

【Talk-1】 AWS DynamoDBへの大量データ 高速投入方法について

伊藤忠テクノソリューションズ株式会社
金融開発第4部

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group



P03

自己紹介



P04

AWS DynamoDB

データ投入方法について



P07

Dyneinの紹介



P09

現状と課題

現在公開されているdyneinの使用感と課題について



P10

対応と結果

dyneinの課題に対する対応とその結果について

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

自己紹介

2022年 伊藤忠テクノソリューションズ株式会社入社
金融開発第4部にてクレジットカード決済代行サービスを提供するシステムの維持保守を担当。
担当システムはAWS上に構築されており、ECS、DynamoDB、S3、EventBridge等々を使用

同年に副業として株式会社メモアカに入社
記憶力を鍛えるトレーニングサイト「メモアカ」、脳トレアプリ「カオナマエ」の開発を担当。

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

AWS DynamoDB

AWSが提供する、言わずと知れたサーバーレス NoSQL の完全マネージド型データベース。高い可用性、スケーラビリティ等々の特徴があります。

利用シーンも多いかと思いますが、私たちのチームでは既存サービスから新規システムへの切り替えに伴い大量件数のデータ投入を行う必要がありました。



無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

当初のデータ投入手法

私たちのチームでは当初、JavaでS3に配置されたファイルからインポートを行うバッチ処理を作成し、運用していました。投入データが一番多いテーブルで約300万件を投入する必要があり、実施に4時間45分もかかっていました。

⇒チーム内外からどうにか短縮できないかとの声があがる

DynamoDBへの各種インポート方法

DynamoDBへのデータインポート方法はいくつかの方法があります

①S3からのインポート

⇒テーブルの新規作成であれば可能

②CLIからの投入

⇒非常に時間がかかる

③AWS Data Pipelineの利用

⇒構築が手間

④dyneinの使用

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

Dyneinの紹介

- AWS Labsが提供するDynamoDBのCLIツール
- Rustで書かれており、処理が非常に高速
- import / exportが可能

<https://github.com/awslabs/dynein>

※6/21にリリースされた v0.3.0 でかなり使い勝手がよくなりました

① openssl3.0に対応

→以前まではopenssl-1.1.1をインストールする必要があった

② credentialを設定せずとも利用可能に

→旧バージョンではcredentialを設定しないといけなかったが、

rusotoからaws-sdk-rustに移行されたため、IAM Role等で動くようになりました

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

利用できるコマンドの紹介

- scan

```
dy scan -t <table> --limit 10
```

- export

```
dy export -t <table> -o output.json
```

- import

```
dy import -t <table> -i input.json
```

他AWS CLIと同様にput, update, delete, queryなども使えます

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

現状と課題

- 速度は非常に早い
 - 1万件程度なら10秒程度でimport/export可能
 - ただし、dynamoのスペックをある程度上げておかないとスループットエラーになる
- 大量件数の場合は課題あり
 - 1万件超えたあたりから認証まわりのエラーが出る
 - 以下が実際に出たエラー

```
$ dy import -t <table_name> -i import_data.json
```

```
8350 items processed (1201.56 items/sec)
```

```
Error: BatchWriteError(DispatchFailure(DispatchFailure { source: ConnectorError { kind: Other(None),  
source: CredentialsNotLoaded(CredentialsNotLoaded { source: Some("no providers in chain provided credentials") } ), connection:  
Unknown } } ))
```

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

GitHub上で問い合わせ



StoneDot commented on Jul 22

Contributor ...

Thank you for using dynein and reporting the error. A credential provider's throttling might cause this issue. The current implementation does not reuse clients. I think that the code change on our side should fix this issue.

<https://github.com/awslabs/dynein/blob/main/src/batch.rs#L240>



StoneDot added the **bug** label on Jul 22

エラー発生の流れ

- ①importコマンド実行
- ②importデータを25件ごとに分割(BatchWriteItemの制約)
- ③ AssumeRoleで認証情報取得
- ④ BatchWriteItemで25件ごとに書き込み

②～④を繰り返す中で、③のAssumeRole部分はその負荷に耐え切れなくなりスロットリングが発生

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

コード修正

認証情報を使いまわすよう修正

```

app.rs  batch.rs x  main.rs  control.rs
221 async fn batch_write_item_api(
222     cx: &app::Context,
223     request_items: HashMap<String, Vec<WriteRequest>>,
224 ) -> Result<
225     Option<HashMap<String, Vec<WriteRequest>>>,
226     aws_sdk_dynamodb::error::SdkError<BatchWriteItemError>,
227 > {
228     debug!(
229         "Calling BatchWriteItem API with request_items: {:?}",
230         &request_items
231     );
232
233     let retry_config = cx
234         .retry
235         .as_ref()
236         .map(|v| v.batch_write_item.as_ref().unwrap_or(&v.default));
237     let config = cx
238         .effective_sdk_config_with_retry(retry_config.cloned());
239     .await;
240
241     // cx.ddb_clientを再利用する
242     let ddb = &cx.ddb_client;
243
244     match ddb
  
```

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

修正後実施結果

- ・前回(Java)とDyneinでの比較
⇒300万件の場合、4時間もの時間を短縮

No.	DB	移行先テーブル名	今回		前回	
			ファイル件数	処理時間	ファイル件数	処理時間
7	Dynamo	table1	85,040	0:02:00	289,535	1:42:00
8	Dynamo	table2	80,445	0:02:00	273,344	0:48:00
9	Dynamo	table3	3,023,608	0:44:00	2,964,305	4:45:00



無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

まとめ

- ・Dyneinは余計なリソースを増やさずに手軽に導入可能なCLIツール
- ・数千件程度であれば快適に利用可能
- ・一方で公開されているバージョンでは大量件数投入には課題あり
- ・修正を待つor自身で修正を行えば非常に強力なツール

無限の未来と、幾千のテクノロジーをつなぐ。

CTC Financial Services Group

無限の未来と、
幾千のテクノロジーをつなぐ。

CTC Financial Services Group

