Memulai dengan ETL tanpa server aktif AWS Glue

AWS Panduan Preskriptif

AWS Panduan Preskriptif: Memulai dengan ETL tanpa server aktif AWS Glue

Table of Contents

Memulai dengan ETL tanpa server aktif AWS Glue

Dheer Toprani dan Adnan Alvee, Amazon Web Services ()AWS

Maret 2024 (sejarah dokumen)

Di Amazon Web Services (AWS) Cloud, AWS Glueterdapat lingkungan tanpa server yang dikelola sepenuhnya tempat Anda dapat mengekstrak, mengubah, dan memuat data (ETL) dalam skala besar. Dengan AWS Glue, Anda dapat mengkategorikan data, membersihkannya, memperkayanya, dan memindahkannya dengan andal di berbagai penyimpanan dan aliran data dengan cara yang hemat biaya.

AWS Glue tanpa server, jadi Anda tidak perlu khawatir tentang penyediaan atau pengelolaan server. Dengan AWS Glue, Anda hanya membayar untuk sumber daya yang Anda gunakan, dan Anda dapat meningkatkan atau menurunkan sesuai kebutuhan.

AWS Glue terdiri dari komponen-komponen berikut:

- AWS Glue ETL — AWS Glue ETL menyediakan opsi batch dan streaming untuk mengekstrak, mengubah, dan memuat data dari satu sumber ke sumber lainnya.
- AWS Glue Data Catalog— Katalog Data adalah repositori pusat untuk mengatur metadata semua aset data Anda. Katalog Data menyediakan antarmuka terpadu tempat Anda dapat mencari, menemukan, dan berbagi aset data di seluruh layanan analisis data.
- AWS Glue DataBrew— DataBrew adalah alat persiapan data tanpa kode yang dapat Anda gunakan untuk mengeksplorasi, membersihkan, dan mengubah data secara visual. Anda dapat memilih dari lebih dari 250 transformasi bawaan untuk mengotomatiskan tugas persiapan data tanpa menulis kode apa pun.

Panduan ini memberikan pengantar tingkat tinggi AWS Glue, termasuk cara kerjanya dan bagaimana Anda bisa mulai menggunakannya. Ini mencakup konsep-konsep kunci yang perlu Anda ketahui sebelum menulis AWS Glue pekerjaan, seperti otomatisasi, pemantauan, dan integrasi dengan layanan lain AWS . Bagian Langkah selanjutnya akan membuat Anda mempercepat penulisan kode AWS Glue. Jika Anda sudah memiliki pengalaman menggunakan AWS Glue, bagian Praktik terbaik akan membantu Anda mengisi kesenjangan dalam pengetahuan Anda. Pada akhir panduan ini, Anda akan dilengkapi dengan pengetahuan dan sumber daya yang Anda butuhkan untuk mulai menggunakan AWS Glue secara efektif.

AWS Glue ETL

AWS Glue ETL mendukung penggalian data dari berbagai sumber, mengubahnya untuk memenuhi kebutuhan bisnis Anda, dan memuatnya ke tujuan pilihan Anda. Layanan ini menggunakan mesin Apache Spark untuk mendistribusikan beban kerja data besar di seluruh node pekerja, memungkinkan transformasi yang lebih cepat dengan pemrosesan dalam memori.

AWS Glue mendukung berbagai sumber data, termasuk Amazon Simple Storage Service (Amazon S3), Amazon DynamoDB, dan Amazon Relational Database Service (Amazon RDS). Untuk mempelajari lebih lanjut tentang sumber data yang didukung, lihat Jenis dan opsi sambungan untuk ETL di AWS Glue.

Penulisan di AWS Glue

AWS Glue menyediakan beberapa cara untuk membuat pekerjaan ETL, tergantung pada pengalaman dan kasus penggunaan Anda:

- Pekerjaan shell Python dirancang untuk menjalankan skrip ETL dasar yang ditulis dengan Python. Pekerjaan ini berjalan pada satu mesin, dan lebih cocok untuk kumpulan data kecil atau menengah.
- Pekerjaan Apache Spark dapat ditulis dalam Python atau Scala. Pekerjaan ini menggunakan Spark untuk menskalakan beban kerja secara horizontal di banyak node pekerja, sehingga mereka dapat menangani kumpulan data besar dan transformasi kompleks.
- AWS Glue streaming ETL menggunakan mesin Apache Spark Structured Streaming untuk mengubah data streaming dalam pekerjaan micro-batch menggunakan semantik yang tepat sekali. Anda dapat membuat pekerjaan AWS Glue streaming dengan Python atau Scala.
- AWS Glue Studioadalah antarmuka boxes-and-arrows gaya visual untuk membuat ETL berbasis Spark dapat diakses oleh pengembang yang baru mengenal pemrograman Apache Spark.

Unit pengolahan data

AWS Glue menggunakan unit pemrosesan data (DPUs) untuk mengukur sumber daya komputasi yang dialokasikan untuk pekerjaan ETL dan menghitung biaya. Setiap DPU setara dengan memori 4 v CPUs dan 16 GB. DPUsharus dialokasikan untuk AWS Glue pekerjaan Anda tergantung pada kompleksitas dan volume datanya. Mengalokasikan jumlah yang sesuai DPUs akan memungkinkan Anda untuk menyeimbangkan kebutuhan kinerja dengan kendala biaya.

AWS Glue menyediakan beberapa jenis pekerja yang dioptimalkan untuk berbagai beban kerja:

- G.1X atau G.2X (untuk sebagian besar data mengubah, bergabung, dan kueri)
- G.4X atau G.8X (untuk transformasi data, agregasi, gabungan, dan kueri yang lebih menuntut)
- G.025X (untuk aliran data volume rendah dan sporadis)
- Standar (untuk AWS Glue versi 1.0 atau yang lebih lama; tidak direkomendasikan untuk versi yang lebih baru AWS Glue)

Menggunakan shell Python

Untuk pekerjaan shell Python, Anda dapat menggunakan 1 DPU untuk menggunakan memori 16 GB atau 0,0625 DPU untuk menggunakan memori 1 GB. Shell Python ditujukan untuk pekerjaan ETL dasar dengan kumpulan data kecil atau menengah (hingga sekitar 10 GB).

Membandingkan jenis pekerja

Tabel berikut menunjukkan jenis AWS Glue pekerja yang berbeda untuk beban kerja batch, streaming, dan AWS Glue Studio ETL menggunakan lingkungan Apache Spark.

	G.1X	G.2X	G.4X	G.8X	G.025X	Standar
vCPU	4	8	16	32	2	4
Memori	16 GB	32 GB	64 GB	128 GB	4 GB	16 GB
Ruang disk	64 GB	128 GB	256 GB	512 GB	64 GB	50 GB
Pelaksana per pekerja	1	1	1	1	1	2
DPU	1	2	4	8	0,25	1

Jenis pekerja Standar tidak disarankan untuk AWS Glue versi 2.0 dan yang lebih baru. Jenis pekerja G.025X hanya tersedia untuk pekerjaan streaming menggunakan AWS Glue versi 3.0 atau yang lebih baru.

AWS Glue Data Catalog

AWS Glue Data CatalogIni adalah repositori metadata terpusat untuk semua aset data Anda di berbagai sumber data. Ini menyediakan antarmuka terpadu untuk menyimpan dan menanyakan informasi tentang format data, skema, dan sumber. Ketika pekerjaan AWS Glue ETL berjalan, ia menggunakan katalog ini untuk memahami informasi tentang data dan memastikan bahwa itu diubah dengan benar.

AWS Glue Data CatalogIni terdiri dari komponen-komponen berikut:

- Database dan tabel
- Crawler dan pengklasifikasi
- Koneksi
- Registri Skema

AWS Glue database dan tabel

AWS Glue Data CatalogIni diatur ke dalam database dan tabel untuk menyediakan struktur logis untuk menyimpan dan mengelola metadata. Struktur ini mendukung kontrol akses data yang tepat pada tingkat tabel atau database dengan menggunakan kebijakan AWS Identity and Access Management (IAM).

AWS Glue Database dapat berisi banyak tabel, dan setiap tabel harus dikaitkan dengan database tunggal. Tabel ini berisi referensi ke data aktual, yang dapat disimpan di salah satu dari berbagai sumber data yang AWS Glue mendukung. AWS Glue tabel juga menyimpan metadata penting seperti nama kolom, tipe data, dan kunci partisi.

Ada beberapa metode berbeda untuk membuat tabel di AWS Glue:

- AWS Glue perayap
- AWS Glue Pekerjaan ETL
- AWS Glue konsol
- `CreateTable`operasi di AWS Glue API
- AWS CloudFormation Template
- AWS Cloud Development Kit (AWS CDK)

- Metastore Apache Hive yang bermigrasi

AWS Glue crawler dan pengklasifikasi

AWS Glue Crawler secara otomatis menemukan dan mengekstrak metadata dari penyimpanan data, dan kemudian memperbarui yang sesuai. AWS Glue Data Catalog Crawler terhubung ke penyimpanan data untuk menyimpulkan skema data. Kemudian membuat atau memperbarui tabel dalam Katalog Data dengan informasi skema yang ditemukannya. Crawler dapat merayapi penyimpanan data berbasis file dan berbasis tabel. Untuk mempelajari lebih lanjut tentang penyimpanan data yang didukung, lihat Penyimpanan data mana yang dapat saya jelajahi?

Crawler menggunakan pengklasifikasi untuk mengenali format data secara akurat dan menentukan bagaimana seharusnya diproses. Secara default, crawler menggunakan satu set pengklasifikasi bawaan umum yang disediakan oleh AWS Glue, tetapi Anda juga dapat menulis pengklasifikasi khusus untuk menangani kasus penggunaan tertentu.

AWS Glue koneksi

Anda dapat menggunakan AWS Glue koneksi untuk menentukan parameter koneksi yang memungkinkan AWS Glue untuk terhubung ke berbagai sumber data. Menambahkan koneksi memusatkan dan menyederhanakan konfigurasi yang diperlukan untuk terhubung ke sumber-sumber ini.

Saat menentukan koneksi, Anda menentukan jenis koneksi, titik akhir koneksi, dan kredensi apa pun yang diperlukan. Setelah koneksi didefinisikan, itu dapat digunakan kembali oleh beberapa AWS Glue pekerjaan dan crawler. Menggunakan koneksi dengan AWS Glue mengurangi kebutuhan untuk berulang kali memasukkan informasi koneksi yang sama, seperti kredensi login atau virtual private cloud (VPC). IDs

AWS Glue Registri Skema

Registri AWS Glue Skema menyediakan lokasi terpusat untuk mengelola dan menegakkan skema aliran data. Ini memungkinkan sistem yang berbeda, seperti produsen data dan konsumen, untuk berbagi skema untuk serialisasi dan deserialisasi. Berbagi skema membantu sistem ini untuk berkomunikasi secara efektif dan menghindari kesalahan selama transformasi.

Registri Skema memastikan bahwa konsumen data hilir dapat menangani perubahan yang dilakukan di hulu, karena mereka mengetahui skema yang diharapkan. Ini mendukung evolusi skema, sehingga

skema dapat berubah dari waktu ke waktu sambil mempertahankan kompatibilitas dengan versi skema sebelumnya.

Registri Skema terintegrasi dengan banyak AWS layanan, termasuk Amazon Kinesis Data Streams, Firehose, dan Amazon Managed Streaming untuk Apache Kafka. Untuk contoh kasus penggunaan dan integrasi, lihat Mengintegrasikan dengan Registri AWS Glue Skema.

Fitur dan konsep penting

Pencatatan log dan pemantauan

AWS Glue memiliki beberapa opsi pencatatan dan pemantauan. Secara default, AWS Glue mengirim log ke grup `aws-glue` log di Amazon CloudWatch. Log ini mencakup informasi seperti waktu mulai dan berakhir, pengaturan konfigurasi, dan kesalahan atau peringatan apa pun yang mungkin terjadi.

Selain itu, pekerjaan AWS Glue Spark ETL menyediakan opsi berikut, yang harus diaktifkan untuk pemantauan lanjutan:

- Metrik Job melaporkan metrik spesifik pekerjaan ke AWS Glue namespace setiap 30 detik. CloudWatch Metrik spesifik pekerjaan ini, seperti catatan yang diproses, ukuran input/output data total, dan waktu proses, memberikan wawasan tentang kinerja pekerjaan. Mereka dapat membantu mengidentifikasi kemacetan atau peluang untuk mengoptimalkan konfigurasi.
- Pencatatan berkelanjutan mengalirkan log pekerjaan Apache Spark real-time ke grup `/aws-glue/jobs/logs-v2` log di. CloudWatch Dengan menggunakan log waktu nyata, Anda dapat memantau AWS Glue pekerjaan secara dinamis saat sedang berjalan.
- Spark UI menyediakan antarmuka web server riwayat Spark untuk melihat informasi tentang pekerjaan Spark, seperti timeline acara setiap tahap, grafik asiklik terarah, dan variabel lingkungan kerja. Log peristiwa Spark UI yang bertahan disimpan di Amazon S3, dan Anda dapat menggunakannya secara real time atau setelah pekerjaan selesai.
- Wawasan Job run menyederhanakan debugging dan pengoptimalan pekerjaan dengan mendengarkan pengecualian Spark umum, melakukan analisis akar penyebab, dan memberikan tindakan yang disarankan untuk memperbaiki masalah. Wawasan disimpan di CloudWatch.

Otomatisasi

AWS Glue menyediakan dua cara utama bagi Anda untuk mengotomatiskan pekerjaan ETL: pemicu dan alur kerja.

AWS Glue pemicu

Saat diaktifkan, AWS Glue pemicu memulai pekerjaan dan perayap yang ditentukan. Pemicu dapat ditembakkan sesuai permintaan, berdasarkan jadwal yang telah ditentukan, atau berdasarkan

peristiwa tertentu. Anda dapat menggunakan pemicu untuk merancang rantai pekerjaan dan crawler yang bergantung. Untuk informasi lebih lanjut, lihat AWS Glue pemicu.

AWS Glue alur kerja

Untuk beban kerja yang lebih kompleks, Anda dapat menggunakan AWS Glue alur kerja untuk membuat grafik asiklik terarah dan membangun dependensi antara AWS Glue entitas terpisah (pemicu, perayap, dan pekerjaan). Alur kerja juga menyediakan antarmuka terpadu tempat Anda dapat berbagi parameter, memantau kemajuan, dan memecahkan masalah di seluruh entitas terkait.

Menyiapkan banyak entitas terkait dalam AWS Glue alur kerja dapat tumbuh semakin kompleks. Pengembang dapat membuat AWS Glue cetak biru untuk berbagi jaringan data yang kompleks dengan ilmuwan data dan analis bisnis. Template ini memungkinkan pembuatan AWS Glue alur kerja yang konsisten dan berulang, mengabstraksi detail teknis.

Untuk mempelajari selengkapnya tentang AWS Glue cetak biru dan alur kerja, lihat Melakukan aktivitas ETL yang kompleks menggunakan cetak biru dan alur kerja di. AWS Glue

Mengatur AWS Glue pekerjaan dengan layanan lain AWS

Untuk opsi otomatisasi lainnya, AWS Glue integrasikan dengan AWS layanan lain, seperti, AWS Lambda AWS Step Functions, dan Alur Kerja Terkelola Amazon untuk Apache Airflow (Amazon MWAA).

Untuk informasi selengkapnya tentang metode orkestrasi untuk pekerjaan AWS Glue ETL, lihat Orkestrasi di. AWS Glue

Bookmark tugas

Bookmark Job AWS Glue digunakan untuk melacak kemajuan pekerjaan ETL, yang mencegah kebutuhan untuk memproses ulang data dalam menjalankan pekerjaan berikutnya. Ketika bookmark pekerjaan diaktifkan, AWS Glue menyimpan catatan data yang telah diproses. Kemudian dengan setiap proses, hanya memproses data baru di sumber data. Untuk informasi selengkapnya, lihat Melacak data yang diproses menggunakan bookmark pekerjaan.

AWS Glue DataBrew

AWS Glue DataBrew adalah layanan persiapan data visual yang dikelola sepenuhnya untuk membersihkan, menormalkan, dan mengubah data. Ini berbeda dari AWS Glue ETL karena Anda tidak memiliki kode tulis untuk bekerja dengannya. DataBrew menyediakan lebih dari 250 transformasi bawaan, dengan point-and-click antarmuka visual untuk membuat dan mengelola pekerjaan transformasi data.

DataBrew tersedia dalam tampilan konsol terpisah dari AWS Glue. Ini terintegrasi secara native dengan beberapa AWS layanan dan mendukung banyak format file yang berbeda. Untuk informasi selengkapnya, lihat Integrasi produk dan layanan.

DataBrew didasarkan pada enam konsep inti berikut:

- Proyek — Seluruh ruang kerja persiapan data di DataBrew
- Dataset — Kumpulan data terstruktur atau semi-terstruktur
- Resep — Satu set langkah transformasi data; setiap langkah dapat berisi banyak tindakan
- Job — Satu set instruksi untuk menjalankan resep atau pekerjaan profil data
- Silsilah data — Pelacakan data dalam antarmuka visual untuk mengidentifikasi asal-usulnya
- Profil data — Tampilan ringkasan dari bentuk data Anda

AWS Glue DataBrew terintegrasi dengan AWS Glue Studio, sehingga Anda dapat mengatur DataBrew resep dalam pekerjaan dan alur kerja AWS Glue ETL Anda. DataBrew resep juga dapat memanfaatkan AWS Glue fitur seperti bookmark pekerjaan, percobaan ulang otomatis, dan penskalaan otomatis. Untuk memulai DataBrew, gunakan AWS Glue DataBrew contoh tutorial proyek.

Praktik terbaik

Saat mengembangkan dengan AWS Glue, pertimbangkan praktik terbaik berikut.

Kembangkan secara lokal terlebih dahulu

Untuk menghemat biaya dan waktu saat membangun pekerjaan ETL Anda, uji kode dan logika bisnis Anda secara lokal terlebih dahulu. Untuk petunjuk tentang menyiapkan wadah Docker yang dapat membantu Anda menguji pekerjaan AWS Glue ETL baik di shell maupun di lingkungan pengembangan terintegrasi (IDE), lihat posting blog Mengembangkan dan menguji AWS Glue pekerjaan secara lokal menggunakan wadah Docker.

Gunakan sesi AWS Glue interaktif

AWS Glue sesi interaktif menyediakan backend Spark tanpa server, ditambah dengan kernel Jupyter open-source yang terintegrasi dengan notebook dan seperti, IntelliJ, dan VS Code. IDEs PyCharm Dengan menggunakan sesi interaktif, Anda dapat menguji kode Anda pada kumpulan data nyata dengan backend AWS Glue Spark dan IDE pilihan Anda. Untuk memulai, ikuti langkah-langkah di Memulai dengan sesi AWS Glue interaktif.

Gunakan partisi untuk menanyakan apa yang Anda butuhkan

Partisi mengacu pada membagi kumpulan data besar menjadi partisi yang lebih kecil berdasarkan kolom atau kunci tertentu. Ketika data dipartisi, AWS Glue dapat melakukan pemindaian selektif pada subset data yang memenuhi kriteria partisi tertentu, daripada memindai seluruh dataset. Ini menghasilkan pemrosesan kueri yang lebih cepat dan lebih efisien, terutama ketika bekerja dengan kumpulan data besar.

Data partisi berdasarkan kueri yang akan dijalankan terhadapnya. Misalnya, jika sebagian besar kueri memfilter pada kolom tertentu, partisi pada kolom itu dapat sangat mengurangi waktu kueri. Untuk mempelajari lebih lanjut tentang mempartisi data, lihat Bekerja dengan data yang dipartisi di. AWS Glue

Optimalkan manajemen memori

Manajemen memori sangat penting ketika menulis pekerjaan AWS Glue ETL karena mereka berjalan pada mesin Apache Spark, yang dioptimalkan untuk pemrosesan dalam memori. Posting blog

Optimalkan manajemen memori dalam AWS Glue memberikan rincian tentang teknik manajemen memori berikut:

- Implementasi daftar Amazon S3 AWS Glue
- Pengelompokan
- Tidak termasuk jalur Amazon S3 dan kelas penyimpanan yang tidak relevan
- Percikan dan AWS Glue baca partisi
- Sisipan massal
- Bergabunglah dengan pengoptimalan
- PySpark fungsi yang ditentukan pengguna () UDFs
- Pemrosesan inkremental

Gunakan format penyimpanan data yang efisien

Saat membuat pekerjaan ETL, kami sarankan untuk mengeluarkan data yang diubah dalam format data berbasis kolom. Format data kolumnar, seperti Apache Parquet dan ORC, biasanya digunakan dalam penyimpanan data besar dan analitik. Mereka dirancang untuk meminimalkan pergerakan data dan memaksimalkan kompresi. Format ini juga memungkinkan pemisahan data ke beberapa pembaca untuk meningkatkan pembacaan paralel selama pemrosesan kueri.

Mengompresi data juga membantu mengurangi jumlah data yang disimpan, dan meningkatkan kinerja operasi baca/tulis. AWS Glue mendukung beberapa format kompresi secara native. Untuk informasi selengkapnya tentang penyimpanan dan kompresi data yang efisien, lihat Membangun pipa data yang efisien kinerja.

Gunakan jenis penskalaan yang sesuai

Memahami kapan harus menskalakan secara horizontal (mengubah jumlah pekerja) atau menskalakan secara vertikal (mengubah jenis pekerja) penting AWS Glue karena dapat berdampak pada biaya dan kinerja pekerjaan ETL.

Umumnya, pekerjaan ETL dengan transformasi kompleks lebih intensif memori dan memerlukan penskalaan vertikal (misalnya, berpindah dari G.1X ke tipe pekerja G.2X). Untuk pekerjaan ETL intensif komputasi dengan volume data yang besar, kami merekomendasikan penskalaan horizontal karena AWS Glue dirancang untuk memproses data tersebut secara paralel di beberapa node.

Memantau metrik AWS Glue pekerjaan di Amazon dengan cermat CloudWatch membantu Anda menentukan apakah hambatan kinerja disebabkan oleh kurangnya memori atau komputasi. Untuk informasi selengkapnya tentang jenis dan penskalaan AWS Glue pekerja, lihat Praktik terbaik untuk menskalakan pekerjaan Apache Spark dan data partisi dengan. AWS Glue

ETL Tanpa Server pada FAQ AWS Glue

Bagian ini memberikan jawaban atas pertanyaan yang sering diajukan tentang ETL tanpa server. AWS Glue

Kapan saya harus menggunakan shell Python alih-alih Apache Spark untuk pekerjaan? AWS Glue

Gunakan Python shell ketika Anda memiliki pekerjaan ETL dasar atau kumpulan data kecil yang tidak memerlukan kemampuan komputasi terdistribusi Apache Spark. Gunakan Apache Spark untuk pekerjaan ETL yang lebih kompleks atau kumpulan data besar yang membutuhkan daya pemrosesan tinggi yang dioptimalkan untuk ditangani oleh Spark.

Apa AWS Glue versi yang direkomendasikan untuk proyek saya?

Kami biasanya merekomendasikan menggunakan versi terbaru dari AWS Glue. Halaman AWS Glue versi mencantumkan perbedaan antar versi, bersama dengan kompatibilitasnya dengan berbagai versi Python dan Spark.

Langkah selanjutnya

Memahami AWS Glue transformasi

Untuk pemrosesan data yang lebih efisien, AWS Glue termasuk fungsi transformasi bawaan. Fungsi berpindah dari transformasi ke transformasi dalam struktur data yang disebut a DynamicFrame, yang merupakan ekstensi ke Apache Spark SQL. DataFrame A DynamicFrame mirip dengan a DataFrame, kecuali bahwa setiap catatan menggambarkan diri sendiri, jadi tidak ada skema yang diperlukan pada awalnya.

Untuk berkenalan dengan beberapa fungsi AWS Glue PySpark bawaan, lihat posting blog Membangun pipa AWS Glue ETL secara lokal tanpa file. Akun AWS

Menulis pekerjaan ETL pertama Anda

Jika Anda belum pernah menulis pekerjaan ETL sebelumnya, Anda dapat memulai dengan menggunakan Tiga jenis pekerjaan AWS Glue ETL untuk mengonversi data ke pola Apache Parquet.

Jika Anda memiliki pengalaman menulis pekerjaan ETL, Anda dapat menggunakan AWS Glue GitHub contoh untuk mengeksplorasi lebih dalam.

Harga

Untuk informasi harga, lihat Harga AWS Glue. Anda juga dapat menggunakan AWS Kalkulator Hargauntuk memperkirakan biaya bulanan Anda untuk menggunakan AWS Glue komponen yang berbeda.

Sumber daya tambahan

Referensi

- AWS Glue Panduan Pengembang
- AWS Glue DataBrew Panduan Pengembang
- AWS Glue API
- AWS Glue streaming ETL
- AWS Glue Studio
- Pekerjaan shell Python di AWS Glue
- Lowongan kerja Apache Spark
- AWS Glue PySparkmengubah referensi
- Katalog Data dan crawler di AWS Glue
- AWS Glue Registri Skema
- Penebangan dan pemantauan di AWS Glue
- AWS Glue versi

Unggahan blog

- Kembangkan dan uji AWS Glue pekerjaan secara lokal menggunakan wadah Docker
- Optimalkan manajemen memori di AWS Glue
- Praktik terbaik untuk menskalakan pekerjaan Apache Spark dan data partisi dengan AWS Glue
- Membangun pipa AWS Glue ETL secara lokal tanpa Akun AWS
- Bekerja dengan data yang dipartisi di AWS Glue

Contoh

- AWS Glue DataBrew proyek sampel
- Memulai dengan sesi AWS Glue interaktif
- AWS Glue Repositori Sampel Kode ETL

Pola

- Buat pipeline layanan ETL untuk memuat data secara bertahap dari Amazon S3 ke Amazon Redshift menggunakan AWS Glue
- Menyebarkan dan mengelola data lake tanpa server di AWS Cloud dengan menggunakan infrastruktur sebagai kode
- Tiga jenis pekerjaan AWS Glue ETL untuk mengonversi data ke Apache Parquet

Riwayat dokumen

Tabel berikut menjelaskan perubahan signifikan pada panduan ini. Jika Anda ingin diberi tahu tentang pembaruan masa depan, Anda dapat berlangganan umpan RSS.

Perubahan	Deskripsi	Tanggal
Diperbarui	Kami menambahkan informasi tentang jenis pekerja ke bagian AWS Glue ETL.	Maret 13, 2024
Diperbarui	Konten diperbarui di seluruh panduan.	15 Maret 2023
Menambahkan praktik terbaik	Kami menambahkan informasi tentang penggunaan partisi untuk kueri data tertentu.	4 Februari 2021
Publikasi awal	—	29 Januari 2021

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