Amazon Kinesis Data Streams for DynamoDB の検証

- やりたいこと
  - DynamoDB の変更ログを Firehose に流したい
  - Amazon Kinesis Data Streams for DynamoDB はどんな感じかを把握する

背景と Summary

背景

DynamoDB の変更ログを Firehose に流す方法は，シンプルな構成では以下の 2 つが考えられる.

- **DynamoDB -> DynamoDB Stream -> Lambda -> Data Firehose**
  - これまで構築してきたリソース構成
  - Lambda で AWS-SDK を使用して Firehose へデータを転送する実装を書く必要がある
- **DynamoDB -> Kinesis Data Stream -> Data Firehose**
  - 今回調べたリソース構成

これら 2 つの比較についてまとめた.

Summary

- 今回調べた「**DynamoDB -> Kinesis Data Stream -> Data Firehose**」というパイプラインの方が楽にログ基盤の構築ができる

以下，メリット（楽だと感じた点）と注意点.

- メリット
  - Kinesis Data Stream を利用すれば，マネージドサービスを（CFn 等で）繋げるだけで，パイプラインの構築ができる．つまり，**リソースの構成がシンプルになる**.
    - DynamoDB Stream を用いたパイプラインは，Lambda の中で SDK を使用して Firehose へデータを流し込んでいる．つまり，SDK でサービス間を繋げる部分を自前で実装する必要がある.
    - 一方，Amazon Kinesis Data Streams for DynamoDB を用いた場合，DynamoDB のテーブルのコンソール上で Kinesis Data Stream を設定するだけで DynamoDB と Data Firehose を繋ぐことができる.
- 注意点
  - **ストリームレコードの整形（DynamoDB の更新イベントのデシリアライズ等）はどちらの場合も必要になる**
    - DynamoDB Stream を用いた場合，Lambda 内でストリームレコードの整形，Firehose への流し込みを行っている.
    - Amazon Kinesis Data Streams for DynamoDB を用いた場合，ストリームレコードの整形に Data Firehose の Transformation（Lambda）を利用する必要がある．ただ，Data Firehose のオプションとしての Transformation であるため，リソースの構成は複雑にならずに済む.
- まだ不明な点
  - データ転送の効率
  - 料金（Kinesis 自体は安くない？）

DynamoDB の変更ログを Firehose に流す方法

DynamoDB の変更ログを Firehose に流す方法は主に以下の 2 つ.

- **DynamoDB -> DynamoDB Stream -> Lambda -> Data Firehose**
  - これまで構築してきたリソース構成
  - シャード数の変更には制限がある（1/2 倍 or 2 倍）
- **DynamoDB -> Kinesis Data Stream -> Data Firehose**
  - 今回調べたリソース構成
  - プロデューサー：DynamoDB
  - コンシューマー：Data Firehose

DynamoDB -> DynamoDB Stream -> Lambda -> Data Firehose

省略.

DynamoDB -> Kinesis Data Stream -> Data Firehose

以下を参照した.

- 公式 doc
  - Change Data Capture for Kinesis Data Streams
  - Writing to Kinesis Data Firehose Using Kinesis Data Streams
- 例
  - classmethod：[Kinesis Data Streams] ストリームを挟んで S3 に一覧ログを記録してみました
  - CyberAgent：Kinesis と Lambda で作る Serverless なログ基盤
    - Kinesis Data Stream を活用したサーバーレスなログ基盤の構築事例

Kinesis Data Stream をデータソースとして使用する Kinesis Data Firehose 配信ストリームを作成する. これにより, Kinesis Data Firehose を使用して, 既存のデータストリームから簡単にデータを読み取り, 目的のストレージサービスにストリーミングデータをロードすることができる.

Kinesis データストリームをソースとして使用するには、Kinesis ストリームリストで既存のストリームを選択するか, 新規作成 を選択して新しい Kinesis データ ストリームを作成します。新しいストリームを作成した後、[更新] を選択して Kinesis ストリーム リストを更新します。ストリームの数が多い場合は、名前によるフィルタを使用してリストをフィルタリングします。

- 注意事項
  - **Kinesis Data Stream -> Kinesis Data Firehose Delivery Stream**
  - Kinesis Data Stream を Kinesis Data Firehose 配信ストリームのデータソースとして設定すると, Kinesis Data Firehose `PutRecord` および `PutRecordBatch` 操作が無効になる. この場合, Kinesis Data Firehose 配信ストリームにデータを追加するには, Kinesis Data Streams の `PutRecord` および `PutRecords` 操作を使用する.

Kinesis Data Firehose は、Kinesis ストリームの LATEST 位置からデータの読み込みを開始します。Kinesis データストリームの位置の詳細については、GetShardIterator を参照してください。Kinesis Data Firehose は、Kinesis Data Streams GetRecords 操作を各シャードごとに 1 秒に 1 回呼び出します。

複数の Kinesis Data Firehose 配信ストリームは、同じ Kinesis ストリームから読み込むことができます。他の Kinesis アプリケーション（コンシューマー）も同じストリームから読み取ることができます。Kinesis Data Firehose 配信ストリームまたは他のコンシューマアプリケーションからの各呼び出しは、シャードの全体的なスロットル制限に対してカウントされます。スロットリングを回避するには、アプリケーションを慎重に計画してください。Kinesis Data Streams の制限の詳細については、Amazon Kinesis Streams Limits を参照してください。

補足

"Amazon Kinesis Data Streams" vs "Amazon Kinesis Data Firehose"

どちらも，特定のサービスから別のサービスへデータを高速に転送するためのサービス．メッセージキューイングのめっちゃスケールする版と考えれば良い．

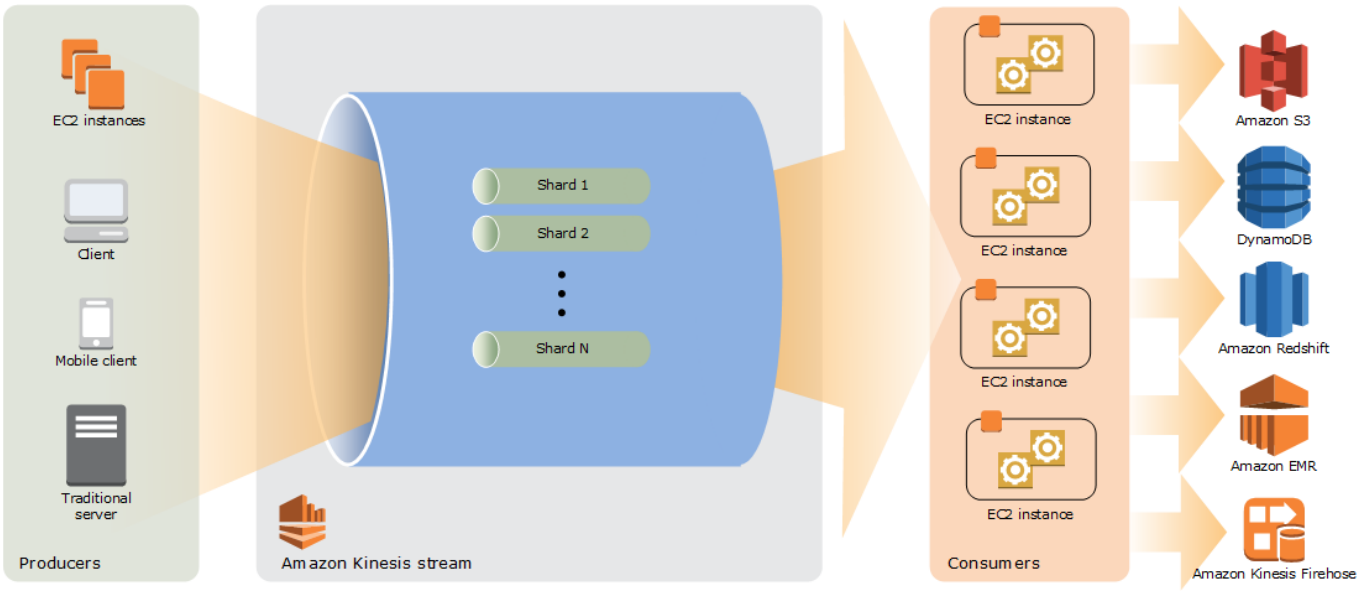
- Kinesis Data Stream
  - プロデューサー，コンシューマーをユーザが任意に設定できる
  - スループット（シャード）は自分で決める
- Kinesis Data Firehose
  - フルマネージド（オートスケールを含む）．設定が非常に楽．
  - コンシューマーは RedShift，S3，ElasticSearch など固定されてる．分析基盤用途の意味合いが強い．

Amazon Kinesis Data Streams 用語と概念

- 公式 doc：Amazon Kinesis Data Streams 用語と概念

アーキテクチャ

以下の図に，Kinesis Data Streams のアーキテクチャの概要を示す． プロデューサーは継続的にデータを Kinesis Data Streams にプッシュし， コンシューマーはリアルタイムでデータを処理する． コンシューマー（Amazon EC2 上で実行されるカスタムアプリケーションや Amazon Kinesis Data Firehose 配信ストリームなど）は，Amazon DynamoDB，Amazon Redshift，Amazon S3 などの AWS のサービスを使用して，その結果を保存できる．



用語

- Kinesis Data Stream，ストリーム
- データレコード
- Shard，シャード
    - ストリーム内の一意に識別されたデータレコードのシーケンス．ストリームは複数のシャードで構成され，各シャードが容量の 1 単位になる．
    - 各シャードは読み取りは最大 1 秒あたり 5 件のトランザクション，データ読み取りの最大合計レートは 1 秒あたり 2 MB と 書き込みについては最大 1 秒あたり 1,000 レコード，データの最大書き込み合計レートは 1 秒あたり 1 MB (パーティションキーを含む) をサポートできる．
    - ストリームのデータ容量は，ストリームに指定したシャードの数によって決まります。ストリームの総容量はシャードの容量の合計です。
    - データ転送速度が増加した場合、ストリームに割り当てられたシャード数を増やしたり、減らしたりできます。詳細については、ストリームをリシャーディングする を参照してください。
- Producer，プロデューサー
    - レコードを Amazon Kinesis Data Streams に送信するもの．例えば，ストリームにログデータを送信するウェブサーバーはプロデューサーである．
- Consumer，コンシューマー
    - Amazon Kinesis Data Streams からレコードを取得して処理するもの．これらのコンシューマーは Amazon Kinesis Data Streams Application と呼ばれる．
    - e.g. EC2 instance
- Amazon Kinesis Data Streams application
    - ストリームのコンシューマーで，一般的に EC2 インスタンスのフリートで実行される．
    - Kinesis Data Streams Application の出力を別のストリームの入力にすることで，リアルタイムにデータを処理する複雑なトポロジを作成できる．アプリケーションは，さまざまな他の AWS サービスにデータを送信することもできる．複数のアプリケーションが 1 つのストリームを使用して，各アプリケーションが同時にかつ独立してストリームからデータを消費できる．
- シーケンス番号
    - 各データレコードには，シャード内のパーティションキーごとに一意のシーケンス番号が割り当てられる．
    - `client.putRecords` を使用してストリームに書き込むと，Kinesis Data Streams によってシーケンス番号が割り当てられる．