AWS re:Invent 2017 ダイジェスト

Analytics Services Update

Makoto Shimura, Data Science Solution Architect
Amazon Web Services Japan, K. K.

2018.02.06

志村 誠 (Makoto Shimura)

所属:
アマゾンウェブサービスジャパン株式会社

業務:
ソリューションアーキテクト
(データサイエンス領域)

経歴:
Hadoopログ解析基盤の開発
データ分析
データマネジメントや組織のデータ活用

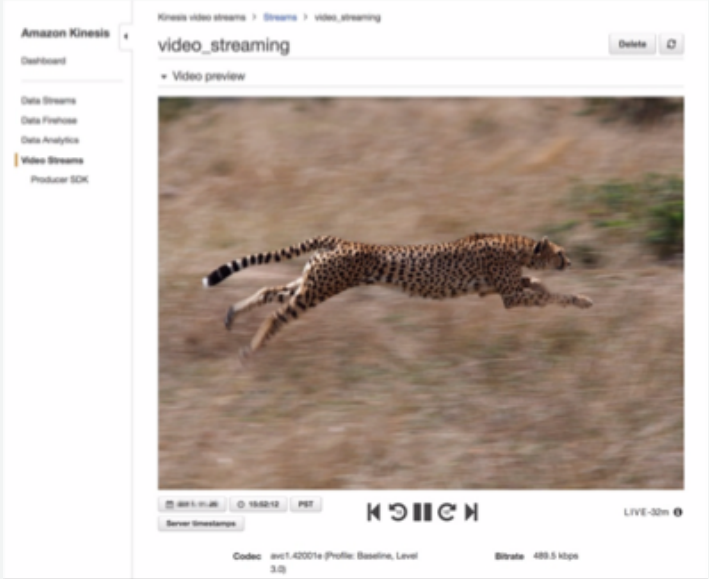
Amazon Kinesis Video

数多くの動画ストリームのインジェストを実現する Amazon Kinesis Video Streams の提供開始

- 大量のカメラ（的な）デバイスからアップロードされる，動画ストリームや時系列データを容易に取り扱うことができるマネージドサービス
- デバイス側は Producer SDK を利用して，Kinesis Video Streams にデータを送信し，Consumer で取得して処理する
- バージニア北部，オレゴン，アイルランド，フランクフルト，東京リージョンで利用可能

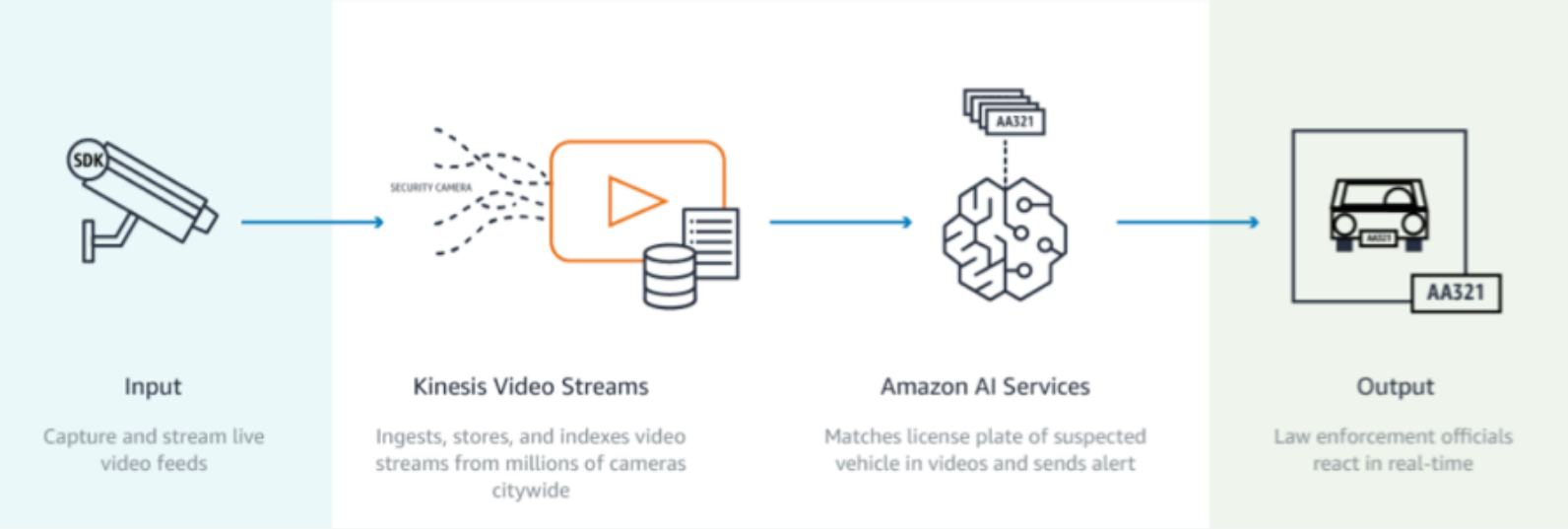
Kinesis Video Streams の流れ

ストリームとして動画を取得し，S3 に保存．ダッシュボードで確認したり，コンシューマで処理を行うことが可能



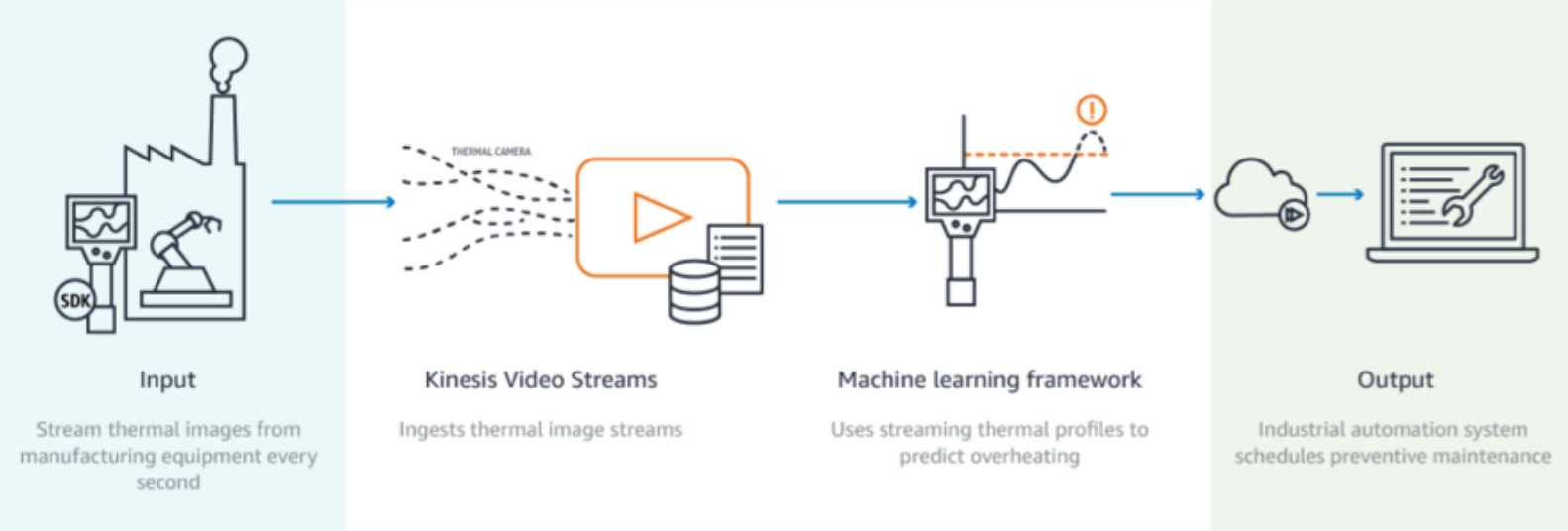
ユースケース: スマートシティ

街中の監視カメラを取得し，Rekognition と連携することで，車のナンバープレートをインデックスとして集約



ユースケース: 工場の自動化

工場に設置した赤外線カメラや深度センサーなどのストリームを取得し，機械学習モデルを適用して，調整を自動化



Amazon Kinesis Data

Amazon Kinesis Data Firehose の配信先として Splunk が一般利用可能に

- Firehose の配信先として Splunk を新たにサポート
  - Splunk Enterprise と Splunk Cloud を含め、HTTP Event Collector (HEC) の Splunk バージョンをサポート
- Splunk は、マシンデータをリアルタイムに解析するためのソフトウェア
  - ログ分析、セキュリティ、コンプライアンス、ビジネスインテリジェンスなどの用途で使用されている
- 以下の設定をすることで、Splunkへの連携が可能
  - Splunk側でHTTP Event Collector (HEC) を作成して、トークンを生成
  - Firehose側でHECの接続エンドポイントとトークンを指定して、配信ストリームを作成

HEC作成
（Splunk側）

配信ストリーム作成
（Firehose側）

Firehoseに
データを送信

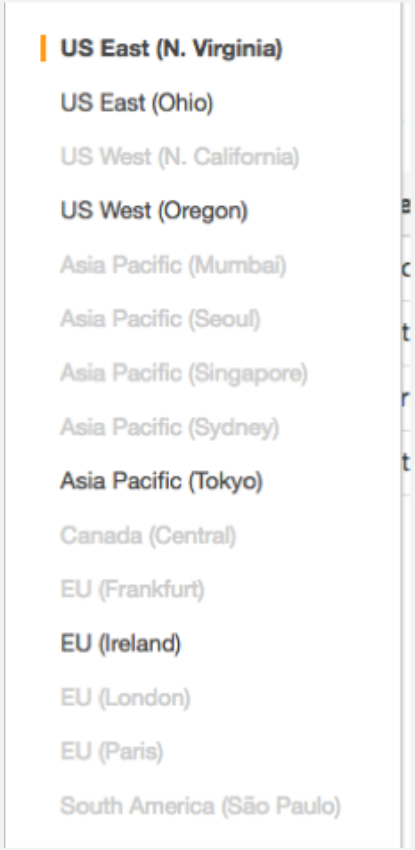
Splunkで解析

aws

AWS Glue

Glue が東京リージョンで利用可能に

- バージニア北部，オハイオ，オレゴン，アイルランドに加えて，東京でも利用可能に
- すでに東京リージョンにきている Kinesis Data Firehose や　Athena と連携して，サーバーレスのデータ変換・分析パイプラインを構築することが可能に



Glue が Scala をサポート

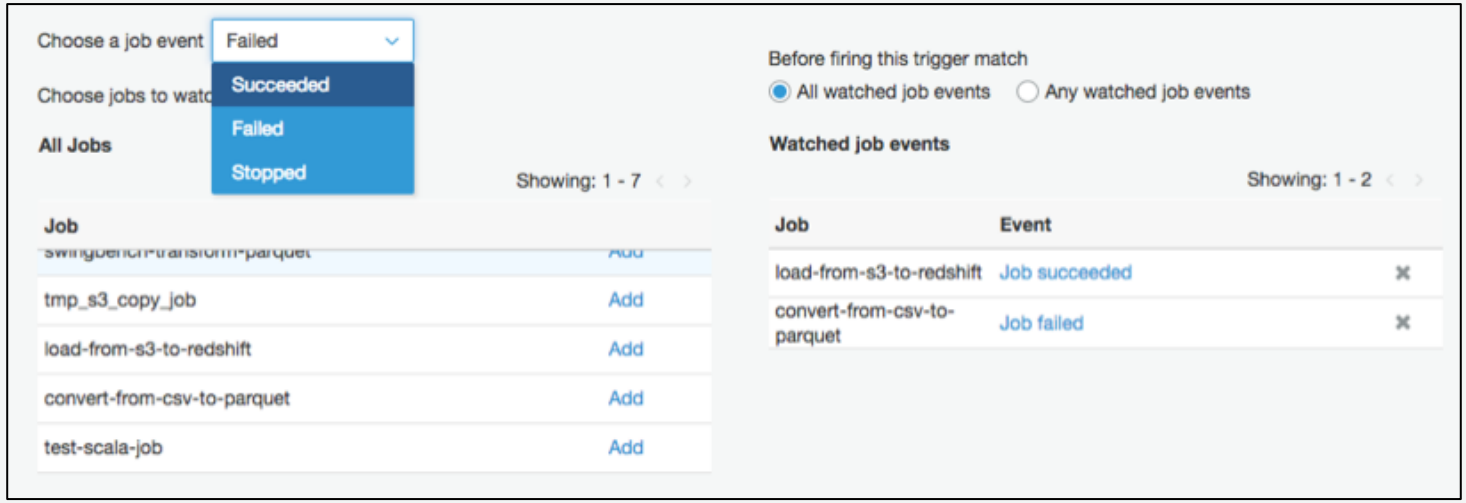
- これまで Python スクリプトのみが使用可能だったが，Scala もサポート
- 基本的な使い方は Python と変わらず，Scala スクリプトを指定してジョブを実行する形
- Scala で書かれた既存の ETL スクリプトの資産を，そのまま Glue で利用可能に

```
import com.amazonaws.services.glue.types.StringNode
import com.amazonaws.services.glue.util.JsonOptions
import com.amazonaws.services.glue.{DynamicRecord, GlueContext}
import org.apache.spark.SparkContext

object DataCleaningLambda {
  def main(args: Array[String]): Unit = {
    val sc: SparkContext = new SparkContext()
    val glueContext: GlueContext = new GlueContext(sc)

    // Data Catalog: database and table name
    val dbName = "payments"
    val tblName = "medicare"
```

aws

新たな Job トリガーをサポート

- 従来，ジョブ実行トリガーとして，前のジョブが成功したとき（succeeded）のみ，次のジョブを実行することが可能だった
- 新たに failed, stopped のステータスも追加され，より柔軟なジョブ実行のパイプラインを作成可能に



Amazon EMR

Kerberos 認証と詳細な EMRFS 許可の有効化サポート

Amazon S3 アクセス用の Kerberos を使用した認証と詳細に設定された EMRFS 許可の有効化が可能に

- Kerberos 認証

  クラスターで実行されているサービス間のリクエスト，クラスターのユーザーアクション，およびリモートサービスからの外部クライアントのリクエストが Kerberos 認証可能に

- 詳細な EMRFS 認可

  ユーザーまたはグループが EMRFS を使用して Amazon S3 にアクセスしたときに，これを引き受ける IAM ロールを指定可能

  - マルチユーザーの Amazon EMR クラスターで S3 のアクセスコントロールを有効化可能
  - IAM ロールを S3 バケット別に指定して使用できるため，クロスアカウント S3 アクセスの有効化が容易に

EMR のリリース 5.9.0 - 5.11.0

- Amazon SageMaker インテグレーション
- MXNet 0.12.0 を利用可能
- I3/P2/P3 インスタンスサポート
- 新しいソフトウェアバージョン
    - Apache Spark 2.2.1
    - Apache Hive 2.3.2
    - Apache Livy 0.4.0
    - Apache Pig 0.17.0
    - Apache Hue 4.0.1
    - Apache Flink 1.3.2
    - Presto 0.184
- Presto の Glue Data Catalog サポート

Amazon Redshift

Amazon Redshiftに新世代のDC2ノードが追加

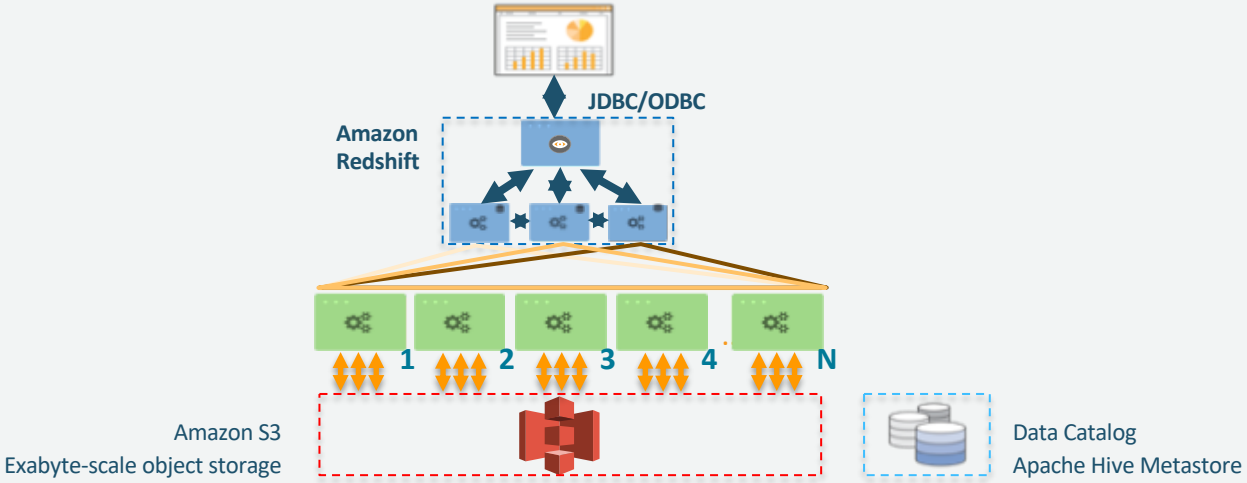
- DS1（旧DW1)の後継としてDS2がリリース
- 高いスループットと低いレイテンシを必要とするDWHワークロードの用途に最適
- DC1との同一価格構成での比較で最大2倍のパフォーマンス
- DC2.8xlargeノードではスライスあたりで2倍のメモリを搭載しており，ストレージレイアウトの改善によって30%多いデータが保管できるように

Dense Compute Node Types

Node Size	vCPU	ECU	RAM (GiB)	Slices Per Node	Storage Per Node
dc1.large	2	7	15	2	160 GB SSD
dc1.8xlarge	32	104	244	32	2.56 TB SSD
dc2.large	2	7	15.25	2	160 GB NVMe-SSD
dc2.8xlarge	32	99	244	16	2.56 TB NVMe-SSD

Amazon Redshift Spectrumが東京リージョンで利用可能に

- Amazon Redshift Spectrum は，データをAmazon S3に置いたままロードせずにAmazon Redshiftからクエリすることが可能に
- Amazon Redshiftが処理可能なデータサイズをペタバイトから，エクサバイト級に押し上げる



動的にキューを切り替える事でパフォーマンス向上

- WLMキューの中でクエリ実行時間の長いものを，クエリの**再実行なしに**他のキューに動的に移動させることで，トータルでのクエリ実行時間を改善
- STL_WLM_RULE_ACTIONシステムテーブルを参照することで，どのクエリがキューが切り替えられたかを確認可能
    - actionカラムにhop(reassign)が記録される
- その他にもQuery Monitoringを利用することで細かくabort/hop/logなどのアクションを可能 (10秒ごとにキュー毎に監視を行っている)
    - メトリクス例: Scan row count, Query CPU time, I/O skew, Nested loop join row count, Spectrum scan row count

Short Query Acceleration (SQA)

- ETL処理の様な実行時間の長いクエリの後に短時間で終わるはずの，ショートクエリがキューイングされると，ショートクエリの実行完了までの時間が想定よりも書かてってしまうことがあった
- **機械学習を使い，ショートクエリを判断することで，指定した時間以内で実行完了出来ると判断すると，**自動的にショートクエリ用のWLMキューに移動する
    - インタラクティブクエリやダッシュボードクエリなどの実行時間の短縮が期待できる．ワークロード依存ではあるが、内部のテストで3倍高速になったケースもある(CTASやSELECTクエリ)
- WLMの設定でSQAを有効にしmaximum run timeをキュー毎に設定 (default 5秒, 1-20秒で設定可能)
    - **Tips: ユーザが設定したキューの合計でコンカレンシーや slot count は15以下にすることが推薦、キュー数が少ないほど SQA に効果的**

aws

Result Caching

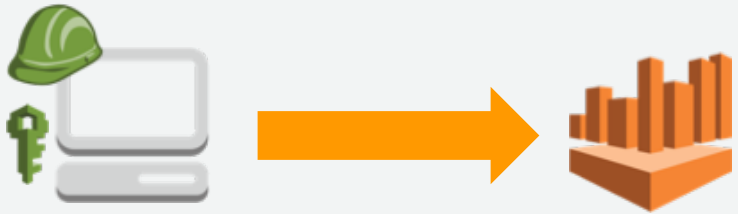
- BIツールやダッシュボードなどから発行される定型的なクエリをキャッシュする事によってクエリのレスポンス速度を向上
- クエリキャッシュにヒットした場合は結果セットはキャッシュから返却されるため実際にクエリの実行は行われない
    - 実際に実行が必要なクエリにリソースを割くことが出来る
- キャッシュにヒットさせるためにはデータの変更が行われておらずクエリも一致する必要がある。そうでない場合はクエリを実行し結果をキャッシュする
    - **enable_result_cache_for_session** パラメーターで無効化可能
    - **SVL_QLOGテーブル** に **source_query カラム**が追加され、キャッシュを実行したクエリIDを識別可能に

aws

Amazon Athena

ODBC ドライバ対応

- AthenaにアクセスするAPIやWeb Console，JDBC ドライバの他にODBC ドライバに対応
    - ODBCドライバ: Windows(32,64bit)/Linux(32,64bit)/OSX
- より多くのアプリケーション環境から Athena にアクセス可能に



Geospatial データへのクエリができるように

- データフォーマットとして，WKT (Well-Known Text) と WKB (Well-Known Binary) をサポート
- point, line, multiline, polygon, multipolygon といった特別な地理データを扱うことが可能に
- 中身は ESRI Java Geometry Library を用いた Presto プラグイン

```
SELECT counties.name,
        COUNT(*) cnt
FROM counties
CROSS JOIN earthquakes
WHERE ST_CONTAINS (counties.boundaryshape, ST_POINT(earthquakes.longitude, earthquakes.latitude))
GROUP BY  counties.name
ORDER BY  cnt DESC
```

Athena の Presto エンジンアップグレード

- 使用している Presto のバージョンが 0.172 に
- このアップグレードにより，以下の機能が使用可能に
  - Lambda 式
  - ヘッダ行のスキップ（テーブルプロパティにおける skip.header.line.count の指定）

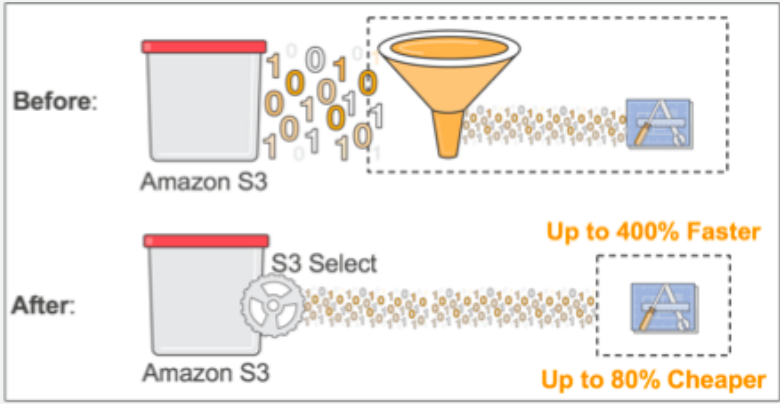
```
SELECT filter(ARRAY [], x -> true); -- []
SELECT filter(ARRAY [5, -6, NULL, 7], x -> x > 0); -- [5, 7]
SELECT filter(ARRAY [5, NULL, 7, NULL], x -> x IS NOT NULL); -- [5, 7]
```

Amazon S3

S3/Glacier Select

- S3やGlacier内に保存されているobjectに対して直接SQLを使ってクエリを実行可能
- S3 Selectはpreview
- オブジェクト全体を取得するのではなく必要なデータのみを取得するクエリを利用可能
- CSV, JSON, GZiP
- S3 SelectのPrestoコネクタも利用可能. クエリを変更すること無くPrestoクエリに必要なデータのみを取得になるため高速化が期待できる

```
Python er = PrintingResponseHandler()
s3 = boto3.client('s3')
response = s3.select_object_content(
    Bucket="super-secret-reinvent-stuff",
    Key="stuff.csv",
    SelectRequest={
        'ExpressionType': 'SQL',
     'Expression': 'SELECT s._1 FROM S3Object AS s'',
        'InputSerialization': {
            'CompressionType': 'NONE',
            'CSV': {
                'FileHeaderInfo': 'IGNORE',
                'RecordDelimiter': '\n',
                'FieldDelimiter': ',',
            }
        },
        'OutputSerialization': {
            'RecordOrder': 'STRICT',
            'CSV': {
                'RecordDelimiter': '\n',
```

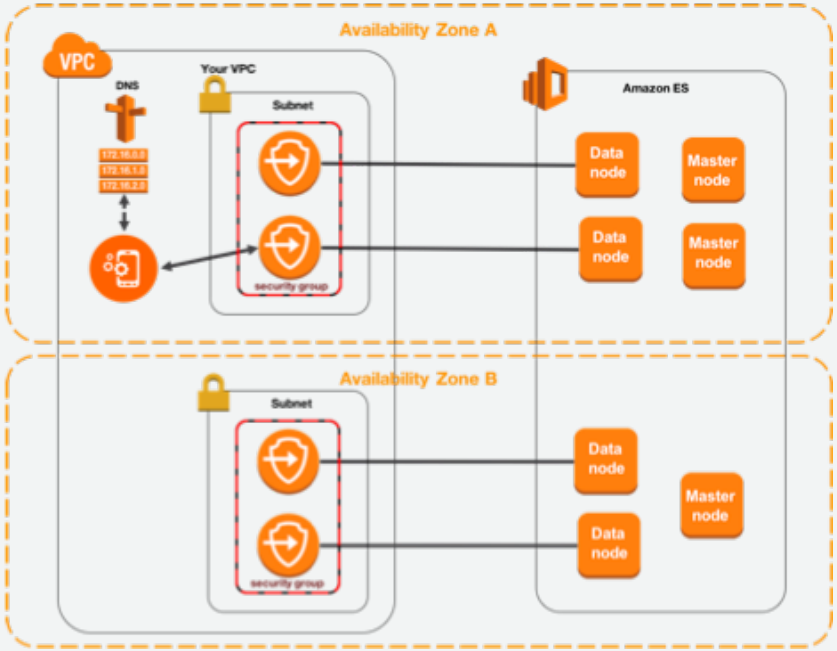


Amazon Elasticsearch Service

Amazon Elasticsearch Serviceで I3インスタンスを使用してPBスケールのクラスター

- I3インスタンスのサポートを開始
- 大規模なログ分析ワークロード用に、単一のElasticsearchクラスタに最大1.5ペタバイトのデータを格納できるようになった
- I3インスタンスを使用すると
    - 各ノードに**最大15テラバイト**のデータを格納でき、**前世代のI2インスタンスのコストの50%未満**で、**インデックス処理のスループットを最大3倍向上**できる
    - **パフォーマンスの向上**（最大330万IOPSおよび16GB/秒のシーケンシャルディスクスループット）により、他のインスタンスタイプと比較して、より要求の厳しいワークロードをサポートし、取り込み率を向上させ、クエリ応答時間を短縮できる
    - **ネットワークの拡張**により、シャードの再配置が高速になり、クラスタの更新とインスタンスのリカバリ時間が短縮される
- Amazon ESのI3インスタンスでの暗号化機能を有効にすることもできる

Amazon Elasticsearch ServiceがVPCをサポート



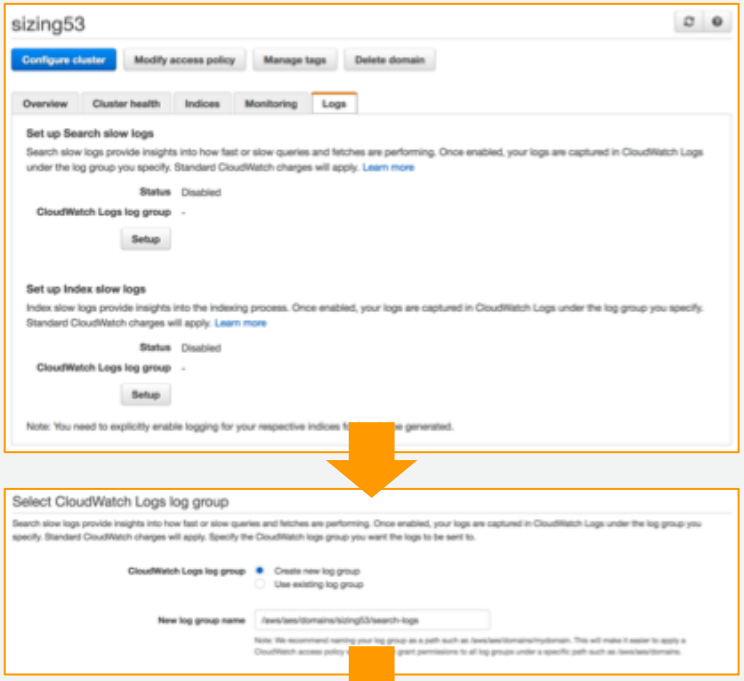
- 各ドメインごとにファイアウォールルールとIPベースのアクセスポリシーを設定して維持することなく，VPCからAmazon ESへのアクセスを簡単に設定できる
- この新機能により，パブリックインターネットを経由することなく、Amazon VPCとAmazon ES間のすべてのトラフィックをAWSネットワーク内に維持することが可能に
- 既存のVPCのSGを使用してアクセスを制御できる

Amazon Elasticsearch Serviceがストレージの暗号化をサポート

- 暗号化を有効にした Amazon ES ドメインでは、プライマリおよびレプリカの Index、ログファイル、メモリスワップファイル、自動スナップショットなど、基礎となるファイルシステムに格納されているすべてのデータが暗号化される
- ドメインの作成時に、暗号化を有効にすることができる
- **EBS と EC2 インスタンスストアの両方**をサポート
- Amazon ES から KMS マスターキーを作成することも、独自のキーを選択することも、できる

- Amazon ES で、この暗号化を有効にしても、以下は対象とならない（**代替手段あり**）
    - 手動スナップショット：S3 管理キーを使用したサーバーサイド暗号化を使用して、スナップショットリポジトリとして使用するバケットを暗号化できる
    - Slow logs : Amazon ES ドメインと同じ KMS 暗号化キーを使用して、CloudWatch Logs ロググループを暗号化できる

Slow logsを使用してAmazon Elasticsearch Serviceドメインを最適化



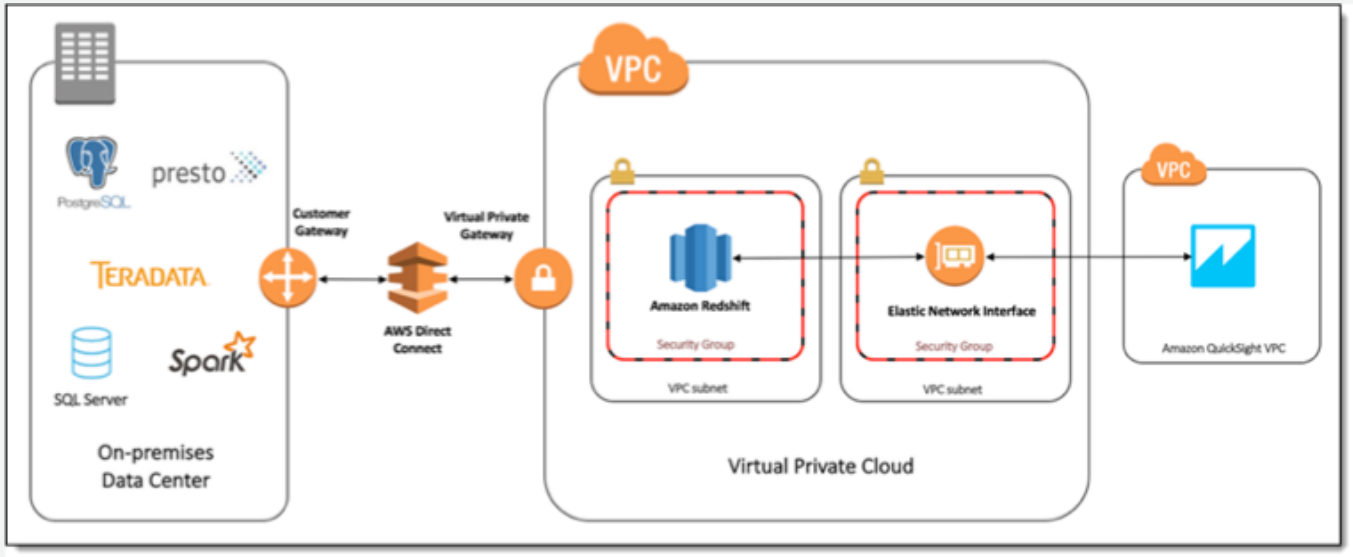
- Slow logsはCloudWatch Logsに公開され、自由にオン/オフ可能
- CloudWatch 料金の支払いのみが必要で、追加のAmazon ES料金はかからない
- ドメインごとに個別に，索引付けおよび検索操作の Slow logs を有効にすることができる
- 検索操作は，QueryとFetchのフェーズそれぞれでSlow Logsを取得可能
- ロギングレベル: trace, debug, info, warn

```
curl -XPUT http://<your domain's endpoint>/index/_settings -d '{"index.search.slowlog.threshold.query.<level>":"10s"}'
```

Amazon QuickSight

プライベート VPC アクセスのプレビュー開始

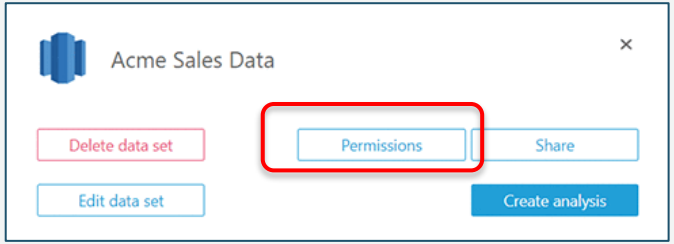
- QuickSight から ENI 経由で，VPC のプライベートサブネット内にあるリソースへのアクセスが可能に
- VPC と DX で接続することで，オンプレミスのデータソースにも接続できるように



Amazon QuickSightで
行レベルセキュリティ（RLS）が利用可能に

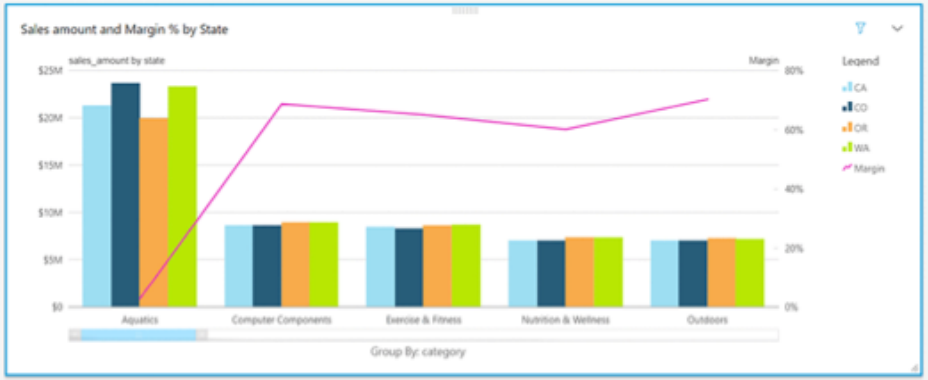
- Amazon QuickSightにRow Level Security (RLS)機能が追加
- 同じダッシュボードでもユーザによって見える範囲を変える事が可能
- 事前にルール定義用のルールデータセットを作成し、Visual用のデータセットのPermissionに登録する
  - 必ずUsernameという列を作成し、そこにユーザ名を設定
  - 右の例だとSusanはState=CAの行のみ閲覧できる
- **Enterprise Editionのみ**利用可能

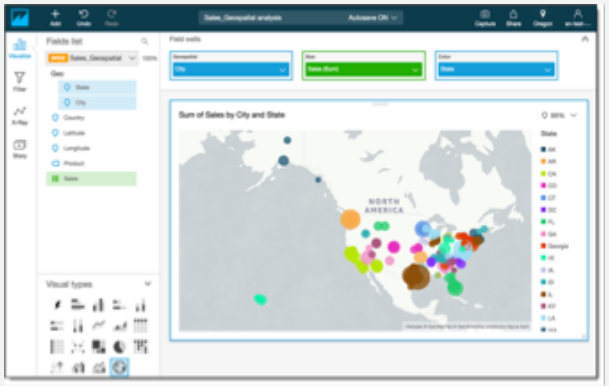
Username	Category	State
Jane	Aquatics, Exercise & Fitness, Outdoors	WA, OR
Susan		CA



各種グラフ関連機能の強化

- コンボチャートのサポート
- 地理データの可視化機能追加
- Pivot テーブル以外に，フラットテーブルの描画機能をサポート
- 1000 カラムを超える横長のデータソースをサポート






aws