Resource Control - シミュレーションテスト

テスト環境

Topology: 12c tidb * 3 + 6c tikv * 3 + 6c pd * 3

シナリオ

3つのビジネステナントが同じクラスターに存在するシナリオ

ユーザー	リソースグループ	RU	Burstable		thread
OLTP	OLTP	4000	YES	重要なビジネスを実行するテナントをシミュレートし、リソースとレスポンスタイムを確保する必要があります。ビジネスには明確なピークとオフピークがあり、昼間にはビジネス量が多く、夜間には非常に少なくなります。	昼：32 夜間：8
BATCH	BATCH	400	YES	高いリソース消費を伴うテナントをシミュレートし、連続した大量のデータ抽出タスクが実行され、応答時間の要求は低いが、データパイプラインの堆積を避けるためにできるだけ速く完了する必要があります。	64
HR	HR	400	NO	重要でない業務をシミュレートし、応答時間に注意を払わず、リソースの過度の使用を避けるために、開発環境やOA（オフィスオートメーション）などの例があります。	8

> - **リソース制御** これはリソース管理の論理ユニットであり、同じリソースグループに属するセッションは同じリソースグループのリソースを共有する。
> - **Request Unit (RU)** これはTiDBにおいて、CPU、IOなどのシステムリソースを統一的に抽象化した単位です。
> - **Burstable** 余分なリソースが利用可能な場合、リソースグループの消費は通常のリソース制限を超えることができます。

テスト手順

1. 三つのビジネスが同時に実行され、OLTPピーク負荷がかかる。
2. OLTPビジネスがオフピーク時に入り、OLTPピーク負荷を減少させます。
3. OLTPビジネスが再びピーク時に戻り、OLTPピーク負荷に切り戻します。

テストデータ準備

```
MySQL [(none)]> CREATE RESOURCE GROUP oltp  RU_PER_SEC=4000 BURSTABLE;
Query OK, 0 rows affected (0.54 sec)

MySQL [(none)]> CREATE RESOURCE GROUP batch RU_PER_SEC=400 BURSTABLE;
Query OK, 0 rows affected (0.52 sec)

MySQL [(none)]> CREATE RESOURCE GROUP hr    RU_PER_SEC=400;
Query OK, 0 rows affected (0.52 sec)

MySQL [(none)]> CREATE USER 'oltp' RESOURCE GROUP oltp;
Query OK, 0 rows affected (0.03 sec)

MySQL [(none)]> GRANT ALL PRIVILEGES ON *.* TO 'oltp'@'%';
Query OK, 0 rows affected (0.03 sec)

MySQL [(none)]> CREATE USER 'batch' RESOURCE GROUP batch;
Query OK, 0 rows affected (0.02 sec)

MySQL [(none)]> GRANT ALL PRIVILEGES ON *.* TO 'batch'@'%';
Query OK, 0 rows affected (0.03 sec)

MySQL [(none)]> CREATE USER 'hr' RESOURCE GROUP hr;
Query OK, 0 rows affected (0.03 sec)

MySQL [(none)]> GRANT ALL PRIVILEGES ON *.* TO 'hr'@'%';
Query OK, 0 rows affected (0.03 sec)

MySQL [(none)]>

MySQL [(none)]> select * from information_schema.resource_groups
    -> where name in ('oltp', 'batch', 'hr');
+-------+-----------+-----------+
| NAME  | RU_PER_SEC | BURSTABLE |
```

```
+-------+-----------+-----------+
| batch |      4000 | YES       |
| hr    |       400 | NO        |
| oltp  |       400 | YES       |
+-------+-----------+-----------+
3 rows in set (0.01 sec)


MySQL [(none)]> select User, User_attributes from mysql.user
    -> where user in ('oltp', 'batch', 'hr');
+-------+----------------------------+
| User  | User_attributes            |
+-------+----------------------------+
| oltp  | {"resource_group": "oltp"}  |
| batch | {"resource_group": "batch"} |
| hr    | {"resource_group": "hr"}    |
+-------+----------------------------+
3 rows in set (0.00 sec)
```

負荷テスト

この実験では、sysbenchのreadonly負荷を使用して、読み取りシナリオでのリソース制御のフィードバックをテストする。

クライアント設定

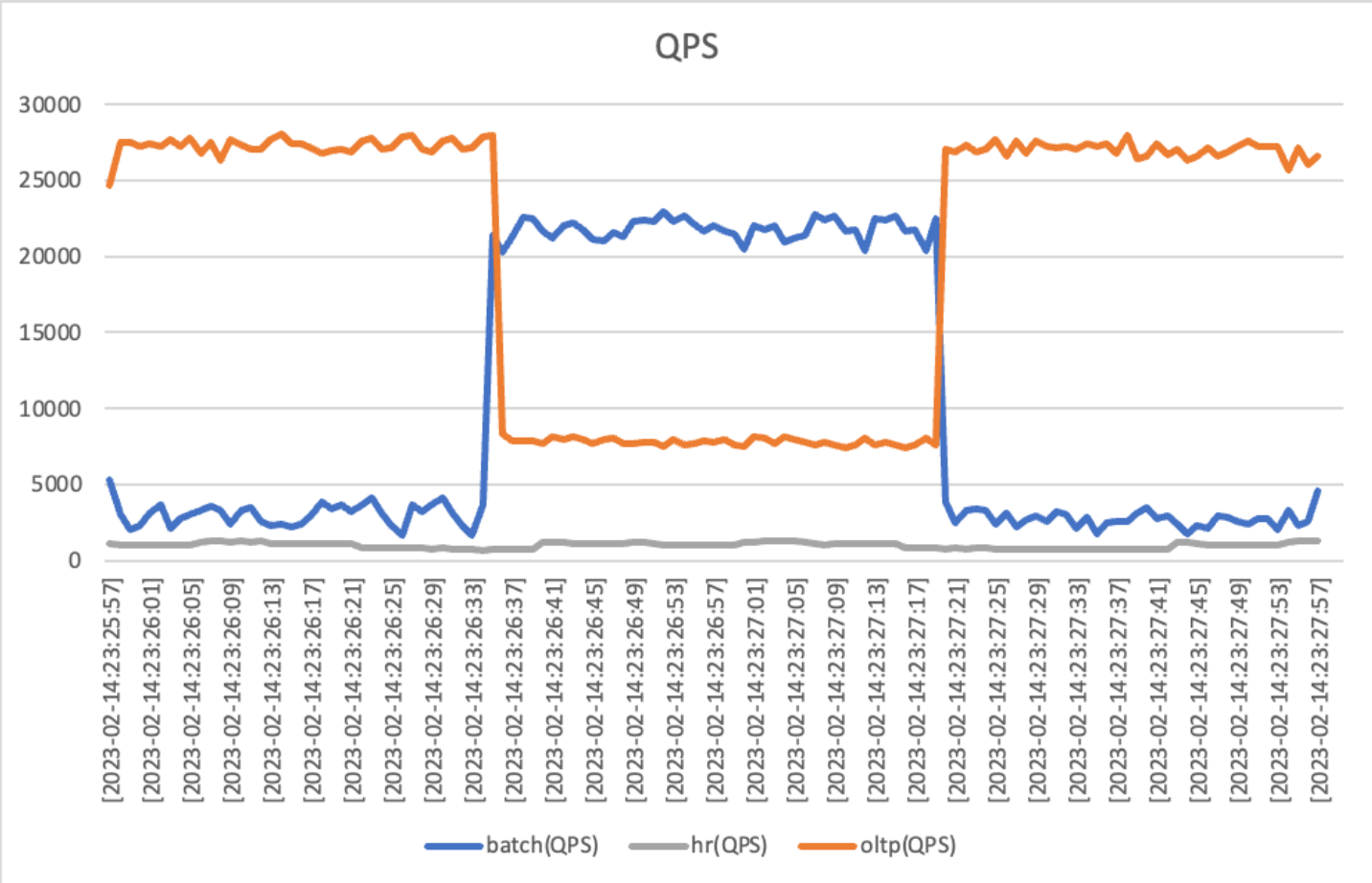
「Batch」を例に取る

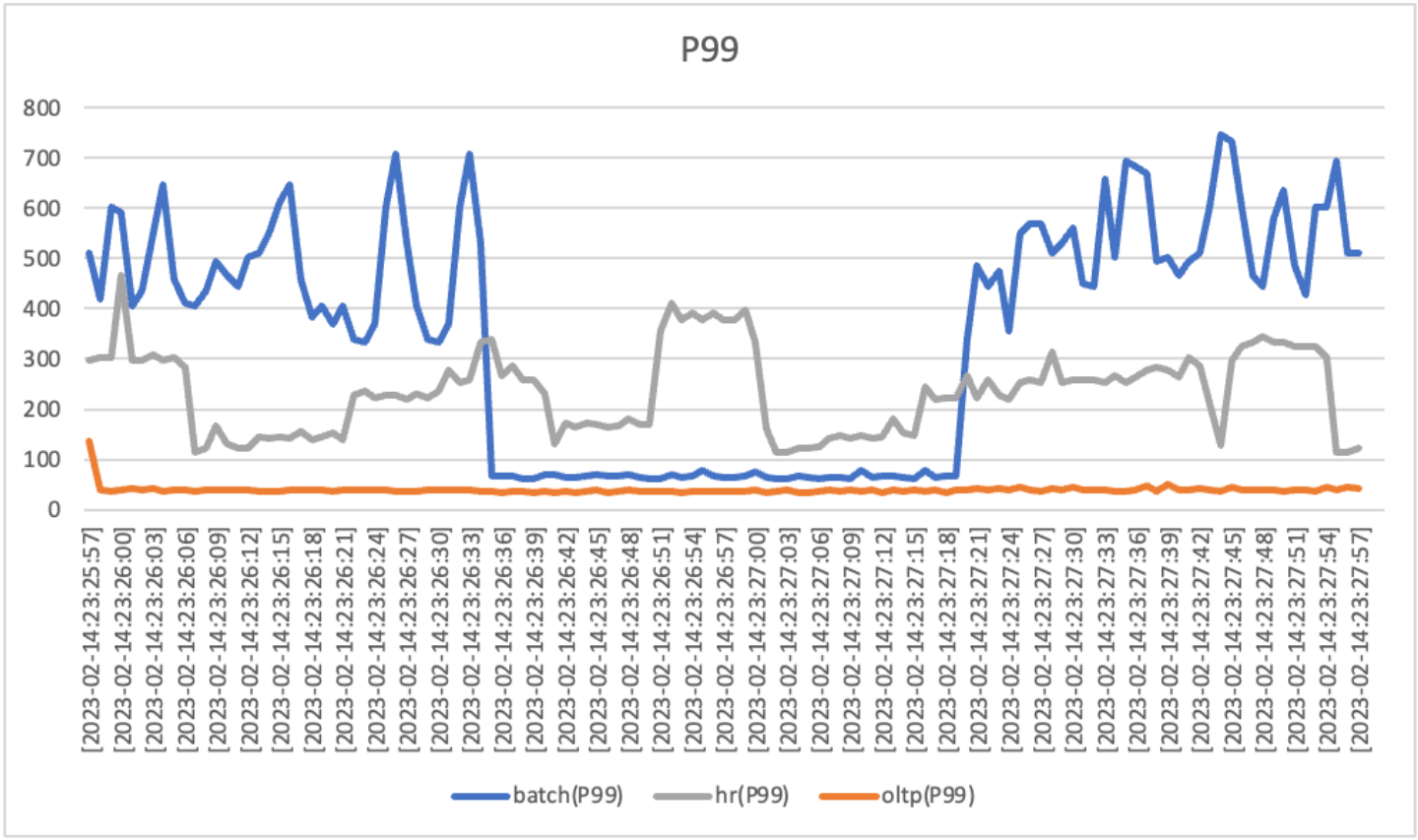
```
sysbench \
--mysql-host=10.68.212.81 \
--mysql-user=oltp \
--mysql-port=4000 \
--mysql-db=sbtest \
--percentile=99 \
--report-interval=1 \
--tables=32 \
--table_size=10000000 \
oltp_read_only \
--range_selects \
--threads=64 \
--time=1800 \
run \
| gawk '{ print strftime("[%Y-%m-%d %H:%M:%S] [batch]"), $0 }'
```

結果

負荷が変化した場合、リソース制御のフィードバックが非常にタイムリーで、ほとんど遅延がありません。

- OLTPテナントの負荷が下がる：
  - BatchテナントのQPSが明らかに向上し、瞬時に自動的に完了する。
  - hrの応答は変わらず維持されている
- OLTPテナントの負荷が再び上昇する：
  - oltpのQPSは即座に元の水準に戻る
  - Batchテナントはすぐにリソースを解放し、QPSが減少し、P99が上昇する
  - この過程中、OLTPテナントのP99にほとんど影響しない




P99
800
700
600
500
400
300
200
100
0
[2023-02-14:23:25:57]
[2023-02-14:23:26:00]
[2023-02-14:23:26:03]
[2023-02-14:23:26:06]
[2023-02-14:23:26:09]
[2023-02-14:23:26:12]
[2023-02-14:23:26:15]
[2023-02-14:23:26:18]
[2023-02-14:23:26:21]
[2023-02-14:23:26:24]
[2023-02-14:23:26:27]
[2023-02-14:23:26:30]
[2023-02-14:23:26:33]
[2023-02-14:23:26:36]
[2023-02-14:23:26:39]
[2023-02-14:23:26:42]
[2023-02-14:23:26:45]
[2023-02-14:23:26:48]
[2023-02-14:23:26:51]
[2023-02-14:23:26:54]
[2023-02-14:23:26:57]
[2023-02-14:23:27:00]
[2023-02-14:23:27:03]
[2023-02-14:23:27:06]
[2023-02-14:23:27:09]
[2023-02-14:23:27:12]
[2023-02-14:23:27:15]
[2023-02-14:23:27:18]
[2023-02-14:23:27:21]
[2023-02-14:23:27:24]
[2023-02-14:23:27:27]
[2023-02-14:23:27:30]
[2023-02-14:23:27:33]
[2023-02-14:23:27:36]
[2023-02-14:23:27:39]
[2023-02-14:23:27:42]
[2023-02-14:23:27:45]
[2023-02-14:23:27:48]
[2023-02-14:23:27:51]
[2023-02-14:23:27:54]
[2023-02-14:23:27:57]
batch(P99)
hr(P99)
oltp(P99)