

Configure DMS to Preserve Timestamp Data without Timezone

Background

Customer has a timestamp column in RDS (Aurora-MySQL), the value is in timestamp format (e.g. `2023-02-24 18:00:01`) instead of timestamptz format (e.g. `2023-02-24 18:00:01+8`). They have both RDS and Redshift timezone set to `Asia/Shanghai`.

They would like to use DMS to migrate the data to Redshift. They are expecting the timestamp value to remain the same after migration.

NOTE: The best practice is to use UTC timeznoe at all database layers, or use timestamptz value instead of timestamp in the data.

Findings

Following tests shows that, for an RDS (Aurora-MySQL) database to preserve its timestamp (without timezone) value whlie it is migrated to Redshift, they need to have following 2 settings in their DMS endpoints.

- In the source endpoint of DMS task, add extra connection attribute `serverTimezone=Asia/Shanghai`
- In the target endpoint of DMS task, add extra connection attributes `initstmt=SET TIMEZONE='Asia/Shanghai'`

Test

Prerequisite

Setup RDS(MySQL) Cluster, Redshift Cluster, and DMS instance with a task to prelicate data between the RDS and Redshift. By default, the new RDS and Redshift cluster has an initial timezone = UTC.

Set RDS Timezone

Set the timezone of RDS to `Asia/Shanghai`.

1. Update the cluster parameter group of RDS.

RDS > Databases > database-3

database-3

Modify Actions

Related

Filter by databases

DB identifier	Role	Engine	Region & AZ	Size	Status	Actions	CPU
database-3	Regional cluster	Aurora MySQL	ap-southeast-1	2 instances	Available	-	-
database-3-instance-1	Writer instance	Aurora MySQL	ap-southeast-1a	db.t3.small	Available	-	10.85%
database-3-instance-1-rds	Reader instance	Aurora MySQL	ap-southeast-1b	db.t3.small	Available	-	11.11%

Connectivity & security | Monitoring | Logs & events | **Configuration** | Maintenance & backups | Tags

Database

Configuration

DB cluster role
Regional cluster

Engine version
5.7.mysql_aurora.2.11.1

Resource ID
cluster-LGDMBMSKFD4QA6XFQA2JPYDNNQ

Amazon Resource Name (ARN)
arn:aws:rds:ap-southeast-1:460453255610:cluster:database-3

Network type
IPv4

Capacity type
Provisioned: single-master

DB cluster ID
database-3

DB cluster parameter group
[my-cluster-parameter-group](#)

Deletion protection
Disabled

Availability

IAM DB authentication
Not enabled

Master username
admin

Master password

Multi-AZ
2 Zones

Encryption

Encryption
Enabled

AWS KMS key
[aws/rds](#)

Published logs

CloudWatch Logs
[Slow query](#)

2. Update the parameter group by setting `time-zone` to `Asia/Shanghai`.

RDS > Parameter groups > my-cluster-parameter-group

my-cluster-parameter-group

Edit parameters

time_zone

Name	Values	Allowed values	Modifiable	Source	Apply type	Data type	Description
default_time_zone			false	engine-default	static	string	Server current time zone
time_zone	Asia/Shanghai	Africa/Harare, Africa/Monrovia, Africa/Nairobi, Africa/Windhoek, America/Bogota, America/Caracas, America/Chihuahua, America/Cuiaba, America/Denver, America/Fortaleza, America/Guatemala, America/Halifax, America/Manaus, America/Matamoros, America/Monterrey, America/Montevidео, America/Phoenix, America/Tijuana, Asia/Ashgabat, Asia/Baghdad, Asia/Baku, Asia/Bangkok, Asia/Beirut, Asia/Calcutta, Asia/Kabul, Asia/Karachi, Asia/Kathmandu, Asia/Muscat, Asia/Riyadh, Asia/Seoul, Asia/Shanghai, Asia/Singapore, Asia/Taipei, Asia/Tehran, Asia/Tokyo, Asia/Ulaanbaatar, Atlantic/Azores, Australia/Adelaide, Australia/Brisbane, Australia/Darwin, Australia/Hobart, Australia/Perth, Australia/Sydney, Canada/Saskatchewan, Brazil/East, Europe/Amsterdam, Europe/Athens, Europe/Dublin, Europe/Helsinki, Europe/Paris, Europe/Prague, Europe/Sarajevo, Pacific/Auckland, Pacific/Guam, Pacific/Honolulu, Pacific/Samoa, US/Alaska, US/Central, US/Eastern, US/East-Indiana, US/Pacific, UTC	true	user	dynamic	string	for changing time zone of db server locally

3. Must restart all DB instances in the cluster.

4. Examine the timezone setting in the MySQL terminal.

```

1 mysql> SELECT @@global.time_zone, @@session.time_zone;
2 +-----+-----+
3 | @@global.time_zone | @@session.time_zone |
4 +-----+-----+
5 | Asia/Shanghai      | Asia/Shanghai      |
6 +-----+-----+
7 1 row in set (0.00 sec)

```

Create Sample Database Table in RDS

When both RDS and Redshift are in default timezone of UTC, the migrated

1. Create a table `test` with a timestamp.

```

1 CREATE TABLE `test` (
2   `id` bigint(20) NOT NULL,
3   `first_name` text,
4   `last_name` text,
5   `email` text,
6   `gender` text,
7   `ip_address` text,
8   `update_time` timestamp NULL DEFAULT NULL,
9   PRIMARY KEY (`id`)
10 ) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

2. Examine the table.

```

1 mysql> describe test;
2 +-----+-----+-----+-----+-----+-----+
3 | Field      | Type      | Null | Key | Default | Extra |
4 +-----+-----+-----+-----+-----+-----+
5 | id         | bigint(20) | NO   | PRI | NULL    |       |
6 | first_name | text       | YES  |     | NULL    |       |
7 | last_name  | text       | YES  |     | NULL    |       |
8 | email      | text       | YES  |     | NULL    |       |
9 | gender     | text       | YES  |     | NULL    |       |
10 | ip_address | text       | YES  |     | NULL    |       |
11 | update_time | timestamp  | YES  |     | NULL    |       |
12 +-----+-----+-----+-----+-----+-----+
13 7 rows in set (0.00 sec)

```

3. Insert some sample data in the table.

```

1 mysql> select * from test;
2 +-----+-----+-----+-----+-----+-----+-----+
3 | id | first_name | last_name | email | gender | ip_address | update_time |
4 +-----+-----+-----+-----+-----+-----+-----+
5 | 1 | alan | alan | alan@gmail.com | male | 1.0.0.1 | 2023-02-24 18:00:01 |
6 | 2 | bob | bob | bob@gmail.com | male | 2.0.0.2 | 2023-02-24 15:00:01 |
7 +-----+-----+-----+-----+-----+-----+-----+
8 2 rows in set (0.00 sec)

```

Test with Default Timezone for Redshift

1. Restart the DMS task to migrate data from RDS to Redshift.
2. Examine the data in Redshift. Their `update_time` field values are 8 hours earlier (-8) than the corresponding value in RDS.

<input type="checkbox"/> id	first_name	last_name	email	gender	ip_address	update_time	operation
<input type="checkbox"/> 1	alan	alan	alan@gmail.com	male	1.0.0.1	2023-02-24 10:00:01	NULL
<input type="checkbox"/> 2	bob	bob	bob@gmail.com	male	2.0.0.2	2023-02-24 07:00:01	NULL

Test with Redshift Timezone = 'Asia/Shanghai'

1. Using Redshift Query Editor, check its current timezone setting. The default timezone value is UTC.

```
1 SHOW TIMEZONE;
```

2. For testing purpose, set the timezone value to `Asia/Singapore`, which is the same timezone +8 as `Asia/Shanghai`.

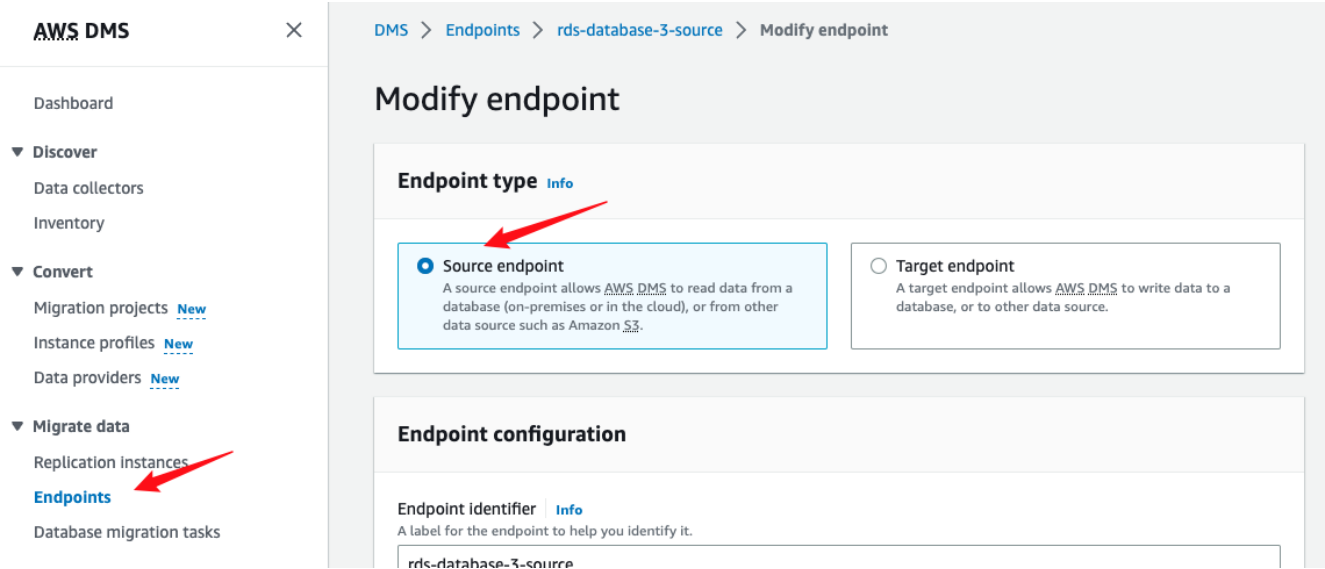
```
1 SET TIMEZONE='Asia/Shanghai';
2 SHOW TIMEZONE;
```

3. Restart the DMS task to migrate data from RDS to Redshift.
4. Examine the data in Redshift. Their `update_time` field values are 8 hours earlier (-8) than the corresponding value in RDS.

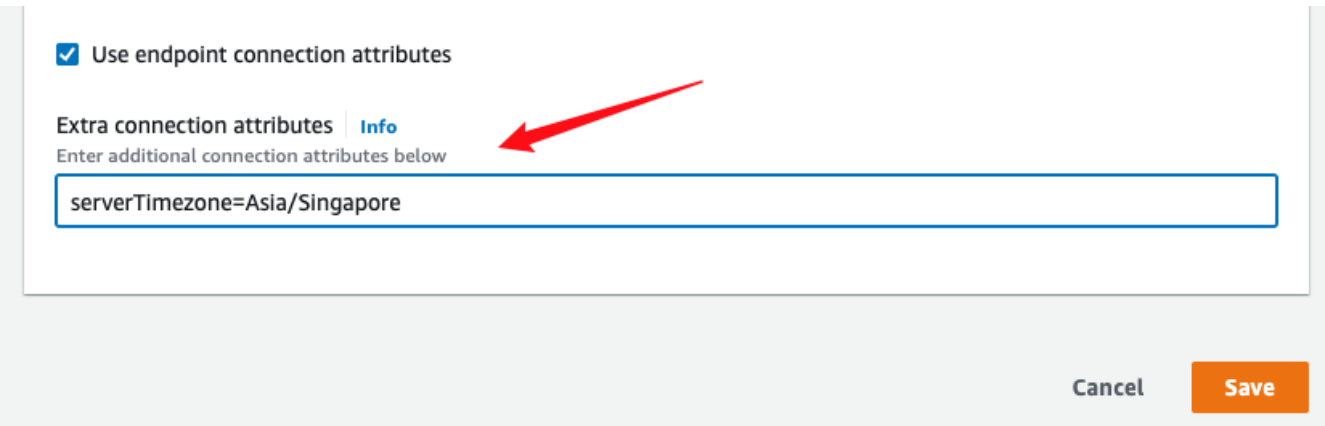
<input type="checkbox"/> id	first_name	last_name	email	gender	ip_address	update_time	operation
<input type="checkbox"/> 1	alan	alan	alan@gmail.com	male	1.0.0.1	2023-02-24 10:00:01	NULL
<input type="checkbox"/> 2	bob	bob	bob@gmail.com	male	2.0.0.2	2023-02-24 07:00:01	NULL

Test by Modifying DMS Source Endpoint

- 1. Modify the DMS source endpoint, which is pointing to the RDS.



- 2. Add an extra connection attributes `serverTimezone=Asia/Singapore` to specify the server timezone.



- 3. Examine the data in Redshift. Their `update_time` field values are 8 hours earlier (-8) than the corresponding value in RDS.

<input type="checkbox"/>	id	first_name	last_name	email	gender	ip_address	update_time	operation
<input type="checkbox"/>	1	alan	alan	alan@gmail.com	male	1.0.0.1	2023-02-24 10:00:01	NULL
<input type="checkbox"/>	2	bob	bob	bob@gmail.com	male	2.0.0.2	2023-02-24 07:00:01	NULL

Test by Modifying DMS Target Endpoint

- 1. Stop the DMS task and modify the DMS target endpoint.

AWS DMS

Dashboard

▼ Discover

Data collectors

Inventory

▼ Convert

Migration projects [New](#)

Instance profiles [New](#)

Data providers [New](#)

▼ Migrate data

Replication Instances

Endpoints

Database migration tasks

DMS > Endpoints > redshift-target > Modify endpoint

Modify endpoint

Endpoint type [Info](#)

☐ Source endpoint
A source endpoint allows AWS DMS to read data from a database (on-premises or in the cloud), or from other data source such as Amazon S3.

☒ Target endpoint
A target endpoint allows AWS DMS to write data to a database, or to other data source.

Endpoint configuration

Endpoint Identifier [Info](#)
A label for the endpoint to help you identify it.

redshift-target

Extra connection attributes [Info](#)
Use extra connection attributes to define additional endpoint settings for [AWS DMS](#)

```
initstmt=SET TIMEZONE='Asia/Singapore'
```

<input type="checkbox"/> id	first_name	last_name	email	gender	ip_address	update_time	operation
<input type="checkbox"/> 1	alan	alan	alan@gmail.com	male	1.0.0.1	2023-02-24 18:00:01	NULL
<input type="checkbox"/> 2	bob	bob	bob@gmail.com	male	2.0.0.2	2023-02-24 15:00:01	NULL