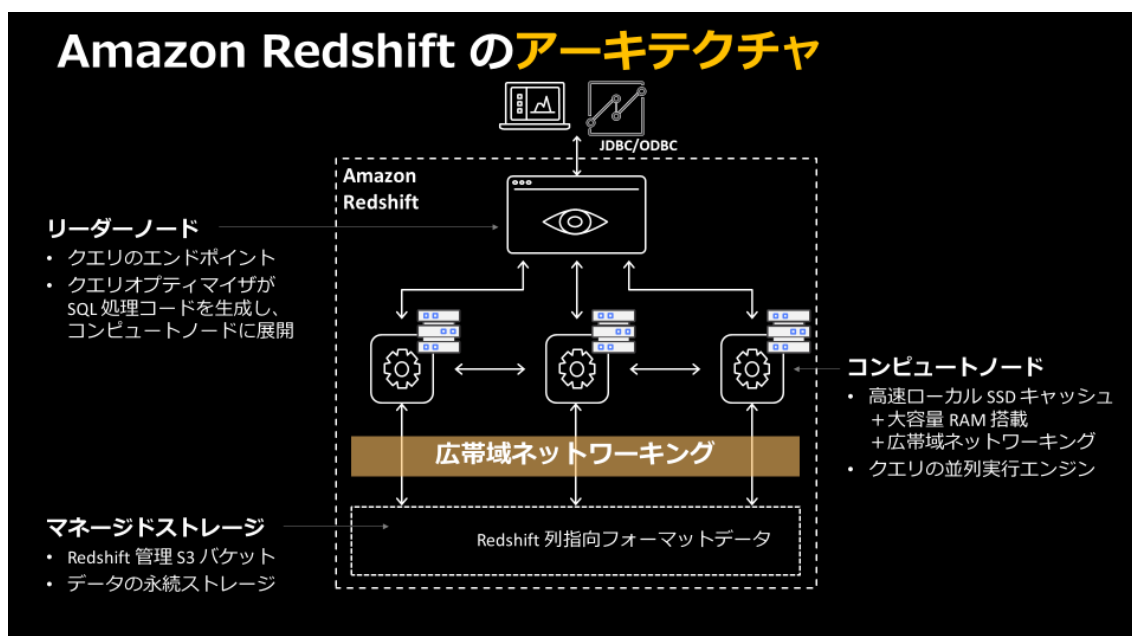


はじめに：

Amazon Redshift Serverless は現在 Preview 中のサービスであり、従来のクラスター型 Redshift と同様に、あらゆる規模で高いパフォーマンスで分析を非常に簡単に実行できる新機能です。使い方はデータを読み込んでクエリを開始するだけで、クラスターのセットアップや管理は不要です。データのクエリやロード中など、データウェアハウスの使用中は秒単位で課金されデータ、ウェアハウスがアイドル状態の場合は課金されないのが特徴です。

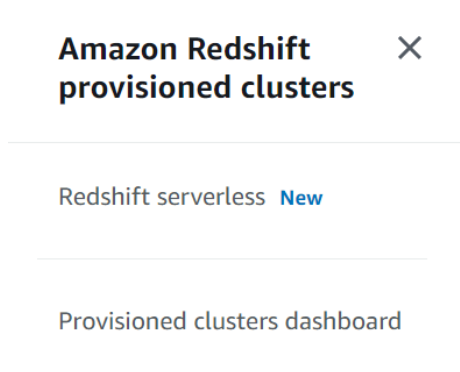
Redshift Serverless では、使用するコンピューティングとストレージに対して別途料金を支払います。コンピューティング性能は Redshift Processing Units (RPU) で測定され、ワークロードの料金は RPU 時間で 1 秒単位で請求されます。ストレージについては、Amazon Redshift が管理するストレージに保存されたデータと、スナップショットに使用したストレージに対して課金されます。これは、RA3 インスタンスを使用してプロビジョニングされたクラスターで支払うのと同じです。



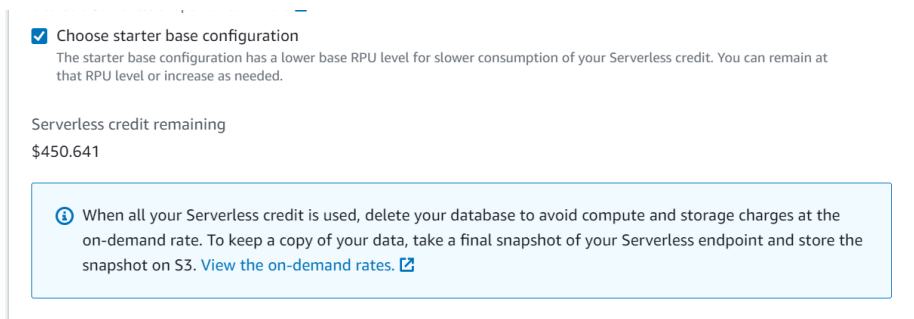
現在 Redshift Serverless は Preview 中であり、仕様は変更になる可能性があります。

1. Redshift マネージメントコンソールに移動します。

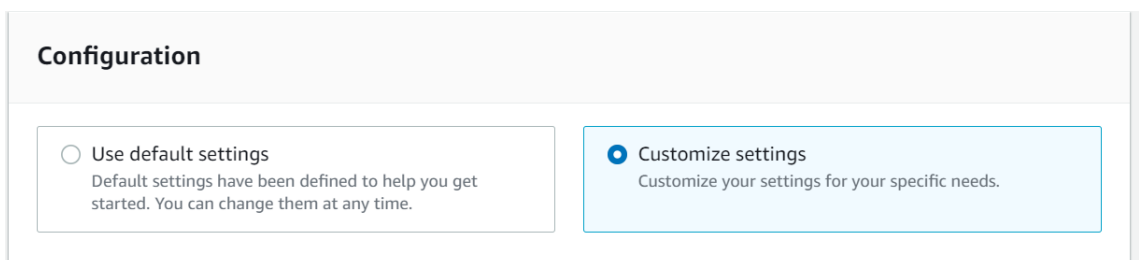
2. 左ペインの[Redshift serverless]をクリックします



3. [Choose starter base configuration]をクリックします。以下の通り検証用 500 \$ クレジットが付与されます。



4. クラスターを起動するための Network は自動で default VPC が設定されています。現在 Redshift Serverless は Public エンドポイントには対応しておらず、VPC 経由でのエンドポイントのみが作成可能です。[Customize Settings]を選びます



5. Admin のパスワードをセットします

Admin user name
The administrator's user name for the first database.

The name must be 1-128 alphanumeric characters, and it can't be a reserved word.

☐ Auto generate password
Amazon Redshift can generate a password for you, or you can specify your own password.

Admin user password
The password of the admin user.

Must be 8-64 characters long. Must contain at least one uppercase letter, one lowercase letter and one number. Can be any printable ASCII character except "/", "'", or "@".

☐ Show password

6. Redshift Serverless は Redshift と異なり IAM のクレデンシャルが不要で、ユーザー/パスワードのみが利用が可能です。このため、IAM を設定せず[Create serverless endpoint]をおします

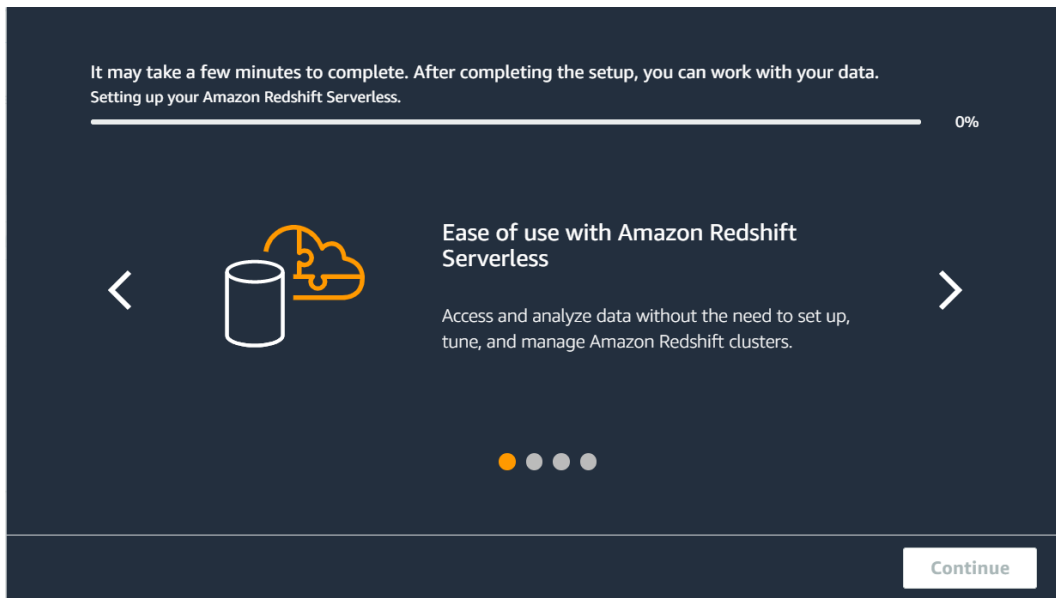
関連付けられた IAM ロール (0) Set default ▼ IAM ロールの管理 ▼

< 1 >

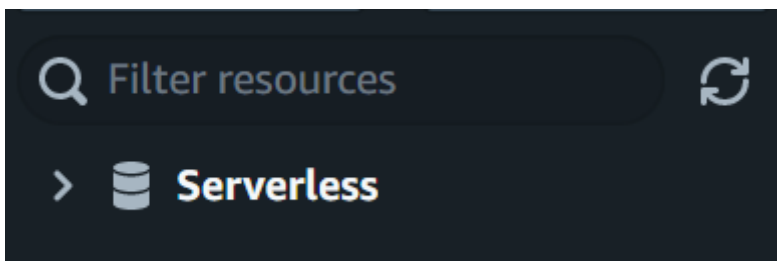
<input type="checkbox"/>	IAM ロール	▼	状態 ▼	Role type ▼
<p>リソースなし</p> <p>関連付けられた IAM ロールがありません</p> <p>IAM ロールを関連付ける</p>				

キャンセル Create serverless endpoint

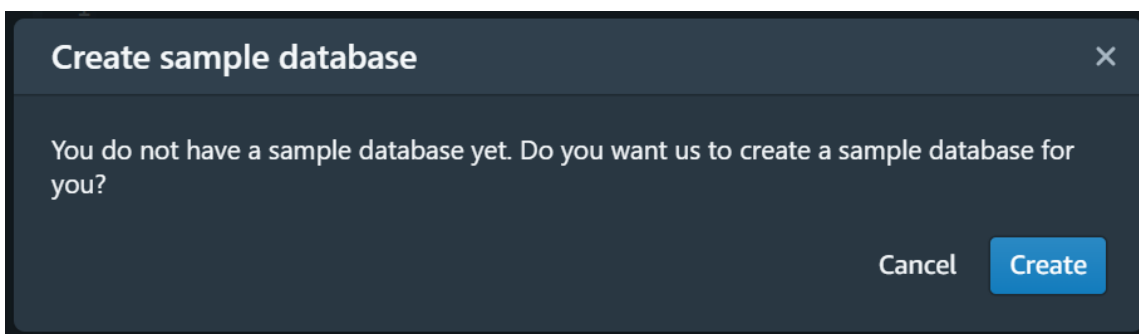
7. しばらくまつと[Continue]ボタンが押せるようになります



8. 起動したら、左ペインから[Query Editor]をクリックします
9. [Serverless]データベースにテストデータをロードします。[>Serverless]の矢印をクリックします



10. 順に、[> sample_data_dev]の矢印、[tickti]のフォルダをクリックするとテストデータがロードされ、サンプルクエリが表示されます。途中以下の確認ダイアログが表示されますので、[Create]をおします



あらかじめ存在している[dev]データベースは、Redshift Serverless の本体であり様々な設定データ、RPU 消費履歴等が格納されています。

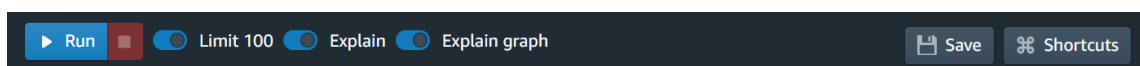
11. [Total Quantity per buyer]を[Run]ボタンをおし実行してみます

```

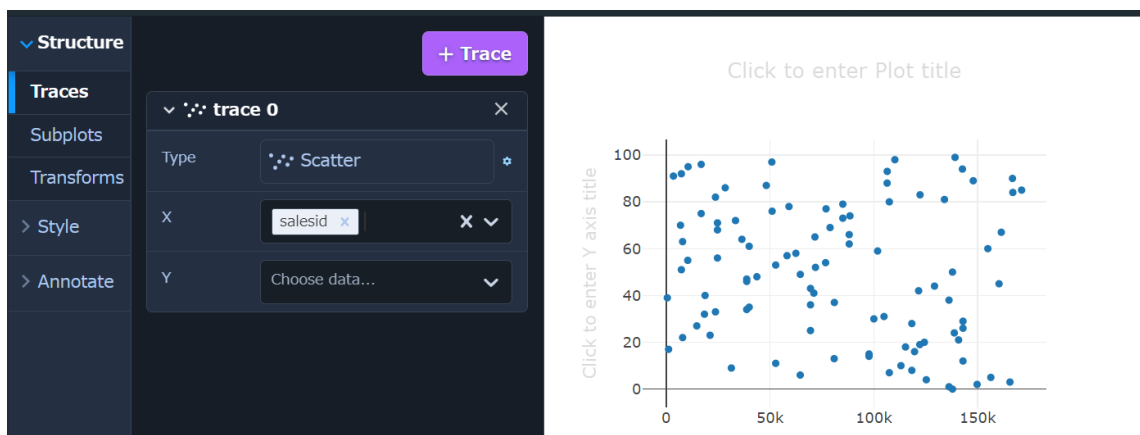
1 SELECT firstname, lastname, total_quantity
2 FROM (SELECT buyerid, sum(qtysold) total_quantity
3       FROM tickit.sales
4       GROUP BY buyerid
5       ORDER BY total_quantity desc limit 10) Q, tickit.users
6 WHERE Q.buyerid = userid
7 ORDER BY Q.total_quantity desc;

```

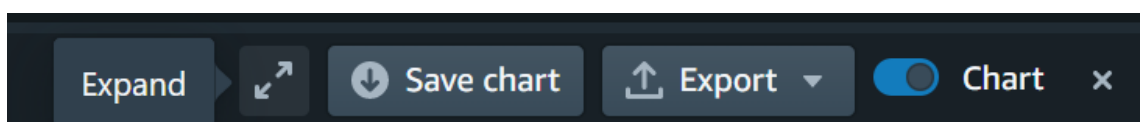
- 1 2. 実行には3桁 ms かかっていることが画面右下からわかります。
- 1 3. もう一度[Run]を押し実行すると、実行速度が2桁 ms になっていることがわかります。これは、Serverless のコールドスタートではないことに注意してください。
Redshift はコンピュータノードに高速な SSD キャッシュを備えています。これによりよく呼び出されるデータがキャッシュ化され処理が高速化します
- 1 4. [Explain]トグルをオンにして再実行すると、実行計画が表示されクエリーチューニングのヒントなどが表示されます



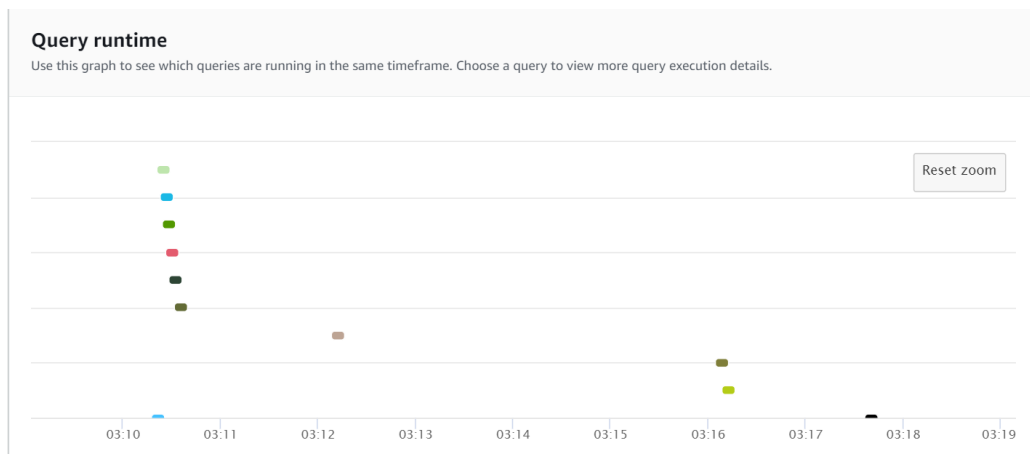
- 1 5. 次に[SELECT * FROM tickit.sales;]を実行した後、[chart]トグルをオンにします
- 1 6. 以下のように様々なテンプレートからグラフを生成しデータ分布を視覚化することが出来るようになっています。



見づらい場合は[Expand]を押してみてください



- 1 7. マネージメントコンソールに戻り、左ペインから[Query and database monitoring]をクリックします
- 1 8. 以下の通りクエリの実行履歴が出力されています。



クエリとロード (12) Any status ▼ End query

Find queries by Query ID or Process ID

<input type="checkbox"/>	Query ID ▼	Status ▼	Start time ▼	Duration ▼	User ID ▼	Database ▼	SQL command ▼	Pro
<input type="checkbox"/>	355	Success	Apr 29th, 2022 03:22:36 AM UTC	2 sec	101	dev	/* Console Query: 814813...	10
<input type="checkbox"/>	279	Success	Apr 29th, 2022 03:17:36 AM UTC	103 ms	101	sample_data_dev	SELECT * FROM ticket.sales...	10
<input type="checkbox"/>	259	Success	Apr 29th, 2022 03:16:08 AM UTC	219 ms	101	sample_data_dev	SELECT eventname, total...	10
<input type="checkbox"/>	255	Success	Apr 29th, 2022 03:16:04 AM UTC	196 ms	101	sample_data_dev	SELECT sum(qtysold) FRO...	10
<input type="checkbox"/>	109	Success	Apr 29th, 2022 03:12:08 AM UTC	287 ms	101	sample_data_dev	SELECT firstname, lastnam...	10

19. 実行されたクエリに付与されている Query ID をクリックすると、実行されたクエリが管理可能な画面に移動します
20. クエリ一覧に対象 Database が[dev]となっているものがあることに注目してください。これは Redshift Serverless の稼働状況などが dev データベースで管理されていることを意味します。つまり、マネージメントコンソール上での操作自体も、Redshift Serverless の Capacity を微小に消費することに留意してください。
21. では続いて、左ペインの [Resource Monitoring]をクリックしてください。[RPU capacity used]等何も表示されていないことがわかります。これは現時点で Preview が抱える問題でありバグ報告済です。このデータが入っている[dev]のテーブルに対して以下のクエリを発行してみます。Query Editor に戻り以下を実行してください。

```
select
    trunc(start_time) "Day"
    , sum(compute_capacity) RPU
    , sum(compute_seconds) RPU_seconds
    , (sum(compute_seconds)/3600)*0.5 "compute_cost in $"
    , sum(data_storage)/1024/1024/1024 "storage in GB"
from
    sys_serverless_usage
group by
    trunc(start_time);
```

今までに消費した RPU の累計及び課金額がわかります。

2 2. マネージメントコンソールの[Serverless Configuration]をクリックします。

[Endpoint]が表示されますので、コピーしておきます。

General information

Serverless namespace e5fb536c-a0c0-4b23-afb8-c75e2bf22ea3	Admin user name admin	Endpoint 294963776963.us-east-1.redshift-se...
Date created April 29, 2022, 12:05 (UTC+09:00)	Database name dev	JDBC URL jdbc:redshift://294963776963.us-east-1.redshift-se...
Status Available	Serverless credit remaining \$450.641	ODBC URL Driver={Amazon Redshift (x64)}; Ser...

2 3. [Recovery Points]にいくつかエントリが存在しています。Redshift Serverless では自動で 30 分おきに取得し、24 時間保存します。

Recovery points

Recovery points are automatically created every 30 minutes and kept for 24 hours to help you restore data after accidental writes or deletes.

Time range: Last 3 hours

Start time: April 28, 2022, 12:35 (UTC+09:00)

End time: April 28, 2022, 12:35 (UTC+09:00)

	Creation time	Data size
<input type="radio"/>	April 29, 2022, 12:35 (UTC+09:00)	18397 MB
<input type="radio"/>	April 29, 2022, 12:04 (UTC+09:00)	66 MB

2 4. [Limits]タブで[Base capacity in Redshift processing units (RPU)]の[Edit]ボタン

を押します。現在デフォルトとして設定されている RPU を調整することでデフォルトのパフォーマンスを変更させることができます。現時点の仕様では、この Capacity は自動で変動しないことに留意してください。(Amazon Aurora Serverless と異なり、Lambda と同じ形態です)

- 2 5. つぎに、Redshift Serverless や Query Editor のマネージメントコンソールを全て閉じた後、Cloud9 のマネージメントコンソールを開きます。(Redshift 関連マネージメントコンソールや Query Editor は、リロード時点で Serverless にクエリを実行してしまうためです)

- 2 6. [Create environment]をおします

- 2 7. 適当な名前を設定し、全てデフォルトのまま Cloud9 を起動します。Cloud9 はデフォルトで default VPC、つまり Redshift Serverless と同じ VPC に起動されます。

- 2 8. 以下のコマンドを実行します。Redshift は Postgresql 互換ですので、postgresql ライブラリから操作が可能です。

```
sudo yum install postgresql-server -y
```

- 2 9. 以下のコマンドを実行します

```
psql -h <endpoint> -U admin -d dev -p 5439
```

<endpoint>は先程コピーした値に置換します。その際、文字列後半の[:5439/dev]は取り除いて下さい。

以下ようになります、。

```
psql -h 294963776963.us-east-1.redshift-serverless.amazonaws.com -U admin -d dev -p 5439
```

- 3 0. パスワードが求められますので、入力をしてください。

- 3 1. 以下の SQL を実行すると値が表示されます。

```
select * from sample_data_dev.tickit.sales limit 10;
```

- 3 2. Redshift Serverless はセッションを 3600 秒保持します。このため 1 時間以上経過したのち再度クエリーを実行すると以下のように Cold スタートが再現します。の挙動は Preview 中に変更があることに留意してください。

```
ec2-user:~/environment $ psql -h 294963776963.us-east-1.redshift-serverless.amazonaws.com -U admin -d dev -p 5439
Password for user admin:
psql (9.2.24, server 8.0.2)
WARNING: psql version 9.2, server version 8.0.
         Some psql features might not work.
SSL connection (cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256)
Type "help" for help.

dev=#
dev=# SELECT * FROM sample_data_dev.tickit.sales limit 10;
ERROR: Remote cluster is initializing. Please try again later.
dev=#
```


おつかれさまでした！

削除は以下をおこなってください

- Redshift Serverless クラスター
- Cloud9