS Prescriptive Guidance](#) untuk menemukan pola yang menarik.

## Sumber daya tambahan

- [Migrasi dan Modernisasi dengan AWS](#)
- [Migrasi ke AWS: Praktik dan Strategi Terbaik](#)
- [Gambaran Umum Kerangka Adopsi AWS Cloud](#)
- [AWS Alat Penilaian](#)
- [AWS Penilaian Portofolio Migrasi \(MPA\)](#)
- [AWS Penilaian Kesiapan Cloud](#)
- Stephen Orban tentang migrasi cloud (posting AWS blog):

- [Mempertimbangkan Migrasi Massal ke Cloud?](#)
- [Proses Migrasi Massal ke Cloud](#)
- [Cloud-Native atau? Lift-and-Shift](#)
- [6 Strategi untuk Migrasi Aplikasi ke Cloud](#)
- [Ya, Anda Dapat Memigrasikan Mainframe Anda ke Cloud](#)
- [4 Alasan untuk Mempertimbangkan Kembali Lift-and-Shift ke Cloud](#) (artikel oleh Joe Chung di medium.com)
- [Memulai dengan AWS Database Migration Service](#) (presentasi di AWS Global Summit Series 2016, Chicago)

## Riwayat dokumen

Tabel berikut menjelaskan perubahan signifikan pada dokumen ini. Jika Anda ingin diberi tahu tentang pembaruan masa depan, Anda dapat berlangganan umpan [RSS](#).

Perubahan	Deskripsi	Tanggal
<a href="#">Penyegaran konten</a>	Informasi dan tautan yang diperbarui di seluruh publikasi.	Oktober 10, 2025
<a href="#">Konten yang diperbarui</a>	Memperbarui tujuan dalam <a href="#">alur kerja keamanan, risiko, dan kepatuhan</a> .	Juni 20, 2022
<a href="#">Konten yang diperbarui</a>	Memperbarui konten dalam topik <a href="#">fase Mobilisasi</a> untuk kejelasan tambahan.	24 Februari 2020
<a href="#">Publikasi awal</a>	—	5 Agustus 2019

# AWS Glosarium Panduan Preskriptif

Berikut ini adalah istilah yang umum digunakan dalam strategi, panduan, dan pola yang disediakan oleh Panduan AWS Preskriptif. Untuk menyarankan entri, silakan gunakan tautan Berikan umpan balik di akhir glosarium.

## Nomor

### 7 Rs

Tujuh strategi migrasi umum untuk memindahkan aplikasi ke cloud. Strategi ini dibangun di atas 5 Rs yang diidentifikasi Gartner pada tahun 2011 dan terdiri dari yang berikut:

- **Refactor/re-architect** — Pindahkan aplikasi dan modifikasi arsitekturnya dengan memanfaatkan sepenuhnya fitur cloud-native untuk meningkatkan kelincahan, kinerja, dan skalabilitas. Ini biasanya melibatkan porting sistem operasi dan database. Contoh: Migrasikan database Oracle lokal Anda ke Amazon Aurora Edition. PostgreSQL-Compatible
- **Replatform (angkat dan bentuk ulang)** — Pindahkan aplikasi ke cloud, dan perkenalkan beberapa tingkat pengoptimalan untuk memanfaatkan kemampuan cloud. Contoh: Memigrasikan database Oracle lokal Anda ke Amazon Relational Database Service (Amazon RDS) untuk Oracle di AWS Cloud
- **Pembelian kembali (drop and shop)** - Beralih ke produk yang berbeda, biasanya dengan beralih dari lisensi tradisional ke model SaaS. Contoh: Migrasikan sistem manajemen hubungan pelanggan (CRM) Anda ke Salesforce.com
- **Rehost (lift dan shift)** — Pindahkan aplikasi ke cloud tanpa membuat perubahan apa pun untuk memanfaatkan kemampuan cloud. Contoh: Migrasikan database Oracle lokal Anda ke Oracle pada instans EC2 di AWS Cloud
- **Relokasi (hypervisor-level lift and shift)** — Pindahkan infrastruktur ke cloud tanpa membeli perangkat keras baru, menulis ulang aplikasi, atau memodifikasi operasi yang ada. Anda memigrasikan server dari platform lokal ke layanan cloud untuk platform yang sama. Contoh: Migrasikan Microsoft Hyper-V aplikasi ke AWS.
- **Pertahankan (kunjungi kembali)** - Simpan aplikasi di lingkungan sumber Anda. Ini mungkin termasuk aplikasi yang memerlukan refactoring besar, dan Anda ingin menunda pekerjaan itu sampai nanti, dan aplikasi lama yang ingin Anda pertahankan, karena tidak ada pembenaran bisnis untuk memigrasikannya.

- Pensiun — Menonaktifkan atau menghapus aplikasi yang tidak lagi diperlukan di lingkungan sumber Anda.

## A

### A2A () Agent-to-Agent

Protokol stateful untuk kolaborasi agen-ke-agen yang mendukung delegasi tugas dan transfer negara.

### ABAC

Lihat [kontrol akses berbasis atribut](#).

### layanan abstrak

Lihat [layanan terkelola](#).

### ASAM

Lihat [atomisitas, konsistensi, isolasi, daya tahan](#).

### migrasi aktif-aktif

Metode migrasi database di mana basis data sumber dan target tetap sinkron (dengan menggunakan alat replikasi dua arah atau operasi penulisan ganda), dan kedua database menangani transaksi dari menghubungkan aplikasi selama migrasi. Metode ini mendukung migrasi dalam batch kecil yang terkontrol alih-alih memerlukan pemotongan satu kali. Ini lebih fleksibel tetapi membutuhkan lebih banyak pekerjaan daripada migrasi [aktif-pasif](#).

### migrasi aktif-pasif

Metode migrasi database di mana database sumber dan target disimpan dalam sinkron, tetapi hanya database sumber yang menangani transaksi dari menghubungkan aplikasi sementara data direplikasi ke database target. Basis data target tidak menerima transaksi apa pun selama migrasi.

### Agen

Sistem AI yang dapat secara mandiri bernalar, merencanakan, dan mengambil tindakan menggunakan alat untuk mencapai tujuan.

## Agen Ops

Praktik operasional untuk membangun, menguji, menyebarkan, dan menjalankan agen AI dalam produksi dalam skala besar.

### fungsi agregat

Fungsi SQL yang beroperasi pada sekelompok baris dan menghitung nilai pengembalian tunggal untuk grup. Contoh fungsi agregat meliputi SUM dan MAX.

## AI

Lihat [kecerdasan buatan](#).

### AIOps

Lihat [operasi kecerdasan buatan](#).

### anonimisasi

Proses menghapus informasi pribadi secara permanen dalam kumpulan data. Anonimisasi dapat membantu melindungi privasi pribadi. Data anonim tidak lagi dianggap sebagai data pribadi.

### anti-pola

Solusi yang sering digunakan untuk masalah berulang di mana solusinya kontra-produktif, tidak efektif, atau kurang efektif daripada alternatif.

### kontrol aplikasi

Pendekatan keamanan yang memungkinkan penggunaan hanya aplikasi yang disetujui untuk membantu melindungi sistem dari malware.

### portofolio aplikasi

Kumpulan informasi rinci tentang setiap aplikasi yang digunakan oleh organisasi, termasuk biaya untuk membangun dan memelihara aplikasi, dan nilai bisnisnya. Informasi ini adalah kunci untuk [penemuan portofolio dan proses analisis dan](#) membantu mengidentifikasi dan memprioritaskan aplikasi yang akan dimigrasi, dimodernisasi, dan dioptimalkan.

### kecerdasan buatan (AI)

Bidang ilmu komputer yang didedikasikan untuk menggunakan teknologi komputasi untuk melakukan fungsi kognitif yang biasanya terkait dengan manusia, seperti belajar, memecahkan masalah, dan mengenali pola. Untuk informasi lebih lanjut, lihat [Apa itu Kecerdasan Buatan?](#)

## operasi kecerdasan buatan (AIOps)

Proses menggunakan teknik pembelajaran mesin untuk memecahkan masalah operasional, mengurangi insiden operasional dan intervensi manusia, dan meningkatkan kualitas layanan. Untuk informasi selengkapnya tentang cara AIOps digunakan dalam strategi AWS migrasi, lihat [panduan integrasi operasi](#).

## enkripsi asimetris

Algoritma enkripsi yang menggunakan sepasang kunci, kunci publik untuk enkripsi dan kunci pribadi untuk dekripsi. Anda dapat berbagi kunci publik karena tidak digunakan untuk dekripsi, tetapi akses ke kunci pribadi harus sangat dibatasi.

## atomisitas, konsistensi, isolasi, daya tahan (ACID)

Satu set properti perangkat lunak yang menjamin validitas data dan keandalan operasional database, bahkan dalam kasus kesalahan, kegagalan daya, atau masalah lainnya.

## kontrol akses berbasis atribut (ABAC)

Praktik membuat izin berbutir halus berdasarkan atribut pengguna, seperti departemen, peran pekerjaan, dan nama tim. Untuk informasi selengkapnya, lihat [ABAC untuk AWS](#) dokumentasi AWS Identity and Access Management (IAM).

## sumber data otoritatif

Lokasi di mana Anda menyimpan versi utama data, yang dianggap sebagai sumber informasi yang paling dapat diandalkan. Anda dapat menyalin data dari sumber data otoritatif ke lokasi lain untuk tujuan memproses atau memodifikasi data, seperti menganonimkan, menyunting, atau membuat nama samaran.

## Zona Ketersediaan

Lokasi berbeda di dalam Wilayah AWS yang terisolasi dari kegagalan di Availability Zone lainnya dan menyediakan konektivitas jaringan latensi rendah yang murah ke Availability Zone lainnya di Wilayah yang sama.

## AWS Kerangka Adopsi Cloud (AWS CAF)

Kerangka pedoman dan praktik terbaik AWS untuk membantu organisasi mengembangkan rencana yang efisien dan efektif untuk bergerak dengan sukses ke cloud. AWS CAF mengatur panduan ke dalam enam area fokus yang disebut perspektif: bisnis, orang, tata kelola, platform, keamanan, dan operasi. Perspektif bisnis, orang, dan tata kelola fokus pada keterampilan dan

proses bisnis; perspektif platform, keamanan, dan operasi fokus pada keterampilan dan proses teknis. Misalnya, perspektif masyarakat menargetkan pemangku kepentingan yang menangani sumber daya manusia (SDM), fungsi kepegawaian, dan manajemen orang. Untuk perspektif ini, AWS CAF memberikan panduan untuk pengembangan, pelatihan, dan komunikasi orang untuk membantu mempersiapkan organisasi untuk adopsi cloud yang sukses. Untuk informasi lebih lanjut, lihat [situs web AWS CAF](#) dan [whitepaper AWS CAF](#).

## AWS Kerangka Kualifikasi Beban Kerja (AWS WQF)

Alat yang mengevaluasi beban kerja migrasi database, merekomendasikan strategi migrasi, dan memberikan perkiraan kerja. AWS WQF disertakan dengan AWS Schema Conversion Tool (AWS SCT). Ini menganalisis skema database dan objek kode, kode aplikasi, dependensi, dan karakteristik kinerja, dan memberikan laporan penilaian.

## B

### bot buruk

[Bot](#) yang dimaksudkan untuk mengganggu atau menyebabkan kerugian bagi individu atau organisasi.

### BCP

Lihat [perencanaan kontinuitas bisnis](#).

### grafik perilaku

Pandangan interaktif yang terpadu tentang perilaku dan interaksi sumber daya dari waktu ke waktu. Anda dapat menggunakan grafik perilaku dengan Amazon Detective untuk memeriksa upaya logon yang gagal, panggilan API yang mencurigakan, dan tindakan serupa. Untuk informasi selengkapnya, lihat [Data dalam grafik perilaku](#) di dokumentasi Detektif.

### sistem big-endian

Sistem yang menyimpan byte paling signifikan terlebih dahulu. Lihat juga [endianness](#).

### klasifikasi biner

Sebuah proses yang memprediksi hasil biner (salah satu dari dua kelas yang mungkin). Misalnya, model ML Anda mungkin perlu memprediksi masalah seperti “Apakah email ini spam atau bukan spam?” atau “Apakah produk ini buku atau mobil?”

## filter mekar

Struktur data probabilistik dan efisien memori yang digunakan untuk menguji apakah suatu elemen adalah anggota dari suatu himpunan.

## blue/green penyebaran

Strategi penyebaran tempat Anda membuat dua lingkungan yang terpisah namun identik. Anda menjalankan versi aplikasi saat ini di satu lingkungan (biru) dan versi aplikasi baru di lingkungan lain (hijau). Strategi ini membantu Anda dengan cepat memutar kembali dengan dampak minimal.

## bot

Aplikasi perangkat lunak yang menjalankan tugas otomatis melalui internet dan mensimulasikan aktivitas atau interaksi manusia. Beberapa bot berguna atau bermanfaat, seperti perayap web yang mengindeks informasi di internet. Beberapa bot lain, yang dikenal sebagai bot buruk, dimaksudkan untuk mengganggu atau membahayakan individu atau organisasi.

## botnet

Jaringan [bot](#) yang terinfeksi oleh [malware](#) dan berada di bawah kendali satu pihak, yang dikenal sebagai bot herder atau operator bot. Botnet adalah mekanisme paling terkenal untuk skala bot dan dampaknya.

## cabang

Area berisi repositori kode. Cabang pertama yang dibuat dalam repositori adalah cabang utama. Anda dapat membuat cabang baru dari cabang yang ada, dan Anda kemudian dapat mengembangkan fitur atau memperbaiki bug di cabang baru. Cabang yang Anda buat untuk membangun fitur biasanya disebut sebagai cabang fitur. Saat fitur siap dirilis, Anda menggabungkan cabang fitur kembali ke cabang utama. Untuk informasi selengkapnya, lihat [Tentang cabang](#) (GitHub dokumentasi).

## akses break-glass

Dalam keadaan luar biasa dan melalui proses yang disetujui, cara cepat bagi pengguna untuk mendapatkan akses ke Akun AWS yang biasanya tidak memiliki izin untuk mengaksesnya. Untuk informasi lebih lanjut, lihat indikator [Implementasikan prosedur break-glass](#) dalam panduan. AWS Well-Architected

## strategi brownfield

Infrastruktur yang ada di lingkungan Anda. Saat mengadopsi strategi brownfield untuk arsitektur sistem, Anda merancang arsitektur di sekitar kendala sistem dan infrastruktur saat ini. Jika Anda memperluas infrastruktur yang ada, Anda dapat memadukan strategi brownfield dan [greenfield](#).

## cache penyangga

Area memori tempat data yang paling sering diakses disimpan.

## kemampuan bisnis

Apa yang dilakukan bisnis untuk menghasilkan nilai (misalnya, penjualan, layanan pelanggan, atau pemasaran). Arsitektur layanan mikro dan keputusan pengembangan dapat didorong oleh kemampuan bisnis. Untuk informasi selengkapnya, lihat bagian [Terorganisir di sekitar kemampuan bisnis](#) dari [Menjalankan layanan mikro kontainer](#) di whitepaper. AWS

## perencanaan kelangsungan bisnis (BCP)

Rencana yang membahas dampak potensial dari peristiwa yang mengganggu, seperti migrasi skala besar, pada operasi dan memungkinkan bisnis untuk melanjutkan operasi dengan cepat.

# C

## KAFE

Lihat [Kerangka Adopsi AWS Cloud](#).

## penyebaran kenari

Rilis versi yang lambat dan bertahap untuk pengguna akhir. Ketika Anda yakin, Anda menyebarkan versi baru dan mengganti versi saat ini secara keseluruhan.

## CCoE

Lihat [Cloud Center of Excellence](#).

## CDC

Lihat [mengubah pengambilan data](#).

## ubah pengambilan data (CDC)

Proses melacak perubahan ke sumber data, seperti tabel database, dan merekam metadata tentang perubahan tersebut. Anda dapat menggunakan CDC untuk berbagai tujuan, seperti mengaudit atau mereplikasi perubahan dalam sistem target untuk mempertahankan sinkronisasi.

## rekayasa kekacauan

Sengaja memperkenalkan kegagalan atau peristiwa yang mengganggu untuk menguji ketahanan sistem. Anda dapat menggunakan [AWS Fault Injection Service \(AWS FIS\)](#) untuk melakukan eksperimen yang menekankan AWS beban kerja Anda dan mengevaluasi responsnya.

## CI/CD

Lihat [integrasi berkelanjutan dan pengiriman berkelanjutan](#).

## klasifikasi

Proses kategorisasi yang membantu menghasilkan prediksi. Model ML untuk masalah klasifikasi memprediksi nilai diskrit. Nilai diskrit selalu berbeda satu sama lain. Misalnya, model mungkin perlu mengevaluasi apakah ada mobil dalam gambar atau tidak.

## Pengembang Warga

Pengguna bisnis yang membuat aplikasi AI menggunakan platform tanpa code/low kode tanpa keterampilan teknis khusus.

## Enkripsi sisi klien

Enkripsi data secara lokal, sebelum target Layanan AWS menerimanya.

## Cloud Center of Excellence (CCoE)

Tim multi-disiplin yang mendorong upaya adopsi cloud di seluruh organisasi, termasuk mengembangkan praktik terbaik cloud, memobilisasi sumber daya, menetapkan jadwal migrasi, dan memimpin organisasi melalui transformasi skala besar. Untuk informasi selengkapnya, lihat [posting CCoE](#) di Blog Strategi AWS Cloud Perusahaan.

## komputasi cloud

Teknologi cloud yang biasanya digunakan untuk penyimpanan data jarak jauh dan manajemen perangkat IoT. Cloud computing umumnya terhubung ke teknologi [edge computing](#).

## model operasi cloud

Dalam organisasi TI, model operasi yang digunakan untuk membangun, mematangkan, dan mengoptimalkan satu atau lebih lingkungan cloud. Untuk informasi selengkapnya, lihat [Membangun Model Operasi Cloud Anda](#).

## tahap adopsi cloud

Empat fase yang biasanya dilalui organisasi ketika mereka bermigrasi ke AWS Cloud:

- Proyek — Menjalankan beberapa proyek terkait cloud untuk bukti konsep dan tujuan pembelajaran
- Foundation — Melakukan investasi dasar untuk meningkatkan adopsi cloud Anda (misalnya, membuat landing zone, mendefinisikan CCoE, membuat model operasi)
- Migrasi — Migrasi aplikasi individual
- Re-invention — Mengoptimalkan produk dan layanan, dan berinovasi di cloud

Tahapan ini didefinisikan oleh Stephen Orban dalam posting blog [The Journey Toward Cloud-First & the Stages of Adoption](#) di blog Strategi AWS Cloud Perusahaan. Untuk informasi tentang bagaimana kaitannya dengan strategi AWS migrasi, lihat [panduan kesiapan migrasi](#).

## CMDB

Lihat [database manajemen konfigurasi](#).

## repositori kode

Lokasi di mana kode sumber dan aset lainnya, seperti dokumentasi, sampel, dan skrip, disimpan dan diperbarui melalui proses kontrol versi. Repositori cloud umum termasuk GitHub atau Bitbucket Cloud. Setiap versi kode disebut cabang. Dalam struktur layanan mikro, setiap repositori dikhususkan untuk satu bagian fungsionalitas. Satu CI/CD pipa dapat menggunakan beberapa repositori.

## cache dingin

Cache buffer yang kosong, tidak terisi dengan baik, atau berisi data basi atau tidak relevan. Ini mempengaruhi kinerja karena instance database harus membaca dari memori utama atau disk, yang lebih lambat daripada membaca dari cache buffer.

## data dingin

Data yang jarang diakses dan biasanya historis. Saat menanyakan jenis data ini, kueri lambat biasanya dapat diterima. Memindahkan data ini ke tingkat penyimpanan atau kelas yang berkinerja lebih rendah dan lebih murah dapat mengurangi biaya.

## visi komputer (CV)

Bidang [AI](#) yang menggunakan pembelajaran mesin untuk menganalisis dan mengekstrak informasi dari format visual seperti gambar dan video digital. Misalnya, Amazon SageMaker AI menyediakan algoritma pemrosesan gambar untuk CV.

## konfigurasi drift

Untuk beban kerja, konfigurasi berubah dari status yang diharapkan. Ini dapat menyebabkan beban kerja menjadi tidak patuh, dan biasanya bertahap dan tidak disengaja.

## database manajemen konfigurasi (CMDB)

Repositori yang menyimpan dan mengelola informasi tentang database dan lingkungan TI, termasuk komponen perangkat keras dan perangkat lunak dan konfigurasinya. Anda biasanya menggunakan data dari CMDB dalam penemuan portofolio dan tahap analisis migrasi.

## paket kesesuaian

Kumpulan AWS Config aturan dan tindakan remediasi yang dapat Anda kumpulkan untuk menyesuaikan kepatuhan dan pemeriksaan keamanan Anda. Anda dapat menerapkan paket kesesuaian sebagai entitas tunggal di Akun AWS dan Region, atau di seluruh organisasi, dengan menggunakan templat YAMM. Untuk informasi selengkapnya, lihat [Paket kesesuaian dalam dokumentasi](#). AWS Config

## integrasi berkelanjutan dan pengiriman berkelanjutan (CI/CD)

Proses mengotomatiskan sumber, membangun, menguji, pementasan, dan tahap produksi dari proses rilis perangkat lunak. CI/CD biasanya digambarkan sebagai pipa. CI/CD dapat membantu Anda mengotomatiskan proses, meningkatkan produktivitas, meningkatkan kualitas kode, dan memberikan lebih cepat. Untuk informasi lebih lanjut, lihat [Manfaat pengiriman berkelanjutan](#). CD juga dapat berarti penerapan berkelanjutan. Untuk informasi selengkapnya, lihat [Continuous Delivery vs Continuous Deployment](#).

## CV

Lihat [visi komputer](#).

## D

### data saat istirahat

Data yang stasioner di jaringan Anda, seperti data yang ada di penyimpanan.

## klasifikasi data

Proses untuk mengidentifikasi dan mengkategorikan data dalam jaringan Anda berdasarkan kekritisannya dan sensitivitasnya. Ini adalah komponen penting dari setiap strategi manajemen risiko keamanan siber karena membantu Anda menentukan perlindungan dan kontrol retensi yang tepat untuk data. Klasifikasi data adalah komponen pilar keamanan dalam AWS Well-Architected Framework. Untuk informasi selengkapnya, lihat [Klasifikasi data](#).

## penyimpangan data

Variasi yang berarti antara data produksi dan data yang digunakan untuk melatih model ML, atau perubahan yang berarti dalam data input dari waktu ke waktu. Penyimpangan data dapat mengurangi kualitas, akurasi, dan keadilan keseluruhan dalam prediksi model ML.

## data dalam transit

Data yang aktif bergerak melalui jaringan Anda, seperti antara sumber daya jaringan.

## jala data

Kerangka arsitektur yang menyediakan kepemilikan data terdistribusi dan terdesentralisasi dengan manajemen dan tata kelola terpusat.

## minimalisasi data

Prinsip pengumpulan dan pemrosesan hanya data yang sangat diperlukan. Mempraktikkan minimalisasi data di dalamnya AWS Cloud dapat mengurangi risiko privasi, biaya, dan jejak karbon analitik Anda.

## perimeter data

Satu set pagar pembatas pencegahan di AWS lingkungan Anda yang membantu memastikan bahwa hanya identitas tepercaya yang mengakses sumber daya tepercaya dari jaringan yang diharapkan. Untuk informasi selengkapnya, lihat [Membangun perimeter data pada AWS](#).

## prapemrosesan data

Untuk mengubah data mentah menjadi format yang mudah diuraikan oleh model ML Anda. Preprocessing data dapat berarti menghapus kolom atau baris tertentu dan menangani nilai yang hilang, tidak konsisten, atau duplikat.

## asal data

Proses melacak asal dan riwayat data sepanjang siklus hidupnya, seperti bagaimana data dihasilkan, ditransmisikan, dan disimpan.

## subjek data

Individu yang datanya dikumpulkan dan diproses.

## gudang data

Sistem manajemen data yang mendukung intelijen bisnis, seperti analitik. Gudang data biasanya berisi sejumlah besar data historis, dan biasanya digunakan untuk kueri dan analisis.

## bahasa definisi database (DDL)

Pernyataan atau perintah untuk membuat atau memodifikasi struktur tabel dan objek dalam database.

## bahasa manipulasi basis data (DHTML)

Pernyataan atau perintah untuk memodifikasi (memasukkan, memperbarui, dan menghapus) informasi dalam database.

## DDL

Lihat [bahasa definisi database](#).

## ansambel yang dalam

Untuk menggabungkan beberapa model pembelajaran mendalam untuk prediksi. Anda dapat menggunakan ansambel dalam untuk mendapatkan prediksi yang lebih akurat atau untuk memperkirakan ketidakpastian dalam prediksi.

## pembelajaran mendalam

Subbidang ML yang menggunakan beberapa lapisan jaringan saraf tiruan untuk mengidentifikasi pemetaan antara data input dan variabel target yang diinginkan.

## pertahanan-mendalam

Pendekatan keamanan informasi di mana serangkaian mekanisme dan kontrol keamanan dilapisi dengan cermat di seluruh jaringan komputer untuk melindungi kerahasiaan, integritas, dan ketersediaan jaringan dan data di dalamnya. Saat Anda mengadopsi strategi ini AWS, Anda menambahkan beberapa kontrol pada lapisan AWS Organizations struktur yang berbeda untuk membantu mengamankan sumber daya. Misalnya, pendekatan defense-in-depth mungkin menggabungkan otentikasi multi-faktor, segmentasi jaringan, dan enkripsi.

## administrator yang didelegasikan

Di AWS Organizations, layanan yang kompatibel dapat mendaftarkan akun AWS anggota untuk mengelola akun organisasi dan mengelola izin untuk layanan tersebut. Akun ini disebut

administrator yang didelegasikan untuk layanan itu. Untuk informasi selengkapnya dan daftar layanan yang kompatibel, lihat [Layanan yang berfungsi dengan AWS Organizations](#) AWS Organizations dokumentasi.

## deployment

Proses pembuatan aplikasi, fitur baru, atau perbaikan kode tersedia di lingkungan target. Deployment melibatkan penerapan perubahan dalam basis kode dan kemudian membangun dan menjalankan basis kode itu di lingkungan aplikasi.

## lingkungan pengembangan

Lihat [lingkungan](#).

## kontrol detektif

Kontrol keamanan yang dirancang untuk mendeteksi, mencatat, dan memperingatkan setelah suatu peristiwa terjadi. Kontrol ini adalah garis pertahanan kedua, memperingatkan Anda tentang peristiwa keamanan yang melewati kontrol pencegahan yang ada. Untuk informasi selengkapnya, lihat Kontrol [Detektif dalam Menerapkan kontrol](#) keamanan pada. AWS

## pemetaan aliran nilai pengembangan (DVSM)

Sebuah proses yang digunakan untuk mengidentifikasi dan memprioritaskan kendala yang mempengaruhi kecepatan dan kualitas dalam siklus hidup pengembangan perangkat lunak. DVSM memperluas proses pemetaan aliran nilai yang awalnya dirancang untuk praktik manufaktur ramping. Ini berfokus pada langkah-langkah dan tim yang diperlukan untuk menciptakan dan memindahkan nilai melalui proses pengembangan perangkat lunak.

## kembar digital

Representasi virtual dari sistem dunia nyata, seperti bangunan, pabrik, peralatan industri, atau jalur produksi. Kembar digital mendukung pemeliharaan prediktif, pemantauan jarak jauh, dan optimalisasi produksi.

## tabel dimensi

Dalam [skema bintang](#), tabel yang lebih kecil yang berisi atribut data tentang data kuantitatif dalam tabel fakta. Atribut tabel dimensi biasanya bidang teks atau angka diskrit yang berperilaku seperti teks. Atribut ini biasanya digunakan untuk pembatasan kueri, pemfilteran, dan pelabelan set hasil.

## musibah

Peristiwa yang mencegah beban kerja atau sistem memenuhi tujuan bisnisnya di lokasi utama yang digunakan. Peristiwa ini dapat berupa bencana alam, kegagalan teknis, atau akibat dari tindakan manusia, seperti kesalahan konfigurasi yang tidak disengaja atau serangan malware.

## pemulihan bencana (DR)

Strategi dan proses yang Anda gunakan untuk meminimalkan downtime dan kehilangan data yang disebabkan oleh [bencana](#). Untuk informasi selengkapnya, lihat [Disaster Recovery of Workloads on AWS: Recovery in the Cloud](#) in the AWS Well-Architected Framework.

## DML~

Lihat [bahasa manipulasi database](#).

## desain berbasis domain

Pendekatan untuk mengembangkan sistem perangkat lunak yang kompleks dengan menghubungkan komponennya ke domain yang berkembang, atau tujuan bisnis inti, yang dilayani setiap komponen. Konsep ini diperkenalkan oleh Eric Evans dalam bukunya, *Domain-Driven Design: Tackling Complexity in the Heart of Software* (Boston: Addison-Wesley Professional, 2003). Untuk informasi tentang cara menggunakan desain berbasis domain dengan pola gambar pencekk, lihat Memodernisasi layanan [web Microsoft ASP.NET \(ASMX\) lama](#) secara bertahap menggunakan container dan Amazon API Gateway.

## DR

Lihat [pemulihan bencana](#).

## deteksi drift

Melacak penyimpangan dari konfigurasi dasar. Misalnya, Anda dapat menggunakan AWS CloudFormation untuk [mendeteksi penyimpangan dalam sumber daya sistem](#), atau Anda dapat menggunakannya AWS Control Tower untuk [mendeteksi perubahan di landing zone](#) yang mungkin memengaruhi kepatuhan terhadap persyaratan tata kelola.

## DVSM

Lihat [pemetaan aliran nilai pengembangan](#).

## E

### EDA

Lihat [analisis data eksplorasi](#).

### EDI

Lihat [pertukaran data elektronik](#).

### komputasi tepi

Teknologi yang meningkatkan daya komputasi untuk perangkat pintar di tepi jaringan IoT. Jika dibandingkan dengan [komputasi awan](#), komputasi tepi dapat mengurangi latensi komunikasi dan meningkatkan waktu respons.

### pertukaran data elektronik (EDI)

Pertukaran otomatis dokumen bisnis antar organisasi. Untuk informasi selengkapnya, lihat [Apa itu Pertukaran Data Elektronik](#).

### enkripsi

Proses komputasi yang mengubah data plaintext, yang dapat dibaca manusia, menjadi ciphertext.

### kunci enkripsi

String kriptografi dari bit acak yang dihasilkan oleh algoritma enkripsi. Panjang kunci dapat bervariasi, dan setiap kunci dirancang agar tidak dapat diprediksi dan unik.

### endianness

Urutan byte disimpan dalam memori komputer. Big-endian sistem menyimpan byte paling signifikan terlebih dahulu. Little-endian sistem menyimpan byte paling tidak signifikan terlebih dahulu.

### titik akhir

Lihat [titik akhir layanan](#).

### layanan endpoint

Layanan yang dapat Anda host di cloud pribadi virtual (VPC) untuk dibagikan dengan pengguna lain. Anda dapat membuat layanan endpoint dengan AWS PrivateLink dan memberikan izin

kepada prinsipal lain Akun AWS atau ke AWS Identity and Access Management (IAM). Akun atau prinsipal ini dapat terhubung ke layanan endpoint Anda secara pribadi dengan membuat titik akhir VPC antarmuka. Untuk informasi selengkapnya, lihat [Membuat layanan titik akhir](#) di dokumentasi Amazon Virtual Private Cloud (Amazon VPC).

## perencanaan sumber daya perusahaan (ERP)

Sistem yang mengotomatiskan dan mengelola proses bisnis utama (seperti akuntansi, [MES](#), dan manajemen proyek) untuk suatu perusahaan.

## enkripsi amplop

Proses mengenkripsi kunci enkripsi dengan kunci enkripsi lain. Untuk informasi selengkapnya, lihat [Enkripsi amplop](#) dalam dokumentasi AWS Key Management Service (AWS KMS).

## lingkungan

Sebuah contoh dari aplikasi yang sedang berjalan. Berikut ini adalah jenis lingkungan yang umum dalam komputasi awan:

- Development Environment — Sebuah contoh dari aplikasi yang berjalan yang hanya tersedia untuk tim inti yang bertanggung jawab untuk memelihara aplikasi. Lingkungan pengembangan digunakan untuk menguji perubahan sebelum mempromosikannya ke lingkungan atas. Jenis lingkungan ini kadang-kadang disebut sebagai lingkungan pengujian.
- lingkungan yang lebih rendah — Semua lingkungan pengembangan untuk aplikasi, seperti yang digunakan untuk build awal dan pengujian.
- lingkungan produksi — Sebuah contoh dari aplikasi yang berjalan yang dapat diakses oleh pengguna akhir. Dalam sebuah CI/CD pipeline, lingkungan produksi adalah lingkungan penyebaran terakhir.
- lingkungan atas — Semua lingkungan yang dapat diakses oleh pengguna selain tim pengembangan inti. Ini dapat mencakup lingkungan produksi, lingkungan praproduksi, dan lingkungan untuk pengujian penerimaan pengguna.

## epik

Dalam metodologi tangkas, kategori fungsional yang membantu mengatur dan memprioritaskan pekerjaan Anda. Epik memberikan deskripsi tingkat tinggi tentang persyaratan dan tugas implementasi. Misalnya, epos keamanan AWS CAF mencakup manajemen identitas dan akses, kontrol detektif, keamanan infrastruktur, perlindungan data, dan respons insiden. Untuk informasi selengkapnya tentang epos dalam strategi AWS migrasi, lihat [panduan implementasi program](#).

## ERP

Lihat [perencanaan sumber daya perusahaan](#).

### analisis data eksplorasi (EDA)

Proses menganalisis dataset untuk memahami karakteristik utamanya. Anda mengumpulkan atau mengumpulkan data dan kemudian melakukan penyelidikan awal untuk menemukan pola, mendeteksi anomali, dan memeriksa asumsi. EDA dilakukan dengan menghitung statistik ringkasan dan membuat visualisasi data.

## F

### tabel fakta

Tabel tengah dalam [skema bintang](#). Ini menyimpan data kuantitatif tentang operasi bisnis. Biasanya, tabel fakta berisi dua jenis kolom: kolom yang berisi ukuran dan yang berisi kunci asing ke tabel dimensi.

### gagal cepat

Filosofi yang menggunakan pengujian yang sering dan bertahap untuk mengurangi siklus hidup pengembangan. Ini adalah bagian penting dari pendekatan tangkas.

### batas isolasi kesalahan

Dalam AWS Cloud, batas seperti Availability Zone, Wilayah AWS, control plane, atau data plane yang membatasi efek kegagalan dan membantu meningkatkan ketahanan beban kerja. Untuk informasi selengkapnya, lihat [Batas Isolasi AWS Kesalahan](#).

### cabang fitur

Lihat [cabang](#).

### fitur

Data input yang Anda gunakan untuk membuat prediksi. Misalnya, dalam konteks manufaktur, fitur bisa berupa gambar yang diambil secara berkala dari lini manufaktur.

### pentingnya fitur

Seberapa signifikan fitur untuk prediksi model. Ini biasanya dinyatakan sebagai skor numerik yang dapat dihitung melalui berbagai teknik, seperti Shapley Additive Explanations (SHAP) dan gradien

terintegrasi. Untuk informasi lebih lanjut, lihat [Interpretabilitas model pembelajaran mesin](#) dengan AWS

## transformasi fitur

Untuk mengoptimalkan data untuk proses ML, termasuk memperkaya data dengan sumber tambahan, menskalakan nilai, atau mengekstrak beberapa set informasi dari satu bidang data. Hal ini memungkinkan model ML untuk mendapatkan keuntungan dari data. Misalnya, jika Anda memecah tanggal "2021-05-27 00:15:37" menjadi "2021", "Mei", "Kamis", dan "15", Anda dapat membantu algoritme pembelajaran mempelajari pola bernuansa yang terkait dengan komponen data yang berbeda.

## beberapa tembakan mendorong

Menyediakan [LLM](#) dengan sejumlah kecil contoh yang menunjukkan tugas dan output yang diinginkan sebelum memintanya untuk melakukan tugas serupa. Teknik ini adalah aplikasi pembelajaran dalam konteks, di mana model belajar dari contoh (bidikan) yang tertanam dalam petunjuk. Few-shot prompt bisa efektif untuk tugas-tugas yang membutuhkan pemformatan, penalaran, atau pengetahuan domain tertentu. Lihat juga [bidikan nol](#).

## FGAC

Lihat kontrol [akses berbutir halus](#).

## kontrol akses berbutir halus (FGAC)

Penggunaan beberapa kondisi untuk mengizinkan atau menolak permintaan akses.

## migrasi flash-cut

Metode migrasi database yang menggunakan replikasi data berkelanjutan melalui [pengambilan data perubahan](#) untuk memigrasikan data dalam waktu sesingkat mungkin, alih-alih menggunakan pendekatan bertahap. Tujuannya adalah untuk menjaga downtime seminimal mungkin.

## FM

Lihat [model pondasi](#).

## model pondasi (FM)

Jaringan saraf pembelajaran mendalam yang besar yang telah melatih kumpulan data besar-besaran data umum dan tidak berlabel. FM mampu melakukan berbagai tugas umum, seperti memahami bahasa, menghasilkan teks dan gambar, dan berbicara dalam bahasa alami. Untuk informasi selengkapnya, lihat [Apa itu Model Foundation](#).

## Gerbang FM

[Perantara terpusat yang mengontrol dan menormalkan akses ke model pondasi.](#) Juga dikenal sebagai gateway LLM.

## G

### AI generatif

Subset model [AI](#) yang telah dilatih pada sejumlah besar data dan yang dapat menggunakan prompt teks sederhana untuk membuat konten dan artefak baru, seperti gambar, video, teks, dan audio. Untuk informasi lebih lanjut, lihat [Apa itu AI Generatif](#).

### pemblokiran geografis

Lihat [pembatasan geografis](#).

### pembatasan geografis (pemblokiran geografis)

Di Amazon CloudFront, opsi untuk mencegah pengguna di negara tertentu mengakses distribusi konten. Anda dapat menggunakan daftar izinkan atau daftar blokir untuk menentukan negara yang disetujui dan dilarang. Untuk informasi selengkapnya, lihat [Membatasi distribusi geografis konten Anda](#) dalam dokumentasi CloudFront

### Alur kerja Gitflow

Pendekatan di mana lingkungan bawah dan atas menggunakan cabang yang berbeda dalam repositori kode sumber. Alur kerja Gitflow dianggap warisan, dan [alur kerja berbasis batang](#) adalah pendekatan modern yang lebih disukai.

### gambar emas

Sebuah snapshot dari sistem atau perangkat lunak yang digunakan sebagai template untuk menyebarkan instance baru dari sistem atau perangkat lunak itu. Misalnya, di bidang manufaktur, gambar emas dapat digunakan untuk menyediakan perangkat lunak pada beberapa perangkat dan membantu meningkatkan kecepatan, skalabilitas, dan produktivitas dalam operasi manufaktur perangkat.

### strategi greenfield

Tidak adanya infrastruktur yang ada di lingkungan baru. [Saat mengadopsi strategi greenfield untuk arsitektur sistem, Anda dapat memilih semua teknologi baru tanpa batasan kompatibilitas](#)

[dengan infrastruktur yang ada, juga dikenal sebagai brownfield.](#) Jika Anda memperluas infrastruktur yang ada, Anda dapat memadukan strategi brownfield dan greenfield.

## pagar pembatas

Aturan tingkat tinggi yang membantu mengatur sumber daya, kebijakan, dan kepatuhan di seluruh unit organisasi (OU). Pagar pembatas preventif menegakkan kebijakan untuk memastikan keselarasan dengan standar kepatuhan. Mereka diimplementasikan dengan menggunakan kebijakan kontrol layanan dan batas izin IAM. Detective guardrails mendeteksi pelanggaran kebijakan dan masalah kepatuhan, dan menghasilkan peringatan untuk remediasi. Mereka diimplementasikan dengan menggunakan AWS Config, AWS Security Hub CSPM, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector, dan pemeriksaan khusus AWS Lambda .

## pagar pembatas (AI)

Mekanisme keamanan yang menyaring, memvalidasi, dan membatasi input dan output [agen](#) untuk membantu memastikan perilaku AI yang bertanggung jawab dan aman.

# H

## HA

Lihat [ketersediaan tinggi](#).

## migrasi database heterogen

Memigrasi database sumber Anda ke database target yang menggunakan mesin database yang berbeda (misalnya, Oracle ke Amazon Aurora). Migrasi heterogen biasanya merupakan bagian dari upaya arsitektur ulang, dan mengubah skema dapat menjadi tugas yang kompleks. [AWS menyediakan AWS SCT](#) yang membantu dengan konversi skema.

## ketersediaan tinggi (HA)

Kemampuan beban kerja untuk beroperasi terus menerus, tanpa intervensi, jika terjadi tantangan atau bencana. Sistem HA dirancang untuk gagal secara otomatis, secara konsisten memberikan kinerja berkualitas tinggi, dan menangani beban dan kegagalan yang berbeda dengan dampak kinerja minimal.

## modernisasi sejarawan

Pendekatan yang digunakan untuk memodernisasi dan meningkatkan sistem teknologi operasional (OT) untuk melayani kebutuhan industri manufaktur dengan lebih baik. Sejarawan

adalah jenis database yang digunakan untuk mengumpulkan dan menyimpan data dari berbagai sumber di pabrik.

#### data penahanan

Sebagian dari data historis berlabel yang ditahan dari kumpulan data yang digunakan untuk melatih model pembelajaran [mesin](#). Anda dapat menggunakan data penahanan untuk mengevaluasi kinerja model dengan membandingkan prediksi model dengan data penahanan.

#### manusia-dalam-lingkaran (HiTL)

Pola alur kerja di mana eksekusi [agen](#) berhenti untuk peninjauan dan persetujuan manusia pada titik keputusan kritis.

#### migrasi database homogen

Memigrasi database sumber Anda ke database target yang berbagi mesin database yang sama (misalnya, Microsoft SQL Server ke Amazon RDS for SQL Server). Migrasi homogen biasanya merupakan bagian dari upaya rehosting atau replatforming. Anda dapat menggunakan utilitas database asli untuk memigrasi skema.

#### data panas

Data yang sering diakses, seperti data real-time atau data translasi terbaru. Data ini biasanya memerlukan tingkat atau kelas penyimpanan berkinerja tinggi untuk memberikan respons kueri yang cepat.

#### perbaikan terbaru

Perbaikan mendesak untuk masalah kritis dalam lingkungan produksi. Karena urgensinya, perbaikan terbaru biasanya dibuat di luar alur kerja DevOps rilis biasa.

#### periode hypercare

Segera setelah cutover, periode waktu ketika tim migrasi mengelola dan memantau aplikasi yang dimigrasi di cloud untuk mengatasi masalah apa pun. Biasanya, periode ini panjangnya 1-4 hari. Pada akhir periode hypercare, tim migrasi biasanya mentransfer tanggung jawab untuk aplikasi ke tim operasi cloud.

I

#### IAC

Lihat [infrastruktur sebagai kode](#).

I

## kebijakan berbasis identitas

Kebijakan yang dilampirkan pada satu atau beberapa prinsip IAM yang mendefinisikan izin mereka dalam lingkungan. AWS Cloud

## aplikasi idle

Aplikasi yang memiliki penggunaan CPU dan memori rata-rata antara 5 dan 20 persen selama periode 90 hari. Dalam proyek migrasi, adalah umum untuk menghentikan aplikasi ini atau mempertahankannya di tempat.

## IIoT

Lihat [Internet of Things industri](#).

## infrastruktur yang tidak dapat diubah

Model yang menyebarkan infrastruktur baru untuk beban kerja produksi alih-alih memperbarui, menambal, atau memodifikasi infrastruktur yang ada. [Infrastruktur yang tidak dapat diubah secara inheren lebih konsisten, andal, dan dapat diprediksi daripada infrastruktur yang dapat berubah](#). Untuk informasi selengkapnya, lihat praktik terbaik [Deploy using immutable infrastructure](#) in the Framework. AWS Well-Architected

## masuk (masuknya) VPC

Dalam arsitektur AWS multi-akun, VPC yang menerima, memeriksa, dan merutekan koneksi jaringan dari luar aplikasi. [Arsitektur Referensi AWS Keamanan](#) merekomendasikan pengaturan akun Jaringan Anda dengan VPC masuk, keluar, dan inspeksi untuk melindungi antarmuka dua arah antara aplikasi Anda dan internet yang lebih luas.

## migrasi inkremental

Strategi cutover di mana Anda memigrasikan aplikasi Anda dalam bagian-bagian kecil alih-alih melakukan satu cutover penuh. Misalnya, Anda mungkin hanya memindahkan beberapa layanan mikro atau pengguna ke sistem baru pada awalnya. Setelah Anda memverifikasi bahwa semuanya berfungsi dengan baik, Anda dapat secara bertahap memindahkan layanan mikro atau pengguna tambahan hingga Anda dapat menonaktifkan sistem lama Anda. Strategi ini mengurangi risiko yang terkait dengan migrasi besar.

## Industri 4.0

Sebuah istilah yang diperkenalkan oleh [Klaus Schwab](#) pada tahun 2016 untuk merujuk pada modernisasi proses manufaktur melalui kemajuan dalam konektivitas, data real-time, otomatisasi, analitik, dan. AI/ML

## infrastruktur

Semua sumber daya dan aset yang terkandung dalam lingkungan aplikasi.

### infrastruktur sebagai kode (IAC)

Proses penyediaan dan pengelolaan infrastruktur aplikasi melalui satu set file konfigurasi. IAC dirancang untuk membantu Anda memusatkan manajemen infrastruktur, menstandarisasi sumber daya, dan menskalakan dengan cepat sehingga lingkungan baru dapat diulang, andal, dan konsisten.

### Internet of Things industri (IIoT)

Penggunaan sensor dan perangkat yang terhubung ke internet di sektor industri, seperti manufaktur, energi, otomotif, perawatan kesehatan, ilmu kehidupan, dan pertanian. Untuk informasi selengkapnya, lihat [Membangun strategi transformasi digital Internet of Things \(IIoT\) industri](#).

### inspeksi VPC

Dalam arsitektur AWS multi-akun, VPC terpusat yang mengelola inspeksi lalu lintas jaringan antara VPC (dalam hal yang sama atau berbeda Wilayah AWS), internet, dan jaringan lokal. [Arsitektur Referensi AWS Keamanan](#) merekomendasikan pengaturan akun Jaringan Anda dengan VPC masuk, keluar, dan inspeksi untuk melindungi antarmuka dua arah antara aplikasi Anda dan internet yang lebih luas.

### Internet of Things (IoT)

Jaringan objek fisik yang terhubung dengan sensor atau prosesor tertanam yang berkomunikasi dengan perangkat dan sistem lain melalui internet atau melalui jaringan komunikasi lokal. Untuk informasi selengkapnya, lihat [Apa itu IoT?](#)

### interpretabilitas

Karakteristik model pembelajaran mesin yang menggambarkan sejauh mana manusia dapat memahami bagaimana prediksi model bergantung pada inputnya. Untuk informasi lebih lanjut, lihat [Interpretabilitas model pembelajaran mesin](#) dengan AWS

### IoT

Lihat [Internet of Things](#).

## Perpustakaan informasi TI (ITIL)

Serangkaian praktik terbaik untuk memberikan layanan TI dan menyelaraskan layanan ini dengan persyaratan bisnis. ITIL menyediakan dasar untuk ITSM.

## Manajemen layanan TI (ITSM)

Kegiatan yang terkait dengan merancang, menerapkan, mengelola, dan mendukung layanan TI untuk suatu organisasi. Untuk informasi tentang mengintegrasikan operasi cloud dengan alat ITSM, lihat panduan [integrasi operasi](#).

## ITIL

Lihat [perpustakaan informasi TI](#).

## ITSM

Lihat [manajemen layanan TI](#).

## L

### kontrol akses berbasis label (LBAC)

Implementasi kontrol akses wajib (MAC) di mana pengguna dan data itu sendiri masing-masing secara eksplisit diberi nilai label keamanan. Persimpangan antara label keamanan pengguna dan label keamanan data menentukan baris dan kolom mana yang dapat dilihat oleh pengguna.

### landing zone

Landing zone adalah AWS lingkungan multi-akun yang dirancang dengan baik yang dapat diskalakan dan aman. Ini adalah titik awal dari mana organisasi Anda dapat dengan cepat meluncurkan dan menyebarkan beban kerja dan aplikasi dengan percaya diri dalam lingkungan keamanan dan infrastruktur mereka. Untuk informasi selengkapnya tentang zona pendaratan, lihat [Menyiapkan lingkungan multi-akun AWS yang aman dan dapat diskalakan](#).

### model bahasa besar (LLM)

Model [AI](#) pembelajaran mendalam yang dilatih sebelumnya pada sejumlah besar data. LLM dapat melakukan beberapa tugas, seperti menjawab pertanyaan, meringkas dokumen, menerjemahkan teks ke bahasa lain, dan menyelesaikan kalimat. Untuk informasi lebih lanjut, lihat [Apa itu LLM](#).

### migrasi besar

Migrasi 300 atau lebih server.

## LBAC

Lihat [kontrol akses berbasis label](#).

hak istimewa paling sedikit

Praktik keamanan terbaik untuk memberikan izin minimum yang diperlukan untuk melakukan tugas. Untuk informasi selengkapnya, lihat [Menerapkan izin hak istimewa terkecil dalam dokumentasi IAM](#).

angkat dan geser

Lihat [7 Rs](#).

sistem endian kecil

Sebuah sistem yang menyimpan byte paling tidak signifikan terlebih dahulu. Lihat juga [endianness](#).

## LLM

Lihat [model bahasa besar](#).

lingkungan yang lebih rendah

Lihat [lingkungan](#).

## M

pembelajaran mesin (ML)

Jenis kecerdasan buatan yang menggunakan algoritma dan teknik untuk pengenalan pola dan pembelajaran. ML menganalisis dan belajar dari data yang direkam, seperti data Internet of Things (IoT), untuk menghasilkan model statistik berdasarkan pola. Untuk informasi selengkapnya, lihat [Machine Learning](#).

cabang utama

Lihat [cabang](#).

malware

Perangkat lunak yang dirancang untuk membahayakan keamanan atau privasi komputer. Malware dapat mengganggu sistem komputer, membocorkan informasi sensitif, atau

mendapatkan akses yang tidak sah. Contoh malware termasuk virus, worm, ransomware, Trojan horse, spyware, dan keyloggers.

## layanan terkelola

Layanan AWS yang AWS mengoperasikan lapisan infrastruktur, sistem operasi, dan platform, dan Anda mengakses titik akhir untuk menyimpan dan mengambil data. Amazon Simple Storage Service (Amazon S3) dan Amazon DynamoDB adalah contoh layanan terkelola. Ini juga dikenal sebagai layanan abstrak.

## sistem eksekusi manufaktur (MES)

Sistem perangkat lunak untuk melacak, memantau, mendokumentasikan, dan mengendalikan proses produksi yang mengubah bahan baku menjadi produk jadi di lantai toko.

## PETA

Lihat [Program Percepatan Migrasi](#).

## MCP

Lihat [Protokol Konteks Model](#).

## Protokol Konteks Model (MCP)

Protokol stateless untuk komunikasi [agen](#) -to- [alat](#).

## Server MCP

Layanan yang mengekspos satu atau lebih [alat](#) melalui [Protokol Konteks Model](#).

## mekanisme

Proses lengkap di mana Anda membuat alat, mendorong adopsi alat, dan kemudian memeriksa hasilnya untuk melakukan penyesuaian. Mekanisme adalah siklus yang memperkuat dan meningkatkan dirinya sendiri saat beroperasi. Untuk informasi selengkapnya, lihat [Membangun mekanisme](#) dalam AWS Well-Architected Kerangka Kerja.

## akun anggota

Semua Akun AWS selain akun manajemen yang merupakan bagian dari organisasi di AWS Organizations. Akun dapat menjadi anggota dari hanya satu organisasi pada suatu waktu.

## MES

Lihat [sistem eksekusi manufaktur](#).

## Transportasi Telemetri Antrian Pesan (MQTT)

[Protokol komunikasi mesin-ke-mesin \(M2M\) yang ringan, berdasarkan pola publish/subscribe, untuk perangkat IoT yang dibatasi sumber daya.](#)

### layanan mikro

Layanan kecil dan independen yang berkomunikasi melalui API yang terdefinisi dengan baik dan biasanya dimiliki oleh tim kecil yang mandiri. Misalnya, sistem asuransi mungkin mencakup layanan mikro yang memetakan kemampuan bisnis, seperti penjualan atau pemasaran, atau subdomain, seperti pembelian, klaim, atau analitik. Manfaat layanan mikro termasuk kelincahan, penskalaan yang fleksibel, penyebaran yang mudah, kode yang dapat digunakan kembali, dan ketahanan. Untuk informasi selengkapnya, lihat [Mengintegrasikan layanan mikro dengan menggunakan layanan tanpa AWS server.](#)

### arsitektur microservices

Pendekatan untuk membangun aplikasi dengan komponen independen yang menjalankan setiap proses aplikasi sebagai layanan mikro. Layanan mikro ini berkomunikasi melalui antarmuka yang terdefinisi dengan baik dengan menggunakan API ringan. Setiap layanan mikro dalam arsitektur ini dapat diperbarui, digunakan, dan diskalakan untuk memenuhi permintaan fungsi tertentu dari suatu aplikasi. Untuk informasi selengkapnya, lihat [Menerapkan layanan mikro di AWS.](#)

### Program Percepatan Migrasi (MAP)

AWS Program yang menyediakan dukungan konsultasi, pelatihan, dan layanan untuk membantu organisasi membangun fondasi operasional yang kuat untuk pindah ke cloud, dan untuk membantu mengimbangi biaya awal migrasi. MAP mencakup metodologi migrasi untuk mengeksekusi migrasi lama dengan cara metodis dan seperangkat alat untuk mengotomatisasi dan mempercepat skenario migrasi umum.

### migrasi dalam skala

Proses memindahkan sebagian besar portofolio aplikasi ke cloud dalam gelombang, dengan lebih banyak aplikasi bergerak pada tingkat yang lebih cepat di setiap gelombang. Fase ini menggunakan praktik terbaik dan pelajaran yang dipetik dari fase sebelumnya untuk mengimplementasikan pabrik migrasi tim, alat, dan proses untuk merampingkan migrasi beban kerja melalui otomatisasi dan pengiriman tangkas. Ini adalah fase ketiga dari [strategi AWS migrasi.](#)

## pabrik migrasi

Cross-functional tim yang merampingkan migrasi beban kerja melalui pendekatan otomatis dan gesit. Tim pabrik migrasi biasanya mencakup operasi, analis dan pemilik bisnis, insinyur migrasi, pengembang, dan DevOps profesional yang bekerja di sprint. Antara 20 dan 50 persen portofolio aplikasi perusahaan terdiri dari pola berulang yang dapat dioptimalkan dengan pendekatan pabrik. Untuk informasi selengkapnya, lihat [diskusi tentang pabrik migrasi](#) dan [panduan Pabrik Migrasi Cloud](#) di kumpulan konten ini.

## metadata migrasi

Informasi tentang aplikasi dan server yang diperlukan untuk menyelesaikan migrasi. Setiap pola migrasi memerlukan satu set metadata migrasi yang berbeda. Contoh metadata migrasi termasuk subnet target, grup keamanan, dan akun. AWS

## pola migrasi

Tugas migrasi berulang yang merinci strategi migrasi, tujuan migrasi, dan aplikasi atau layanan migrasi yang digunakan. Contoh: Rehost migrasi ke Amazon EC2 AWS dengan Layanan Migrasi Aplikasi.

## Penilaian Portofolio Migrasi (MPA)

Alat online yang menyediakan informasi untuk memvalidasi kasus bisnis untuk bermigrasi ke. AWS Cloud MPA menyediakan penilaian portofolio terperinci (ukuran kanan server, harga, perbandingan TCO, analisis biaya migrasi) serta perencanaan migrasi (analisis data aplikasi dan pengumpulan data, pengelompokan aplikasi, prioritas migrasi, dan perencanaan gelombang). [Alat MPA](#) (memerlukan login) tersedia gratis untuk semua AWS konsultan dan konsultan APN Partner.

## Penilaian Kesiapan Migrasi (MRA)

Proses mendapatkan wawasan tentang status kesiapan cloud organisasi, mengidentifikasi kekuatan dan kelemahan, dan membangun rencana aksi untuk menutup kesenjangan yang diidentifikasi, menggunakan CAF. AWS Untuk informasi selengkapnya, lihat [panduan kesiapan migrasi](#). MRA adalah tahap pertama dari [strategi AWS migrasi](#).

## strategi migrasi

Pendekatan yang digunakan untuk memigrasikan beban kerja ke. AWS Cloud Untuk informasi lebih lanjut, lihat entri [7 Rs](#) di glosarium ini dan lihat [Memobilisasi organisasi Anda untuk mempercepat](#) migrasi skala besar.

## ML

Lihat [pembelajaran mesin](#).

### modernisasi

Mengubah aplikasi usang (warisan atau monolitik) dan infrastrukturnya menjadi sistem yang gesit, elastis, dan sangat tersedia di cloud untuk mengurangi biaya, mendapatkan efisiensi, dan memanfaatkan inovasi. Untuk informasi selengkapnya, lihat [Strategi untuk memodernisasi aplikasi di AWS Cloud](#).

### penilaian kesiapan modernisasi

Evaluasi yang membantu menentukan kesiapan modernisasi aplikasi organisasi; mengidentifikasi manfaat, risiko, dan dependensi; dan menentukan seberapa baik organisasi dapat mendukung keadaan masa depan aplikasi tersebut. Hasil penilaian adalah cetak biru arsitektur target, peta jalan yang merinci fase pengembangan dan tonggak untuk proses modernisasi, dan rencana aksi untuk mengatasi kesenjangan yang diidentifikasi. Untuk informasi lebih lanjut, lihat [Mengevaluasi kesiapan modernisasi untuk aplikasi di AWS Cloud](#).

### aplikasi monolitik (monolit)

Aplikasi yang berjalan sebagai layanan tunggal dengan proses yang digabungkan secara ketat. Aplikasi monolitik memiliki beberapa kelemahan. Jika satu fitur aplikasi mengalami lonjakan permintaan, seluruh arsitektur harus diskalakan. Menambahkan atau meningkatkan fitur aplikasi monolitik juga menjadi lebih kompleks ketika basis kode tumbuh. Untuk mengatasi masalah ini, Anda dapat menggunakan arsitektur microservices. Untuk informasi lebih lanjut, lihat [Mengurai monolit](#) menjadi layanan mikro.

## MPA

Lihat [Penilaian Portofolio Migrasi](#).

## MQTT

Lihat [Transportasi Telemetri Antrian Pesan](#).

### klasifikasi multiclass

Sebuah proses yang membantu menghasilkan prediksi untuk beberapa kelas (memprediksi satu dari lebih dari dua hasil). Misalnya, model ML mungkin bertanya “Apakah produk ini buku, mobil, atau telepon?” atau “Kategori produk mana yang paling menarik bagi pelanggan ini?”

## infrastruktur yang bisa berubah

Model yang memperbarui dan memodifikasi infrastruktur yang ada untuk beban kerja produksi. Untuk meningkatkan konsistensi, keandalan, dan prediktabilitas, AWS Well-Architected Framework merekomendasikan penggunaan [infrastruktur yang tidak dapat diubah](#) sebagai praktik terbaik.

## O

### OAC

Lihat [kontrol akses asal](#).

### OAI

Lihat [identitas akses asal](#).

### OCM

Lihat [manajemen perubahan organisasi](#).

## migrasi offline

Metode migrasi di mana beban kerja sumber diturunkan selama proses migrasi. Metode ini melibatkan waktu henti yang diperpanjang dan biasanya digunakan untuk beban kerja kecil dan tidak kritis.

## OI

Lihat [integrasi operasi](#).

## OLA

Lihat [perjanjian tingkat operasional](#).

## migrasi online

Metode migrasi di mana beban kerja sumber disalin ke sistem target tanpa diambil offline. Aplikasi yang terhubung ke beban kerja dapat terus berfungsi selama migrasi. Metode ini melibatkan waktu henti nol hingga minimal dan biasanya digunakan untuk beban kerja produksi yang kritis.

## OPC-UA

Lihat [Komunikasi Proses Terbuka - Arsitektur Terpadu](#).

## Komunikasi Proses Terbuka - Arsitektur Terpadu () OPC-UA

Protokol komunikasi mesin-ke-mesin (M2M) untuk otomasi industri. OPC-UA menyediakan standar interoperabilitas dengan enkripsi data, otentikasi, dan skema otorisasi.

### perjanjian tingkat operasional (OLA)

Perjanjian yang menjelaskan apa yang dijanjikan kelompok TI fungsional untuk diberikan satu sama lain, untuk mendukung perjanjian tingkat layanan (SLA).

### Tinjauan Kesiapan Operasional (ORR)

Daftar pertanyaan dan praktik terbaik terkait yang membantu Anda memahami, mengevaluasi, mencegah, atau mengurangi ruang lingkup insiden dan kemungkinan kegagalan. Untuk informasi selengkapnya, lihat [Ulasan Kesiapan Operasional \(ORR\) dalam Kerangka Kerja AWS Well-Architected](#)

### teknologi operasional (OT)

Sistem perangkat keras dan perangkat lunak yang bekerja dengan lingkungan fisik untuk mengendalikan operasi industri, peralatan, dan infrastruktur. Di bidang manufaktur, integrasi sistem OT dan teknologi informasi (TI) adalah fokus utama untuk transformasi [Industri 4.0](#).

### integrasi operasi (OI)

Proses modernisasi operasi di cloud, yang melibatkan perencanaan kesiapan, otomatisasi, dan integrasi. Untuk informasi selengkapnya, lihat [panduan integrasi operasi](#).

### jejak organisasi

Jejak yang dibuat oleh AWS CloudTrail itu mencatat semua peristiwa untuk semua Akun AWS dalam organisasi di AWS Organizations. Jejak ini dibuat di setiap Akun AWS bagian organisasi dan melacak aktivitas di setiap akun. Untuk informasi selengkapnya, lihat [Membuat jejak untuk organisasi](#) dalam CloudTrail dokumentasi.

### manajemen perubahan organisasi (OCM)

Kerangka kerja untuk mengelola transformasi bisnis utama yang mengganggu dari perspektif orang, budaya, dan kepemimpinan. OCM membantu organisasi mempersiapkan, dan transisi ke, sistem dan strategi baru dengan mempercepat adopsi perubahan, mengatasi masalah transisi, dan mendorong perubahan budaya dan organisasi. Dalam strategi AWS migrasi, kerangka kerja ini disebut percepatan orang, karena kecepatan perubahan yang diperlukan dalam proyek adopsi cloud. Untuk informasi lebih lanjut, lihat [panduan OCM](#).

## kontrol akses asal (OAC)

Di CloudFront, opsi yang disempurnakan untuk membatasi akses untuk mengamankan konten Amazon Simple Storage Service (Amazon S3) Anda. OAC mendukung semua bucket S3 di semua Wilayah AWS, enkripsi sisi server dengan AWS KMS (SSE-KMS), dan dinamis PUT dan DELETE permintaan ke bucket S3.

## identitas akses asal (OAI)

Di CloudFront, opsi untuk membatasi akses untuk mengamankan konten Amazon S3 Anda. Saat Anda menggunakan OAI, CloudFront buat prinsipal yang dapat diautentikasi oleh Amazon S3. Prinsipal yang diautentikasi dapat mengakses konten dalam bucket S3 hanya melalui distribusi tertentu. CloudFront Lihat juga [OAC](#), yang menyediakan kontrol akses yang lebih terperinci dan ditingkatkan.

## ORR

Lihat [tinjauan kesiapan operasional](#).

## OT

Lihat [teknologi operasional](#).

## keluar (jalan keluar) VPC

Dalam arsitektur AWS multi-akun, VPC yang menangani koneksi jaringan yang dimulai dari dalam aplikasi. [Arsitektur Referensi AWS Keamanan](#) merekomendasikan pengaturan akun Jaringan Anda dengan VPC masuk, keluar, dan inspeksi untuk melindungi antarmuka dua arah antara aplikasi Anda dan internet yang lebih luas.

## P

### batas izin

Kebijakan manajemen IAM yang dilampirkan pada prinsipal IAM untuk menetapkan izin maksimum yang dapat dimiliki pengguna atau peran. Untuk informasi selengkapnya, lihat [Batas izin](#) dalam dokumentasi IAM.

### Informasi Identifikasi Pribadi (PII)

Informasi yang, jika dilihat secara langsung atau dipasangkan dengan data terkait lainnya, dapat digunakan untuk menyimpulkan identitas individu secara wajar. Contoh PII termasuk nama, alamat, dan informasi kontak.

## PII

Lihat informasi yang [dapat diidentifikasi secara pribadi](#).

### buku pedoman

Serangkaian langkah yang telah ditentukan sebelumnya yang menangkap pekerjaan yang terkait dengan migrasi, seperti mengirimkan fungsi operasi inti di cloud. Buku pedoman dapat berupa skrip, runbook otomatis, atau ringkasan proses atau langkah-langkah yang diperlukan untuk mengoperasikan lingkungan modern Anda.

## PLC

Lihat [pengontrol logika yang dapat diprogram](#).

## PLM

Lihat [manajemen siklus hidup produk](#).

### kebijakan

[Objek yang dapat menentukan izin \(lihat kebijakan berbasis identitas\), menentukan kondisi akses \(lihat kebijakan berbasis sumber daya\), atau menentukan izin maksimum untuk semua akun dalam organisasi di \(lihat kebijakan kontrol layanan\). AWS Organizations](#)

### persistensi poliglot

Secara independen memilih teknologi penyimpanan data microservice berdasarkan pola akses data dan persyaratan lainnya. Jika layanan mikro Anda memiliki teknologi penyimpanan data yang sama, mereka dapat menghadapi tantangan implementasi atau mengalami kinerja yang buruk. Layanan mikro lebih mudah diimplementasikan dan mencapai kinerja dan skalabilitas yang lebih baik jika mereka menggunakan penyimpanan data yang paling sesuai dengan kebutuhan mereka.

### penilaian portofolio

Proses menemukan, menganalisis, dan memprioritaskan portofolio aplikasi untuk merencanakan migrasi. Untuk informasi selengkapnya, lihat [Mengevaluasi kesiapan migrasi](#).

### predikat

Kondisi kueri yang mengembalikan `true` atau `false`, biasanya terletak di `WHERE` klausa.

### predikat pushdown

Teknik pengoptimalan kueri database yang menyaring data dalam kueri sebelum transfer. Ini mengurangi jumlah data yang harus diambil dan diproses dari database relasional, dan meningkatkan kinerja kueri.

## kontrol preventif

Kontrol keamanan yang dirancang untuk mencegah suatu peristiwa terjadi. Kontrol ini adalah garis pertahanan pertama untuk membantu mencegah akses tidak sah atau perubahan yang tidak diinginkan ke jaringan Anda. Untuk informasi selengkapnya, lihat [Kontrol pencegahan dalam Menerapkan kontrol](#) keamanan pada. AWS

## principal

Entitas AWS yang dapat melakukan tindakan dan mengakses sumber daya. Entitas ini biasanya merupakan pengguna root untuk Akun AWS, peran IAM, atau pengguna. Untuk informasi selengkapnya, lihat Prinsip dalam [istilah dan konsep Peran](#) dalam dokumentasi IAM.

## privasi berdasarkan desain

Pendekatan rekayasa sistem yang memperhitungkan privasi melalui seluruh proses pengembangan.

## zona host pribadi

Container yang menyimpan informasi tentang bagaimana Anda ingin Amazon Route 53 merespons kueri DNS untuk domain dan subdomainnya dalam satu atau beberapa VPC. Untuk informasi selengkapnya, lihat [Bekerja dengan zona yang dihosting pribadi](#) di dokumentasi Route 53.

## kontrol proaktif

[Kontrol keamanan](#) yang dirancang untuk mencegah penyebaran sumber daya yang tidak sesuai. Kontrol ini memindai sumber daya sebelum disediakan. Jika sumber daya tidak sesuai dengan kontrol, maka itu tidak disediakan. Untuk informasi selengkapnya, lihat [panduan referensi Kontrol](#) dalam AWS Control Tower dokumentasi dan lihat [Kontrol proaktif](#) dalam Menerapkan kontrol keamanan pada AWS.

## manajemen siklus hidup produk (PLM)

Manajemen data dan proses untuk suatu produk di seluruh siklus hidupnya, mulai dari desain, pengembangan, dan peluncuran, melalui pertumbuhan dan kematangan, hingga penurunan dan penghapusan.

## lingkungan produksi

Lihat [lingkungan](#).

## pengontrol logika yang dapat diprogram (PLC)

Di bidang manufaktur, komputer yang sangat andal dan mudah beradaptasi yang memantau mesin dan mengotomatiskan proses manufaktur.

## rantai cepat

Menggunakan output dari satu prompt [LLM](#) sebagai input untuk prompt berikutnya untuk menghasilkan respons yang lebih baik. Teknik ini digunakan untuk memecah tugas yang kompleks menjadi subtugas, atau untuk secara iteratif memperbaiki atau memperluas respons awal. Ini membantu meningkatkan akurasi dan relevansi respons model dan memungkinkan hasil yang lebih terperinci dan dipersonalisasi.

## pseudonimisasi

Proses penggantian pengenal pribadi dalam kumpulan data dengan nilai placeholder. Pseudonimisasi dapat membantu melindungi privasi pribadi. Data pseudonim masih dianggap sebagai data pribadi.

## publish/subscribe (pub/sub)

Pola yang memungkinkan komunikasi asinkron antara layanan mikro untuk meningkatkan skalabilitas dan daya tanggap. Misalnya, dalam [MES](#) berbasis layanan mikro, layanan mikro dapat mempublikasikan pesan peristiwa ke saluran yang dapat berlangganan layanan mikro lainnya. Sistem dapat menambahkan layanan mikro baru tanpa mengubah layanan penerbitan.

# Q

## rencana kueri

Serangkaian langkah, seperti instruksi, yang digunakan untuk mengakses data dalam sistem database relasional SQL.

## regresi rencana kueri

Ketika pengoptimal layanan database memilih rencana yang kurang optimal daripada sebelum perubahan yang diberikan ke lingkungan database. Hal ini dapat disebabkan oleh perubahan statistik, kendala, pengaturan lingkungan, pengikatan parameter kueri, dan pembaruan ke mesin database.

## R

### Matriks RACI

Lihat [bertanggung jawab, akuntabel, dikonsultasikan, diinformasikan \(RACI\)](#).

### LAP

Lihat [Retrieval Augmented Generation](#).

### ransomware

Perangkat lunak berbahaya yang dirancang untuk memblokir akses ke sistem komputer atau data sampai pembayaran dilakukan.

### Matriks RASCI

Lihat [bertanggung jawab, akuntabel, dikonsultasikan, diinformasikan \(RACI\)](#).

### RCAC

Lihat [kontrol akses baris dan kolom](#).

### replika baca

Salinan database yang digunakan untuk tujuan read-only. Anda dapat merutekan kueri ke replika baca untuk mengurangi beban pada database utama Anda.

### arsitek ulang

Lihat [7 Rs](#).

### tujuan titik pemulihan (RPO)

Jumlah waktu maksimum yang dapat diterima sejak titik pemulihan data terakhir. Ini menentukan apa yang dianggap sebagai kehilangan data yang dapat diterima antara titik pemulihan terakhir dan gangguan layanan.

### tujuan waktu pemulihan (RTO)

Penundaan maksimum yang dapat diterima antara gangguan layanan dan pemulihan layanan.

### refactor

Lihat [7 Rs](#).

## Region

Kumpulan AWS sumber daya di wilayah geografis. Masing-masing Wilayah AWS terisolasi dan independen dari yang lain untuk memberikan toleransi kesalahan, stabilitas, dan ketahanan.

Untuk informasi selengkapnya, lihat [Menentukan Wilayah AWS akun yang dapat digunakan](#).

## regresi

Teknik ML yang memprediksi nilai numerik. Misalnya, untuk memecahkan masalah “Berapa harga rumah ini akan dijual?” Model ML dapat menggunakan model regresi linier untuk memprediksi harga jual rumah berdasarkan fakta yang diketahui tentang rumah (misalnya, luas persegi).

## rehost

Lihat [7 Rs](#).

## melepaskan

Dalam proses penyebaran, tindakan mempromosikan perubahan pada lingkungan produksi.

## memindahkan

Lihat [7 Rs](#).

## memplatform ulang

Lihat [7 Rs](#).

## pembelian kembali

Lihat [7 Rs](#).

## ketahanan

Kemampuan aplikasi untuk melawan atau pulih dari gangguan. [Ketersediaan tinggi](#) dan [pemulihan bencana](#) adalah pertimbangan umum ketika merencanakan ketahanan di AWS Cloud

Untuk informasi lebih lanjut, lihat [AWS Cloud Ketahanan](#).

## kebijakan berbasis sumber daya

Kebijakan yang dilampirkan ke sumber daya, seperti bucket Amazon S3, titik akhir, atau kunci enkripsi. Jenis kebijakan ini menentukan prinsipal mana yang diizinkan mengakses, tindakan yang didukung, dan kondisi lain yang harus dipenuhi.

## matriks yang bertanggung jawab, akuntabel, dikonsultasikan, diinformasikan (RACI)

Matriks yang mendefinisikan peran dan tanggung jawab untuk semua pihak yang terlibat dalam kegiatan migrasi dan operasi cloud. Nama matriks berasal dari jenis tanggung jawab yang

didefinisikan dalam matriks: bertanggung jawab (R), akuntabel (A), dikonsultasikan (C), dan diinformasikan (I). Jenis dukungan (S) adalah opsional. Jika Anda menyertakan dukungan, matriks disebut matriks RASCI, dan jika Anda mengecualikannya, itu disebut matriks RACI.

## kontrol responsif

Kontrol keamanan yang dirancang untuk mendorong remediasi efek samping atau penyimpangan dari garis dasar keamanan Anda. Untuk informasi selengkapnya, lihat [Kontrol responsif](#) dalam Menerapkan kontrol keamanan pada AWS.

## melestarikan

Lihat [7 Rs](#).

## pensiun

Lihat [7 Rs](#).

## Retrieval Augmented Generation (RAG)

Teknologi [AI generatif](#) di mana [LLM](#) mereferensikan sumber data otoritatif yang berada di luar sumber data pelatihannya sebelum menghasilkan respons. Misalnya, model RAG mungkin melakukan pencarian semantik dari basis pengetahuan organisasi atau data kustom. Untuk informasi lebih lanjut, lihat [Apa itu RAG](#).

## rotasi

Proses memperbarui [rahasia](#) secara berkala untuk membuatnya lebih sulit bagi penyerang untuk mengakses kredensial.

## kontrol akses baris dan kolom (RCAC)

Penggunaan ekspresi SQL dasar dan fleksibel yang telah menetapkan aturan akses. RCAC terdiri dari izin baris dan topeng kolom.

## RPO

Lihat [tujuan titik pemulihan](#).

## RTO

Lihat [tujuan waktu pemulihan](#).

## buku runbook

Satu set prosedur manual atau otomatis yang diperlukan untuk melakukan tugas tertentu. Ini biasanya dibangun untuk merampingkan operasi berulang atau prosedur dengan tingkat kesalahan yang tinggi.

## D

### SAML 2.0

Standar terbuka yang digunakan oleh banyak penyedia identitas (IdPs). Fitur ini memungkinkan sistem masuk tunggal gabungan (SSO), sehingga pengguna dapat masuk ke Konsol Manajemen AWS atau memanggil operasi AWS API tanpa Anda harus membuat pengguna di IAM untuk semua orang di organisasi Anda. Untuk informasi lebih lanjut tentang federasi berbasis SAMP 2.0, lihat [Tentang federasi berbasis SAMP 2.0](#) dalam dokumentasi IAM.

### SCADA

Lihat [kontrol pengawasan dan akuisisi data](#).

### SCP

Lihat [kebijakan kontrol layanan](#).

### Rahasia

Dalam AWS Secrets Manager, informasi rahasia atau terbatas, seperti kata sandi atau kredensial pengguna, yang Anda simpan dalam bentuk terenkripsi. Ini terdiri dari nilai rahasia dan metadatanya. Nilai rahasia dapat berupa biner, string tunggal, atau beberapa string. Untuk informasi selengkapnya, lihat [Apa yang ada di rahasia Secrets Manager?](#) dalam dokumentasi Secrets Manager.

### keamanan dengan desain

Pendekatan rekayasa sistem yang memperhitungkan keamanan melalui seluruh proses pengembangan.

### kontrol keamanan

Pagar pembatas teknis atau administratif yang mencegah, mendeteksi, atau mengurangi kemampuan pelaku ancaman untuk mengeksploitasi kerentanan keamanan. [Ada empat jenis kontrol keamanan utama: preventif, detektif, responsif, dan proaktif.](#)

## pengerasan keamanan

Proses mengurangi permukaan serangan untuk membuatnya lebih tahan terhadap serangan. Ini dapat mencakup tindakan seperti menghapus sumber daya yang tidak lagi diperlukan, menerapkan praktik keamanan terbaik untuk memberikan hak istimewa paling sedikit, atau menonaktifkan fitur yang tidak perlu dalam file konfigurasi.

## sistem informasi keamanan dan manajemen acara (SIEM)

Alat dan layanan yang menggabungkan sistem manajemen informasi keamanan (SIM) dan manajemen acara keamanan (SEM). Sistem SIEM mengumpulkan, memantau, dan menganalisis data dari server, jaringan, perangkat, dan sumber lain untuk mendeteksi ancaman dan pelanggaran keamanan, dan untuk menghasilkan peringatan.

## otomatisasi respons keamanan

Tindakan yang telah ditentukan dan diprogram yang dirancang untuk secara otomatis merespons atau memulihkan peristiwa keamanan. Otomatisasi ini berfungsi sebagai kontrol keamanan [detektif](#) atau [responsif](#) yang membantu Anda menerapkan praktik terbaik AWS keamanan. Contoh tindakan respons otomatis termasuk memodifikasi grup keamanan VPC, menambal instans Amazon EC2, atau memutar kredensial.

## enkripsi sisi server

Enkripsi data di tujuannya, oleh Layanan AWS yang menerimanya.

## kebijakan kontrol layanan (SCP)

Kebijakan yang menyediakan kontrol terpusat atas izin untuk semua akun di organisasi. AWS Organizations SCP menentukan pagar pembatas atau menetapkan batasan pada tindakan yang dapat didelegasikan oleh administrator kepada pengguna atau peran. Anda dapat menggunakan SCP sebagai daftar izin atau daftar penolakan, untuk menentukan layanan atau tindakan mana yang diizinkan atau dilarang. Untuk informasi selengkapnya, lihat [Kebijakan kontrol layanan](#) dalam AWS Organizations dokumentasi.

## titik akhir layanan

URL titik masuk untuk file Layanan AWS. Anda dapat menggunakan endpoint untuk terhubung secara terprogram ke layanan target. Untuk informasi selengkapnya, lihat [Layanan AWS titik akhir](#) di Referensi Umum AWS.

## perjanjian tingkat layanan (SLA)

Perjanjian yang menjelaskan apa yang dijanjikan tim TI untuk diberikan kepada pelanggan mereka, seperti waktu kerja dan kinerja layanan.

## indikator tingkat layanan (SLI)

Pengukuran aspek kinerja layanan, seperti tingkat kesalahan, ketersediaan, atau throughputnya.

## tujuan tingkat layanan (SLO)

Metrik target yang mewakili kesehatan layanan, yang diukur dengan indikator [tingkat layanan](#).

## model tanggung jawab bersama

Model yang menjelaskan tanggung jawab yang Anda bagikan AWS untuk keamanan dan kepatuhan cloud. AWS bertanggung jawab atas keamanan cloud, sedangkan Anda bertanggung jawab atas keamanan di cloud. Untuk informasi selengkapnya, lihat [Model tanggung jawab bersama](#).

## Bayangan AI

Aplikasi [AI](#) yang tidak sah dibuat atau digunakan di luar saluran yang diatur dalam suatu organisasi.

## SIEM

Lihat [informasi keamanan dan sistem manajemen acara](#).

## titik kegagalan tunggal (SPOF)

Kegagalan dalam satu komponen penting dari aplikasi yang dapat mengganggu sistem.

## SLA

Lihat [perjanjian tingkat layanan](#).

## SLI

Lihat [indikator tingkat layanan](#).

## SLO

Lihat [tujuan tingkat layanan](#).

## model split-and-lead

Pola untuk menskalakan dan mempercepat proyek modernisasi. Ketika fitur baru dan rilis produk didefinisikan, tim inti berpisah untuk membuat tim produk baru. Ini membantu meningkatkan

kemampuan dan layanan organisasi Anda, meningkatkan produktivitas pengembang, dan mendukung inovasi yang cepat. Untuk informasi lebih lanjut, lihat [Pendekatan bertahap untuk memodernisasi aplikasi](#) di AWS Cloud

## SPOF

Lihat [satu titik kegagalan](#).

## skema bintang

Struktur organisasi database yang menggunakan satu tabel fakta besar untuk menyimpan data transaksional atau terukur dan menggunakan satu atau lebih tabel dimensi yang lebih kecil untuk menyimpan atribut data. Struktur ini dirancang untuk digunakan dalam [gudang data](#) atau untuk tujuan intelijen bisnis.

## pola ara pencekik

Pendekatan untuk memodernisasi sistem monolitik dengan menulis ulang secara bertahap dan mengganti fungsionalitas sistem sampai sistem warisan dapat dinonaktifkan. Pola ini menggunakan analogi pohon ara yang tumbuh menjadi pohon yang sudah mapan dan akhirnya mengatasi dan menggantikan inangnya. Pola ini [diperkenalkan oleh Martin Fowler](#) sebagai cara untuk mengelola risiko saat menulis ulang sistem monolitik. Untuk contoh cara menerapkan pola ini, lihat [Memodernisasi layanan web ASP.NET Microsoft \(ASMX\) lama secara bertahap menggunakan container dan Amazon API Gateway](#).

## subnet

Rentang alamat IP dalam VPC Anda. Subnet harus berada di Availability Zone tunggal.

## kontrol pengawasan dan akuisisi data (SCADA)

Di bidang manufaktur, sistem yang menggunakan perangkat keras dan perangkat lunak untuk memantau aset fisik dan operasi produksi.

## enkripsi simetris

Algoritma enkripsi yang menggunakan kunci yang sama untuk mengenkripsi dan mendekripsi data.

## pengujian sintetis

Menguji sistem dengan cara yang mensimulasikan interaksi pengguna untuk mendeteksi potensi masalah atau untuk memantau kinerja. Anda dapat menggunakan [Amazon CloudWatch Synthetics](#) untuk membuat tes ini.

## sistem prompt

Teknik untuk memberikan konteks, instruksi, atau pedoman ke [LLM](#) untuk mengarahkan perilakunya. Permintaan sistem membantu mengatur konteks dan menetapkan aturan untuk interaksi dengan pengguna.

## T

### tag

Key-value pasangan yang bertindak sebagai metadata untuk mengatur sumber daya AWS . Tag membantu Anda mengelola, mengidentifikasi, mengatur, dan memfilter sumber daya. Untuk informasi selengkapnya, lihat [Menandai sumber daya AWS](#).

### variabel target

Nilai yang Anda coba prediksi dalam ML yang diawasi. Ini juga disebut sebagai variabel hasil. Misalnya, dalam pengaturan manufaktur, variabel target bisa menjadi cacat produk.

### daftar tugas

Alat yang digunakan untuk melacak kemajuan melalui runbook. Daftar tugas berisi ikhtisar runbook dan daftar tugas umum yang harus diselesaikan. Untuk setiap tugas umum, itu termasuk perkiraan jumlah waktu yang dibutuhkan, pemilik, dan kemajuan.

### lingkungan uji

Lihat [lingkungan](#).

### pelatihan

Untuk menyediakan data bagi model ML Anda untuk dipelajari. Data pelatihan harus berisi jawaban yang benar. Algoritma pembelajaran menemukan pola dalam data pelatihan yang memetakan atribut data input ke target (jawaban yang ingin Anda prediksi). Ini menghasilkan model ML yang menangkap pola-pola ini. Anda kemudian dapat menggunakan model ML untuk membuat prediksi pada data baru yang Anda tidak tahu targetnya.

### alat

Fungsi atau API yang dapat [dipanggil agen](#) untuk melakukan operasi di sistem eksternal.

## gerbang transit

Hub transit jaringan yang dapat Anda gunakan untuk menghubungkan VPC dan jaringan lokal Anda. Untuk informasi selengkapnya, lihat [Apa itu gateway transit](#) dalam AWS Transit Gateway dokumentasi.

## alur kerja berbasis batang

Pendekatan di mana pengembang membangun dan menguji fitur secara lokal di cabang fitur dan kemudian menggabungkan perubahan tersebut ke cabang utama. Cabang utama kemudian dibangun untuk pengembangan, praproduksi, dan lingkungan produksi, secara berurutan.

## akses tepercaya

Memberikan izin ke layanan yang Anda tentukan untuk melakukan tugas di organisasi Anda di dalam AWS Organizations dan di akunnya atas nama Anda. Layanan tepercaya menciptakan peran terkait layanan di setiap akun, ketika peran itu diperlukan, untuk melakukan tugas manajemen untuk Anda. Untuk informasi selengkapnya, lihat [Menggunakan AWS Organizations dengan AWS layanan lain](#) dalam AWS Organizations dokumentasi.

## penyetelan

Untuk mengubah aspek proses pelatihan Anda untuk meningkatkan akurasi model ML. Misalnya, Anda dapat melatih model ML dengan membuat set pelabelan, menambahkan label, dan kemudian mengulangi langkah-langkah ini beberapa kali di bawah pengaturan yang berbeda untuk mengoptimalkan model.

## tim dua pizza

Sebuah DevOps tim kecil yang bisa Anda beri makan dengan dua pizza. Ukuran tim dua pizza memastikan peluang terbaik untuk berkolaborasi dalam pengembangan perangkat lunak.

# U

## waswas

Sebuah konsep yang mengacu pada informasi yang tidak tepat, tidak lengkap, atau tidak diketahui yang dapat merusak keandalan model ML prediktif. Ada dua jenis ketidakpastian: ketidakpastian epistemik disebabkan oleh data yang terbatas dan tidak lengkap, sedangkan ketidakpastian aleatorik disebabkan oleh kebisingan dan keacakan yang melekat dalam data.

## tugas yang tidak terdiferensiasi

Juga dikenal sebagai angkat berat, pekerjaan yang diperlukan untuk membuat dan mengoperasikan aplikasi tetapi itu tidak memberikan nilai langsung kepada pengguna akhir atau memberikan keunggulan kompetitif. Contoh tugas yang tidak terdiferensiasi termasuk pengadaan, pemeliharaan, dan perencanaan kapasitas.

## lingkungan atas

Lihat [lingkungan](#).

## V

### menyedot debu

Operasi pemeliharaan database yang melibatkan pembersihan setelah pembaruan tambahan untuk merebut kembali penyimpanan dan meningkatkan kinerja.

### kendali versi

Proses dan alat yang melacak perubahan, seperti perubahan kode sumber dalam repositori.

### Peering VPC

Koneksi antara dua VPC yang memungkinkan Anda merutekan lalu lintas dengan menggunakan alamat IP pribadi. Untuk informasi selengkapnya, lihat [Apa itu peering VPC](#) di dokumentasi VPC Amazon.

### kerentanan

Kelemahan perangkat lunak atau perangkat keras yang membahayakan keamanan sistem.

## W

### cache hangat

Cache buffer yang berisi data terkini dan relevan yang sering diakses. Instance database dapat membaca dari cache buffer, yang lebih cepat daripada membaca dari memori utama atau disk.

### data hangat

Data yang jarang diakses. Saat menanyakan jenis data ini, kueri yang cukup lambat biasanya dapat diterima.

## fungsi jendela

Fungsi SQL yang melakukan perhitungan pada sekelompok baris yang berhubungan dengan catatan saat ini. Fungsi jendela berguna untuk memproses tugas, seperti menghitung rata-rata bergerak atau mengakses nilai baris berdasarkan posisi relatif dari baris saat ini.

## beban kerja

Kumpulan sumber daya dan kode yang memberikan nilai bisnis, seperti aplikasi yang dihadapi pelanggan atau proses backend.

## aliran kerja

Grup fungsional dalam proyek migrasi yang bertanggung jawab atas serangkaian tugas tertentu. Setiap alur kerja independen tetapi mendukung alur kerja lain dalam proyek. Misalnya, alur kerja portofolio bertanggung jawab untuk memprioritaskan aplikasi, perencanaan gelombang, dan mengumpulkan metadata migrasi. Alur kerja portofolio mengirimkan aset ini ke alur kerja migrasi, yang kemudian memigrasikan server dan aplikasi.

## CACING

Lihat [menulis sekali, baca banyak](#).

## WQF

Lihat [AWS Kerangka Kualifikasi Beban Kerja](#).

## tulis sekali, baca banyak (WORM)

Model penyimpanan yang menulis data satu kali dan mencegah data dihapus atau dimodifikasi. Pengguna yang berwenang dapat membaca data sebanyak yang diperlukan, tetapi mereka tidak dapat mengubahnya. Infrastruktur penyimpanan data ini dianggap [tidak dapat diubah](#).

## Z

### eksploitasi zero-day

Serangan, biasanya malware, yang memanfaatkan kerentanan [zero-day](#).

### kerentanan zero-day

Cacat atau kerentanan yang tak tanggung-tanggung dalam sistem produksi. Aktor ancaman dapat menggunakan jenis kerentanan ini untuk menyerang sistem. Pengembang sering menyadari kerentanan sebagai akibat dari serangan tersebut.

## bidikan zero-shot

Memberikan [LLM](#) dengan instruksi untuk melakukan tugas tetapi tidak ada contoh (tembakan) yang dapat membantu membimbingnya. LLM harus menggunakan pengetahuan pra-terlatih untuk menangani tugas. Efektivitas bidikan nol tergantung pada kompleksitas tugas dan kualitas prompt. Lihat juga beberapa [bidikan yang diminta](#).

## aplikasi zombie

Aplikasi yang memiliki CPU rata-rata dan penggunaan memori di bawah 5 persen. Dalam proyek migrasi, adalah umum untuk menghentikan aplikasi ini.

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.