Erste Schritte mit serverlosem ETL AWS Glue

AWS Prescriptive Guidance

AWS Prescriptive Guidance: Erste Schritte mit serverlosem ETL AWS Glue

Table of Contents

Erste Schritte mit serverlosem ETL AWS Glue

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

März 2024 (Geschichte der Dokumente)

In der Amazon Web Services (AWS) Cloud AWS Gluebefindet sich eine vollständig verwaltete serverlose Umgebung, in der Sie Daten (ETL) in großem Umfang extrahieren, transformieren und laden können. Damit AWS Glue können Sie Daten kategorisieren, bereinigen, anreichern und zuverlässig und kostengünstig zwischen verschiedenen Datenspeichern und Datenströmen verschieben.

AWS Glue ist serverlos, sodass Sie sich keine Gedanken über die Bereitstellung oder Verwaltung von Servern machen müssen. Mit zahlen Sie nur für die Ressourcen AWS Glue, die Sie tatsächlich nutzen, und Sie können nach Bedarf nach oben oder unten skalieren.

AWS Glue besteht aus den folgenden Komponenten:

- AWS Glue ETL — AWS Glue ETL bietet Batch- und Streaming-Optionen zum Extrahieren, Transformieren und Laden von Daten aus einer Quelle in eine andere.
- AWS Glue Data Catalog— Data Catalog ist ein zentrales Repository für die Organisation der Metadaten all Ihrer Datenbestände. Data Catalog bietet eine einheitliche Oberfläche, über die Sie Datenbestände über Datenanalysedienste hinweg suchen, entdecken und gemeinsam nutzen können.
- AWS Glue DataBrew— DataBrew ist ein Tool zur Datenvorbereitung ohne Code, mit dem Sie Daten visuell untersuchen, bereinigen und transformieren können. Sie können aus mehr als 250 vorgefertigten Transformationen wählen, um Datenvorbereitungsaufgaben zu automatisieren, ohne Code schreiben zu müssen.

Dieses Handbuch bietet eine allgemeine Einführung AWS Glue in die Funktionsweise und die ersten Schritte. Es behandelt die wichtigsten Konzepte, die Sie vor der Erstellung von AWS Glue Aufgaben kennen müssen, z. B. Automatisierung, Überwachung und Integration mit anderen AWS Diensten. Im Abschnitt „Nächste Schritte“ werden Sie mit dem Schreiben von Code vertraut gemacht. AWS Glue Wenn Sie bereits Erfahrung mit der Verwendung haben AWS Glue, hilft Ihnen der Abschnitt „Bewährte Methoden“ dabei, etwaige Wissenslücken zu schließen. Am Ende dieses Leitfadens werden Sie mit dem Wissen und den Ressourcen ausgestattet sein, die Sie benötigen, um mit der AWS Glue effektiven Nutzung zu beginnen.

AWS Glue ETL

AWS Glue ETL unterstützt das Extrahieren von Daten aus verschiedenen Quellen, deren Transformation entsprechend Ihren Geschäftsanforderungen und das Laden in ein Ziel Ihrer Wahl. Dieser Service verwendet die Apache Spark-Engine, um Big-Data-Workloads auf die Worker-Nodes zu verteilen und so schnellere Transformationen mit In-Memory-Verarbeitung zu ermöglichen.

AWS Glue unterstützt eine Vielzahl von Datenquellen, darunter Amazon Simple Storage Service (Amazon S3), Amazon DynamoDB und Amazon Relational Database Service (Amazon RDS). Weitere Informationen zu unterstützten Datenquellen finden Sie unter Verbindungstypen und Optionen für ETL in. AWS Glue

Inhaltserstellung in AWS Glue

AWS Glue bietet je nach Erfahrung und Anwendungsfall mehrere Möglichkeiten zum Verfassen von ETL-Jobs:

- Python-Shell-Jobs sind für die Ausführung grundlegender ETL-Skripts konzipiert, die in Python geschrieben wurden. Diese Jobs werden auf einem einzigen Computer ausgeführt und eignen sich besser für kleine oder mittelgroße Datensätze.
- Apache Spark-Jobs können entweder in Python oder Scala geschrieben werden. Diese Jobs verwenden Spark, um Workloads horizontal über viele Worker-Knoten hinweg zu skalieren, sodass sie große Datensätze und komplexe Transformationen verarbeiten können.
- AWS Glue Streaming ETL verwendet die Apache Spark Structured Streaming Engine, um Streaming-Daten mithilfe der Exact-Once-Semantik in Mikro-Batch-Jobs umzuwandeln. Sie können AWS Glue Streaming-Jobs entweder in Python oder Scala erstellen.
- AWS Glue Studioist eine Oberfläche boxes-and-arrows im visuellen Stil, die SPARK-basiertes ETL für Entwickler zugänglich macht, die mit der Apache Spark-Programmierung noch nicht vertraut sind.

Datenverarbeitungseinheiten

AWS Glue verwendet Datenverarbeitungseinheiten (DPUs), um die einem ETL-Job zugewiesenen Rechenressourcen zu messen und die Kosten zu berechnen. Jede DPU entspricht 4 V CPUs und 16 GB Arbeitsspeicher. DPUssollte Ihrem AWS Glue Job je nach Komplexität und Datenvolumen

zugewiesen werden. Durch die Zuweisung der entsprechenden Menge können DPUs Sie Leistungsanforderungen und Kostenbeschränkungen in Einklang bringen.

AWS Glue bietet mehrere Workertypen, die für verschiedene Workloads optimiert sind:

- G.1X oder G.2X (für die meisten Datentransformationen, Verknüpfungen und Abfragen)
- G.4X oder G.8X (für anspruchsvollere Datentransformationen, Aggregationen, Verknüpfungen und Abfragen)
- G.025X (für Datenströme mit geringem Volumen und sporadischen Datenströmen)
- Standard (für AWS Glue Versionen 1.0 oder früher; nicht empfohlen für spätere Versionen von) AWS Glue

Python-Shell verwenden

Für einen Python-Shell-Job können Sie entweder 1 DPU verwenden, um 16 GB Speicher zu verwenden, oder 0,0625 DPU, um 1 GB Speicher zu verwenden. Die Python-Shell ist für grundlegende ETL-Jobs mit kleinen oder mittleren Datensätzen (bis zu etwa 10 GB) vorgesehen.

Arbeitnehmertypen im Vergleich

Die folgende Tabelle zeigt die verschiedenen AWS Glue Worker-Typen für Batch-, Streaming- und AWS Glue Studio ETL-Workloads, die die Apache Spark-Umgebung verwenden.

	G.1X	G2X	G4X	G8X	G.025X	Standard
vCPU	4	8	16	32	2	4
Arbeitsspeicher	16 GB	32 GB	64 GB	128 GB	4 GB	16 GB
Festplattenkapazität	64 GB	128 GB	256 GB	512 GB	64 GB	50 GB
Testamentsvollstrecker pro Arbeiter	1	1	1	1	1	2

DPU	1	2	4	8	0,25	1

Der Standard-Worker-Typ wird für AWS Glue Version 2.0 und höher nicht empfohlen. Der Workertyp G.025X ist nur für Streaming-Jobs mit AWS Glue Version 3.0 oder höher verfügbar.

AWS Glue Data Catalog

Das AWS Glue Data Catalogist ein zentralisiertes Metadaten-Repository für all Ihre Datenbestände aus verschiedenen Datenquellen. Es bietet eine einheitliche Oberfläche zum Speichern und Abfragen von Informationen zu Datenformaten, Schemas und Quellen. Wenn ein AWS Glue ETL-Job ausgeführt wird, verwendet er diesen Katalog, um Informationen zu den Daten zu verstehen und sicherzustellen, dass sie korrekt transformiert werden.

Der AWS Glue Data Catalogbesteht aus den folgenden Komponenten:

- Datenbanken und Tabellen
- Crawler und Classifier
- Verbindungen
- Schema Registry

AWS Glue Datenbanken und Tabellen

Die AWS Glue Data Catalogist in Datenbanken und Tabellen unterteilt, um eine logische Struktur für das Speichern und Verwalten von Metadaten bereitzustellen. Diese Struktur unterstützt eine präzise Datenzugriffskontrolle auf Tabellen- oder Datenbankebene mithilfe von AWS Identity and Access Management (IAM-) Richtlinien.

Eine AWS Glue Datenbank kann viele Tabellen enthalten, und jede Tabelle muss einer einzelnen Datenbank zugeordnet sein. Diese Tabellen enthalten Verweise auf die eigentlichen Daten, die in einer der verschiedenen AWS Glue unterstützten Datenquellen gespeichert werden können. AWS Glue Tabellen speichern auch wichtige Metadaten wie Spaltennamen, Datentypen und Partitionsschlüssel.

Es gibt verschiedene Methoden zum Erstellen einer Tabelle in AWS Glue:

- AWS Glue Crawler
- AWS Glue ETL-Job
- AWS Glue Konsole
- `CreateTable`Betrieb in der AWS Glue API
- AWS CloudFormation Vorlage
- AWS Cloud Development Kit (AWS CDK)

- Ein migrierter Apache Hive-Metastore

AWS Glue Crawler und Klassifikatoren

Ein AWS Glue Crawler erkennt und extrahiert automatisch Metadaten aus einem Datenspeicher und aktualisiert sie dann entsprechend. AWS Glue Data Catalog Der Crawler stellt eine Verbindung zum Datenspeicher her, um das Schema der Daten abzuleiten. Anschließend erstellt oder aktualisiert er Tabellen im Datenkatalog mit den gefundenen Schemainformationen. Ein Crawler kann sowohl dateibasierte als auch tabellenbasierte Datenspeicher durchsuchen. Weitere Informationen zu unterstützten Datenspeichern finden Sie unter Welche Datenspeicher kann ich crawlen?

Der Crawler verwendet Klassifikatoren, um das Format von Daten genau zu erkennen und zu bestimmen, wie sie verarbeitet werden sollen. Standardmäßig verwendet der Crawler eine Reihe gängiger integrierter Klassifikatoren, die von bereitgestellt werden. Sie können AWS Glue jedoch auch benutzerdefinierte Klassifikatoren für bestimmte Anwendungsfälle schreiben.

AWS Glue Verbindungen

Sie können AWS Glue Verbindungen verwenden, um Verbindungsparameter zu definieren, die es ermöglichen, eine Verbindung AWS Glue zu verschiedenen Datenquellen herzustellen. Das Hinzufügen von Verbindungen zentralisiert und vereinfacht die Konfiguration, die für die Verbindung mit diesen Quellen erforderlich ist.

Beim Definieren einer Verbindung geben Sie den Verbindungstyp, den Verbindungsendpunkt und alle erforderlichen Anmeldeinformationen an. Nachdem eine Verbindung definiert wurde, kann sie von mehreren AWS Glue Jobs und Crawlern wiederverwendet werden. Die Verwendung von Verbindungen mit AWS Glue reduziert die Notwendigkeit, wiederholt dieselben Verbindungsinformationen wie Anmeldeinformationen oder Virtual Private Cloud (VPC) IDs einzugeben.

AWS Glue Schemaregistrierung

Die AWS Glue Schemaregistry bietet einen zentralen Ort für die Verwaltung und Durchsetzung von Datenstromschemas. Es ermöglicht unterschiedlichen Systemen, wie Datenproduzenten und Datenverbrauchern, die gemeinsame Nutzung eines Schemas für die Serialisierung und Deserialisierung. Die gemeinsame Nutzung eines Schemas hilft diesen Systemen, effektiv zu kommunizieren und Fehler bei der Transformation zu vermeiden.

Die Schemaregistry stellt sicher, dass nachgelagerte Datenverbraucher mit Änderungen umgehen können, die im Upstream vorgenommen wurden, da sie das erwartete Schema kennen. Sie unterstützt die Schemaentwicklung, sodass sich ein Schema im Laufe der Zeit ändern kann und gleichzeitig die Kompatibilität mit früheren Versionen des Schemas gewahrt bleibt.

Die Schema Registry lässt sich in viele AWS Dienste integrieren, darunter Amazon Kinesis Data Streams, Firehose und Amazon Managed Streaming for Apache Kafka. Beispiele für Anwendungsfälle und Integrationen finden Sie unter Integration mit Schema Registry. AWS Glue

Wichtige Funktionen und Konzepte

Protokollierung und Überwachung

AWS Glue hat mehrere Protokollierungs- und Überwachungsoptionen. AWS Glue Sendet standardmäßig Protokolle an die `aws-glue` Protokollgruppe in Amazon CloudWatch. Diese Protokolle enthalten Informationen wie Start- und Endzeit, Konfigurationseinstellungen und eventuell aufgetretene Fehler oder Warnungen.

Darüber hinaus bieten AWS Glue Spark-ETL-Jobs die folgenden Optionen, die für eine erweiterte Überwachung aktiviert sein müssen:

- Job-Metriken melden auftragsspezifische Metriken CloudWatch alle 30 Sekunden an den AWS Glue Namespace. Diese auftragsspezifischen Messwerte, wie z. B. verarbeitete Datensätze, input/output Gesamtdatengröße und Laufzeit, bieten Einblicke in die Leistung eines Jobs. Sie können dabei helfen, Engpässe oder Möglichkeiten zur Optimierung von Konfigurationen zu identifizieren.
- Durch die kontinuierliche Protokollierung werden Apache Spark-Jobprotokolle in Echtzeit an die `/aws-glue/jobs/logs-v2` Protokollgruppe in gestreamt. CloudWatch Mithilfe von Echtzeitprotokollen können Sie AWS Glue Jobs dynamisch überwachen, während sie ausgeführt werden.
- Die Spark-Benutzeroberfläche bietet eine Spark-History-Server-Weboberfläche zum Anzeigen von Informationen über den Spark-Job, wie z. B. die Ereigniszeitleiste jeder Phase, ein gerichtetes azyklisches Diagramm und Job-Umgebungsvariablen. Die persistenten Spark-UI-Ereignisprotokolle werden in Amazon S3 gespeichert, und Sie können sie in Echtzeit oder nach Abschluss des Auftrags verwenden.
- Job Run Insights vereinfacht das Debuggen und Optimieren von Jobs, indem es auf häufig auftretende Spark-Ausnahmen wartet, eine Ursachenanalyse durchführt und Handlungsempfehlungen zur Behebung von Problemen bereitstellt. Die Erkenntnisse werden gespeichert in CloudWatch.

Automatisierung

AWS Glue bietet Ihnen zwei Hauptmethoden zur Automatisierung von ETL-Jobs: Trigger und Workflows.

AWS Glue löst aus

Wenn sie ausgelöst werden, starten AWS Glue Trigger bestimmte Jobs und Crawler. Ein Trigger kann bei Bedarf, auf der Grundlage eines vordefinierten Zeitplans oder auf der Grundlage bestimmter Ereignisse ausgelöst werden. Sie können Trigger verwenden, um eine Kette von abhängigen Jobs und Crawlern zu entwerfen. Weitere Informationen finden Sie unter AWS Glue Trigger.

AWS Glue Workflows

Für komplexere Workloads können Sie AWS Glue Workflows verwenden, um gerichtete azyklische Graphen zu erstellen und Abhängigkeiten zwischen einzelnen AWS Glue Entitäten (Triggern, Crawlern und Jobs) aufzubauen. Workflows bieten außerdem eine einheitliche Oberfläche, über die Sie Parameter gemeinsam nutzen, den Fortschritt überwachen und Probleme in allen zugehörigen Entitäten beheben können.

Die Einrichtung vieler verknüpfter Entitäten innerhalb von AWS Glue Workflows kann immer komplexer werden. Entwickler können AWS Glue Pläne für die gemeinsame Nutzung komplexer Daten-Pipelines mit Datenwissenschaftlern und Geschäftsanalysten erstellen. Diese Vorlagen ermöglichen die konsistente und wiederholbare Erstellung von AWS Glue Workflows, wobei die technischen Details weggelassen werden.

Weitere Informationen zu AWS Glue Blueprints und Workflows finden Sie unter Durchführen komplexer ETL-Aktivitäten mithilfe von Blueprints und Workflows in. AWS Glue

Orchestrierung von AWS Glue Jobs mit anderen Diensten AWS

Für mehr Automatisierungsoptionen AWS Glue lässt es sich in andere AWS Dienste wie AWS Lambda AWS Step Functions, und Amazon Managed Workflows for Apache Airflow (Amazon MWAA) integrieren.

Weitere Informationen zu den Orchestrierungsmethoden für AWS Glue ETL-Jobs finden Sie unter Orchestrierung in. AWS Glue

Auftrags-Lesezeichen

Job-Lesezeichen in AWS Glue werden verwendet, um den Fortschritt von ETL-Jobs zu verfolgen, wodurch verhindert wird, dass Daten in nachfolgenden Jobausführungen erneut verarbeitet werden müssen. Wenn Job-Lesezeichen aktiviert sind AWS Glue , wird eine Aufzeichnung der Daten geführt, die bereits verarbeitet wurden. Anschließend werden bei jedem Lauf nur die neuen Daten in der Datenquelle verarbeitet. Weitere Informationen finden Sie unter Verfolgen verarbeiteter Daten mithilfe von Job-Lesezeichen.

AWS Glue DataBrew

AWS Glue DataBrew ist ein vollständig verwalteter Service zur visuellen Datenvorbereitung zur Bereinigung, Normalisierung und Transformation von Daten. Er unterscheidet sich von AWS Glue ETL dadurch, dass Sie keinen Code schreiben müssen, um damit zu arbeiten. DataBrew bietet mehr als 250 integrierte Transformationen mit einer visuellen point-and-click Oberfläche für die Erstellung und Verwaltung von Datentransformationsaufträgen.

DataBrew ist in einer separaten Konsolenansicht von AWS Glue verfügbar. Es ist nativ in mehrere AWS Dienste integriert und unterstützt viele verschiedene Dateiformate. Weitere Informationen finden Sie unter Produkt- und Serviceintegrationen.

DataBrew basiert auf den folgenden sechs Kernkonzepten:

- Projekt — Der gesamte Arbeitsbereich zur Datenaufbereitung in DataBrew
- Datensatz — Eine Sammlung strukturierter oder halbstrukturierter Daten
- Rezept — Eine Reihe von Schritten zur Datentransformation; jeder Schritt kann viele Aktionen beinhalten
- Job — Eine Reihe von Anweisungen zum Ausführen eines Rezepts- oder Datenprofiljobs
- Datenherkunft — Die Verfolgung von Daten in einer visuellen Oberfläche, um ihren Ursprung zu identifizieren
- Datenprofil — Eine zusammenfassende Ansicht der Form Ihrer Daten

AWS Glue DataBrew ist in integriert AWS Glue Studio, sodass Sie DataBrew Rezepte innerhalb Ihrer AWS Glue ETL-Jobs und Workflows orchestrieren können. DataBrew Rezepte können auch AWS Glue Funktionen wie Job-Lesezeichen, automatische Wiederholungsversuche und automatische Skalierung nutzen. Verwenden Sie zunächst DataBrew das Tutorial zum AWS Glue DataBrew Beispielprojekt.

Bewährte Methoden

Beachten Sie bei der Entwicklung mit AWS Glue die folgenden bewährten Methoden.

Entwickeln Sie zuerst lokal

Um bei der Erstellung Ihrer ETL-Jobs Kosten und Zeit zu sparen, sollten Sie Ihren Code und Ihre Geschäftslogik zunächst lokal testen. Anweisungen zum Einrichten eines Docker-Containers, mit dem Sie AWS Glue ETL-Jobs sowohl in einer Shell als auch in einer integrierten Entwicklungsumgebung (IDE) testen können, finden Sie im Blogbeitrag AWS Glue Jobs lokal mithilfe eines Docker-Containers entwickeln und testen.

Verwenden Sie interaktive Sitzungen AWS Glue

AWS Glue interaktive Sitzungen bieten ein serverloses Spark-Backend in Verbindung mit einem Open-Source-Jupyter-Kernel, der sich in Notebooks IDEs wie PyCharm IntelliJ und VS Code integrieren lässt. Mithilfe interaktiver Sitzungen können Sie Ihren Code mit dem Spark-Backend und der IDE Ihrer Wahl an echten Datensätzen testen. AWS Glue Folgen Sie zunächst den Schritten unter Erste Schritte mit AWS Glue interaktiven Sitzungen.

Verwenden Sie Partitionierung, um genau das abzufragen, was Sie benötigen

Partitionierung bezieht sich auf die Aufteilung eines großen Datensatzes in kleinere Partitionen, die auf bestimmten Spalten oder Schlüsseln basieren. Wenn Daten partitioniert sind, AWS Glue können selektive Scans für eine Teilmenge von Daten durchgeführt werden, die bestimmte Partitionierungskriterien erfüllen, anstatt den gesamten Datensatz zu scannen. Dies führt zu einer schnelleren und effizienteren Abfrageverarbeitung, insbesondere bei der Arbeit mit großen Datensätzen.

Partitionieren Sie Daten auf der Grundlage der Abfragen, die für sie ausgeführt werden. Wenn die meisten Abfragen beispielsweise nach einer bestimmten Spalte filtern, kann die Partitionierung nach dieser Spalte die Abfragezeit erheblich reduzieren. Weitere Informationen zum Partitionieren von Daten finden Sie unter Arbeiten mit partitionierten Daten in. AWS Glue

Optimieren Sie die Speicherverwaltung

Die Speicherverwaltung ist beim Schreiben von AWS Glue ETL-Jobs von entscheidender Bedeutung, da sie auf der Apache Spark-Engine ausgeführt werden, die für die In-Memory-Verarbeitung optimiert ist. Der Blogbeitrag Optimieren Sie die Speicherverwaltung in AWS Glue enthält Einzelheiten zu den folgenden Speicherverwaltungstechniken:

- Implementierung der Amazon S3 S3-Liste von AWS Glue
- Gruppierung
- Ohne irrelevante Amazon S3 S3-Pfade und Speicherklassen
- Partitionierung mit Spark und AWS Glue Read
- Beilagen in großen Mengen
- Optimierungen verbinden
- PySpark benutzerdefinierte Funktionen () UDFs
- Inkrementelle Verarbeitung

Verwenden Sie effiziente Datenspeicherformate

Bei der Erstellung von ETL-Jobs empfehlen wir, transformierte Daten in einem spaltenbasierten Datenformat auszugeben. Spaltenförmige Datenformate wie Apache Parquet und ORC werden häufig bei der Speicherung und Analyse großer Datenmengen verwendet. Sie wurden entwickelt, um Datenbewegungen zu minimieren und die Komprimierung zu maximieren. Diese Formate ermöglichen auch die Aufteilung von Daten auf mehrere Leser, um mehr parallel Lesevorgänge während der Abfrageverarbeitung zu erreichen.

Das Komprimieren von Daten trägt auch dazu bei, die Menge der gespeicherten Daten zu reduzieren, und es verbessert die Leistung von Lese-/Schreibvorgängen. AWS Glue unterstützt mehrere Komprimierungsformate nativ. Weitere Informationen zur effizienten Datenspeicherung und Komprimierung finden Sie unter Aufbau einer leistungseffizienten Datenpipeline.

Verwenden Sie die entsprechende Art der Skalierung

Es ist wichtig zu wissen, wann eine horizontale Skalierung (Änderung der Anzahl der Mitarbeiter) oder eine vertikale Skalierung (Änderung des Mitarbeitertyps) erforderlich ist, AWS Glue da sich dies auf die Kosten und Leistung der ETL-Jobs auswirken kann.

Im Allgemeinen sind ETL-Jobs mit komplexen Transformationen speicherintensiver und erfordern eine vertikale Skalierung (z. B. die Umstellung vom G.1X- auf den G.2X-Worker-Typ). Für rechenintensive ETL-Jobs mit großen Datenmengen empfehlen wir die horizontale Skalierung, da sie darauf ausgelegt AWS Glue ist, diese Daten parallel über mehrere Knoten zu verarbeiten.

Durch die genaue Überwachung der AWS Glue Job-Metriken in Amazon CloudWatch können Sie feststellen, ob ein Leistungsengpass auf einen Mangel an Arbeitsspeicher oder Rechenleistung zurückzuführen ist. Weitere Informationen zu AWS Glue Workertypen und Skalierung finden Sie unter Bewährte Methoden zur Skalierung von Apache Spark-Jobs und zur Partitionierung von Daten mit. AWS Glue

Serverloses ETL auf Häufig gestellte Fragen AWS Glue

Dieser Abschnitt enthält Antworten auf häufig gestellte Fragen zu serverless ETL on. AWS Glue

Wann sollte ich Python-Shell anstelle von Apache Spark für AWS Glue Jobs verwenden?

Verwenden Sie die Python-Shell, wenn Sie einfache ETL-Jobs oder kleine Datensätze haben, die die verteilten Rechenfunktionen von Apache Spark nicht benötigen. Verwenden Sie Apache Spark für komplexere ETL-Jobs oder große Datensätze, die die hohe Verarbeitungsleistung benötigen, für die Spark optimiert ist.

Was ist die empfohlene AWS Glue Version für mein Projekt?

Wir empfehlen generell, die neueste Version von zu verwenden AWS Glue. Auf der AWS Glue Versionsseite werden die Unterschiede zwischen den Versionen sowie deren Kompatibilität mit verschiedenen Versionen von Python und Spark aufgeführt.

Nächste Schritte

Transformationen verstehen AWS Glue

Für eine effizientere Datenverarbeitung AWS Glue enthält es integrierte Transformationsfunktionen. Die Funktionen werden in einer Datenstruktur namens a, einer Erweiterung von Apache Spark SQL DynamicFrame, von Transformation zu Transformation übergeben DataFrame. A DynamicFrame ist ähnlich wie a DataFrame, außer dass jeder Datensatz sich selbst beschreibt, sodass zunächst kein Schema erforderlich ist.

Um sich mit verschiedenen AWS Glue PySpark integrierten Funktionen vertraut zu machen, lesen Sie den Blogbeitrag Lokales Erstellen einer AWS Glue ETL-Pipeline ohne. AWS-Konto

Erstellen Sie Ihren ersten ETL-Job

Wenn Sie noch keinen ETL-Job geschrieben haben, können Sie zunächst die Drei AWS Glue ETL-Auftragstypen für die Konvertierung von Daten in das Apache Parquet-Muster verwenden.

Wenn Sie Erfahrung mit dem Schreiben von ETL-Jobs haben, können Sie sich AWS Glue GitHub anhand der Beispiele eingehender damit befassen.

Preisgestaltung

Preisinformationen finden Sie unter AWS Glue Preise. Sie können den auch verwenden AWS -Preisrechner, um Ihre monatlichen Kosten für die Verwendung verschiedener AWS Glue Komponenten zu schätzen.

Weitere Ressourcen

Referenzen

- AWS Glue Entwicklerhandbuch
- AWS Glue DataBrew Entwicklerhandbuch
- AWS Glue -API
- AWS Glue ETL streamen
- AWS Glue Studio
- Python-Shell-Jobs in AWS Glue
- Apache Spark-Jobs
- AWS Glue PySparktransformiert Referenz
- Datenkatalog und Crawler in AWS Glue
- AWS Glue Schemaregistrierung
- Anmeldung und Überwachung AWS Glue
- AWS Glue -Versionen

Blog-Posts

- Entwickeln und testen Sie AWS Glue Jobs lokal mit einem Docker-Container
- Optimieren Sie die Speicherverwaltung in AWS Glue
- Bewährte Methoden zur Skalierung von Apache Spark-Jobs und zur Partitionierung von Daten mit AWS Glue
- Lokaler Aufbau einer AWS Glue ETL-Pipeline ohne AWS-Konto
- Arbeiten Sie mit partitionierten Daten in AWS Glue

Beispiele

- AWS Glue DataBrew Beispielprojekt
- Erste Schritte mit AWS Glue interaktiven Sitzungen
- AWS Glue Repository mit ETL-Codebeispielen

Muster

- Erstellen Sie eine ETL-Servicepipeline zum schrittweisen Laden von Daten von Amazon S3 nach Amazon Redshift mithilfe von AWS Glue
- Bereitstellen und verwalten Sie einen serverlosen Data Lake auf dem, AWS Cloud indem Sie Infrastruktur als Code verwenden
- Drei AWS Glue ETL-Auftragstypen für die Konvertierung von Daten in Apache Parquet

Dokumentverlauf

In der folgenden Tabelle werden wichtige Änderungen in diesem Leitfaden beschrieben. Um Benachrichtigungen über zukünftige Aktualisierungen zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Aktualisiert	Wir haben dem AWS Glue ETL-Abschnitt Informationen über Worker-Typen hinzugefügt.	13. März 2024
Aktualisiert	Der Inhalt des Handbuchs wurde im gesamten Handbuch aktualisiert.	15. März 2023
Eine bewährte Methode wurde hinzugefügt	Wir haben Informationen zur Verwendung der Partitionierung zur Abfrage bestimmter Daten hinzugefügt.	4. Februar 2021
Erste Veröffentlichung	—	29. Januar 2021

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.