

基于Greenplum to Greenplum 实现数据联邦

赵雪静

VMware 高级研发工程师

vmware®

EXPLORE

2022

Greenplum to Greenplum



背景



gp_parallel_retrieve_cursor



greenplum fdw

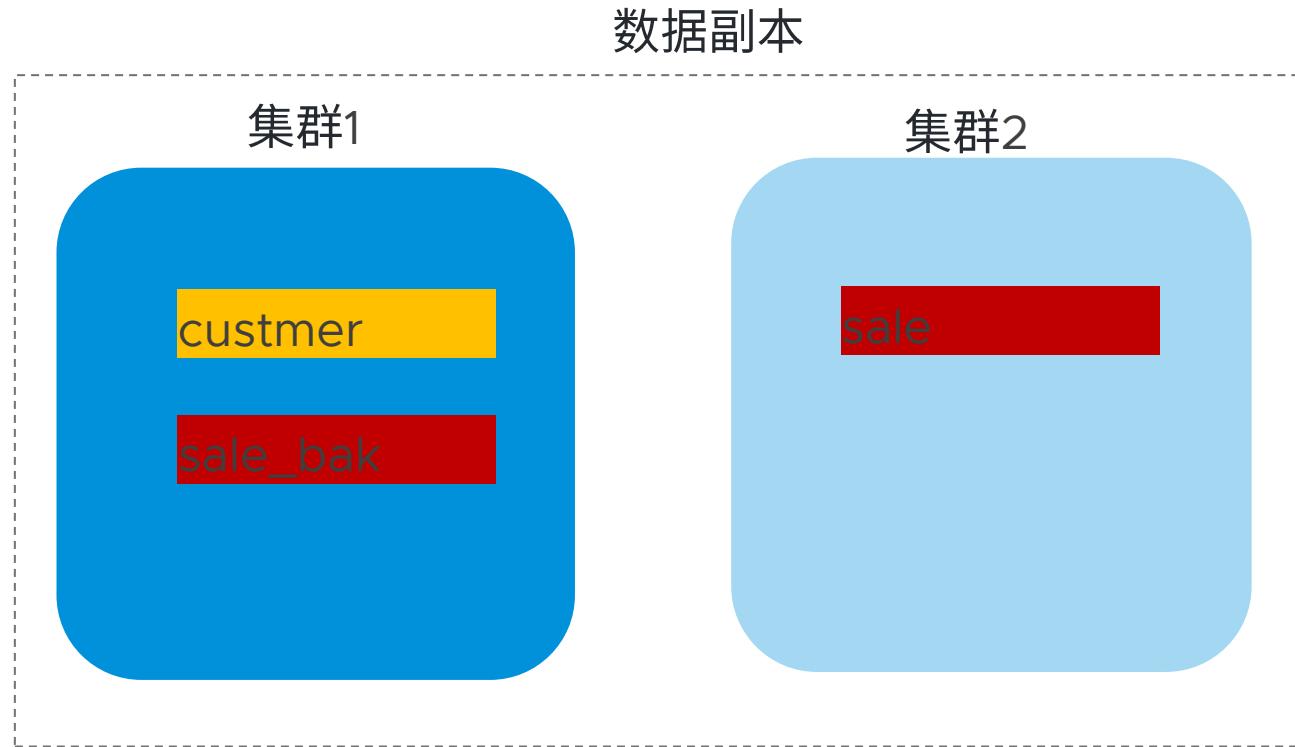


演示

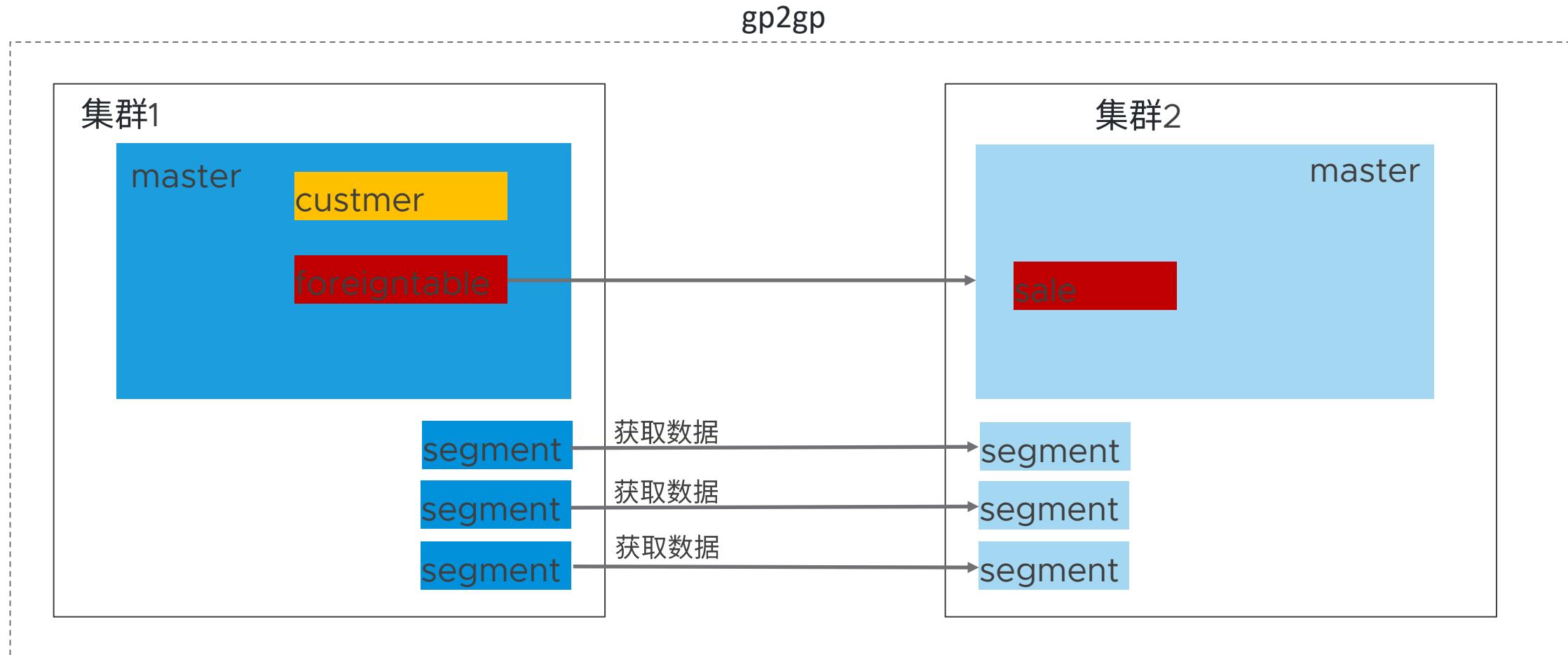
Greenplum to Greenplum背景

很多大企业中有多个Greenplum集群用于各种业务，很多情况下，由于业务目的，他们经常需要在一个集群中访问另外一个集群中的数据，当然可以通过在不同集群中存储相同数据的副本解决跨集群查询问题，但这会造成数据冗余、磁盘浪费，gp2gp功能可以帮助用户方便、快速的实现跨集群的数据并行查询。

Greenplum to Greenplum背景



Greenplum to greenplum



Greenplum to Greenplum

背景

gp_parallel_retrieve_cursor

greenplum_fdw

演示

Gp_parallel_retrieve_cursor组件

Parallel retrieve cursor

- cursor
 - 允许用户通过fetch方式一次获取部分数据，避免一次性处理所有结果可能造成的内存溢出。
- gp_parallel_retrieve_cursor
 - 是一种特殊的cursor，最终数据无需通过gather motion汇总到QD，而是保存在多个节点中，允许用户通过retrieve方式直连节点，从节点并行获取数据。

Gp_parallel_retrieve_cursor组件

parallel retrieve cursor

- Normal cursor

```
postgres=# explain declare c1 cursor for select * from t1;
          QUERY PLAN
-----
 Gather Motion 3:1  (slice1; segments: 3)  (cost=0.00..1063.00 rows=96300 width=14)
   -> Seq Scan on t1  (cost=0.00..1063.00 rows=32100 width=14)
 Optimizer: Postgres query optimizer
(3 rows)
```

- Parallel retrieve cursor

```
postgres=# explain declare c1 parallel retrieve cursor for select * from t1;
          QUERY PLAN
-----
 Seq Scan on t1  (cost=0.00..1063.00 rows=32100 width=14)
 Optimizer: Postgres query optimizer
(2 rows)
```

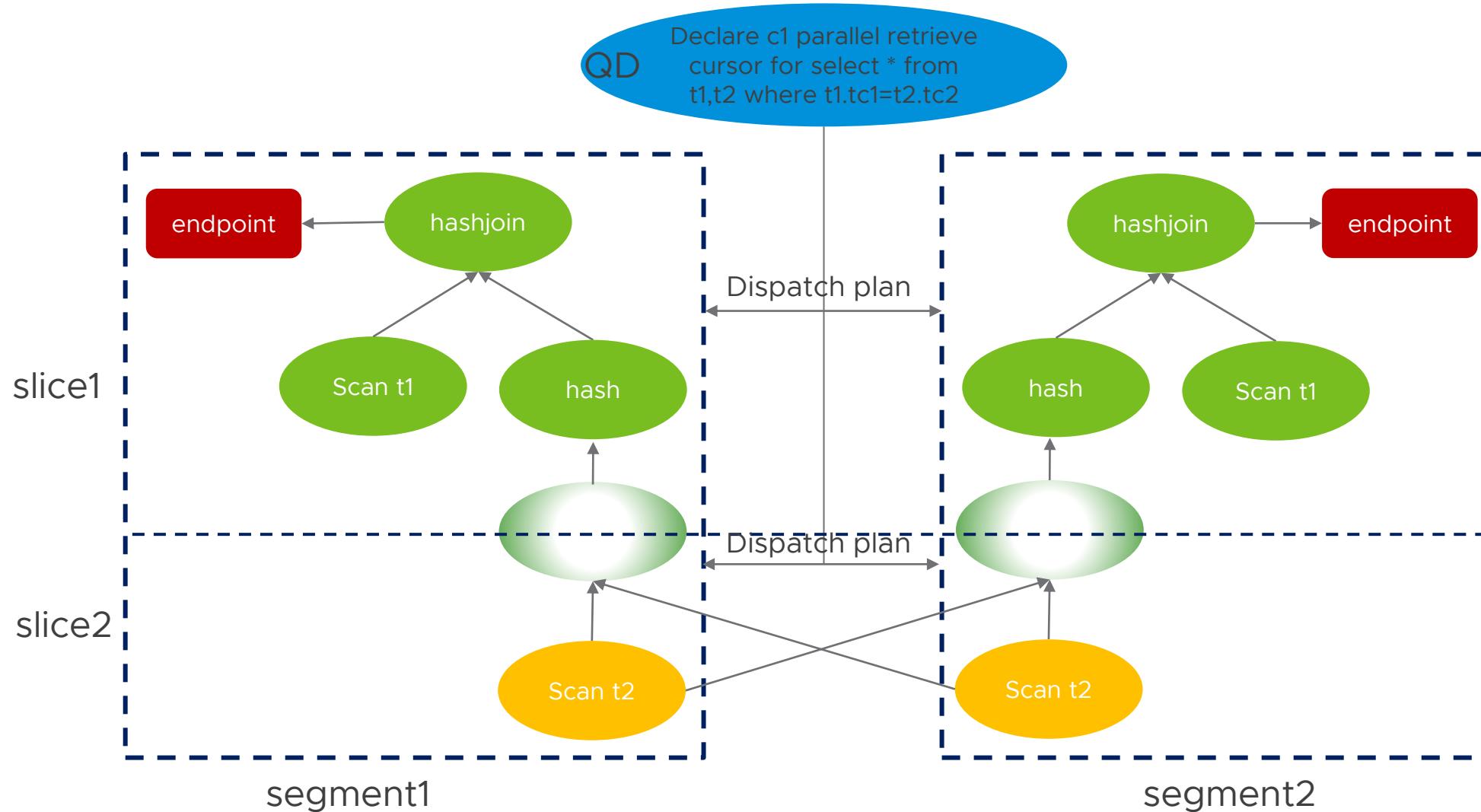
Gp_parallel_retrieve_cursor组件

Endpoint

一个Endpoint和一个消息队列相关联，顶层的QE通过将最终数据存储在这个节点对应的Endpoint，而不是将数据gather到QD，并且可以由另一个进程使用retrieve方式连接到该endpoint，并获取对应的数据，一个endpoint代表了该节点上result source。

Gp_parallel_retrieve_cursor组件

Endpoint



Gp_parallel_retrieve_cursor组件

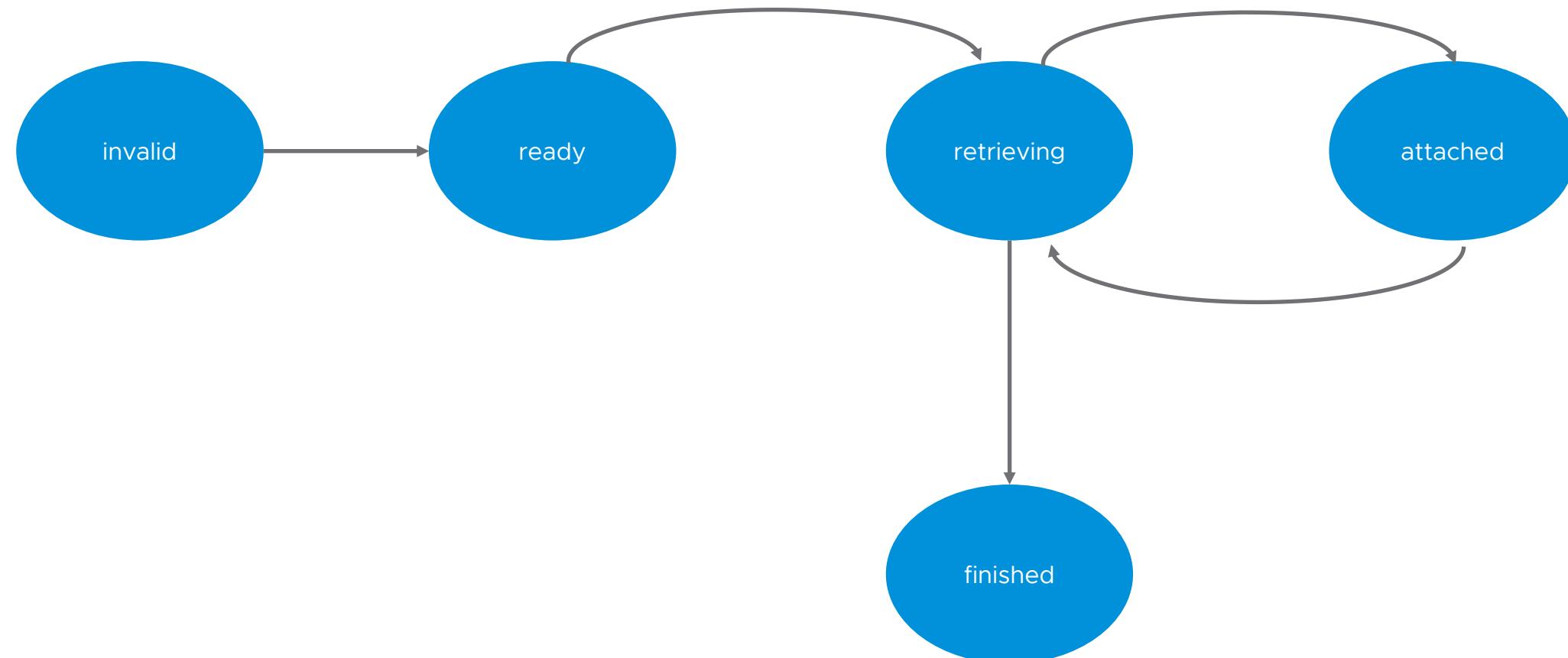
Endpoint

可以通过gp_endpoints视图来获取到parallel retrieve cursor相关联的所有endpoint信息。

```
postgres=# declare c1 parallel retrieve cursor for select * from t1;
DECLARE PARALLEL RETRIEVE CURSOR
postgres=# select * from gp_endpoints;
 gp_segment_id | auth_token          | cursorname | sessionid | hostname | port | username | state | endpointname
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
 0 | 0518139548e08639f234fd756ed8f0b6 | c1         |           | 33       | ubuntu   | 6002    | gpadmin  | READY   | c10000002100000003
 1 | a2ae1128847909885767adf47609203a | c1         |           | 33       | ubuntu   | 6003    | gpadmin  | READY   | c10000002100000003
 2 | c97e980f1dd8878866464875b92a443e | c1         |           | 33       | ubuntu   | 6004    | gpadmin  | READY   | c10000002100000003
(3 rows)
```

Gp_parallel_retrieve_cursor组件

Endpoint对应状态信息



Gp_parallel_retrieve_cursor组件

- Normal cursor使用

psql连接master

Declare

```
postgres=# declare c1 cursor for select * from t1;  
DECLARE CURSOR
```

fetch

```
postgres=# fetch 1 c1;  
 tc1  
----  
 2  
(1 row)
```

Gp_parallel_retrieve_cursor组件

➤ Parallel retrieve cursor使用

psql连接master

Declare

```
postgres=# declare c1 parallel retrieve cursor for select * from t1;
DECLARE PARALLEL RETRIEVE CURSOR
```

Get endpoint

```
postgres=# select * from gp_endpoints;
 gp_segment_id | auth_token          | cursorname | sessionid | hostname | port | username | state | endpointname
-----+-----+-----+-----+-----+-----+-----+-----+-----+
      0 | d5f0e7046d31704c227d6e55ee834d3b | c1         |        7 | ubuntu   | 6002 | gpadmin  | READY | c1000000070000000d
      1 | 969c931282506c08123d5afcf2584331 | c1         |        7 | ubuntu   | 6003 | gpadmin  | READY | c1000000070000000d
      2 | a80c714506aebcecea34b651b44889fb | c1         |        7 | ubuntu   | 6004 | gpadmin  | READY | c1000000070000000d
(3 rows)
```

psql直连某个endpoint节点

```
gpadmin@ubuntu:~$ PGOPTIONS=''-c gp_retrieve_conn=true' psql -d postgres -h ubuntu -p 6003
Password:
```

Retrieve

```
postgres=# retrieve 1 from endpoint c1000000070000000d;
 tc1
-----
 1
(1 row)
```

Gp_parallel_retrieve_cursor组件

Parallel retrieve cursor提供了如下几个function供用户方便的获取Parallel Retrieve Cursor的信息以及查看endpoint中的数据是否被retrieve完成：

- gp_get_endpoints
- gp_get_session_endpoints
- gp_get_segment_endpoints
- gp_wait_parallel_retrieve_cursor

Greenplum to Greenplum



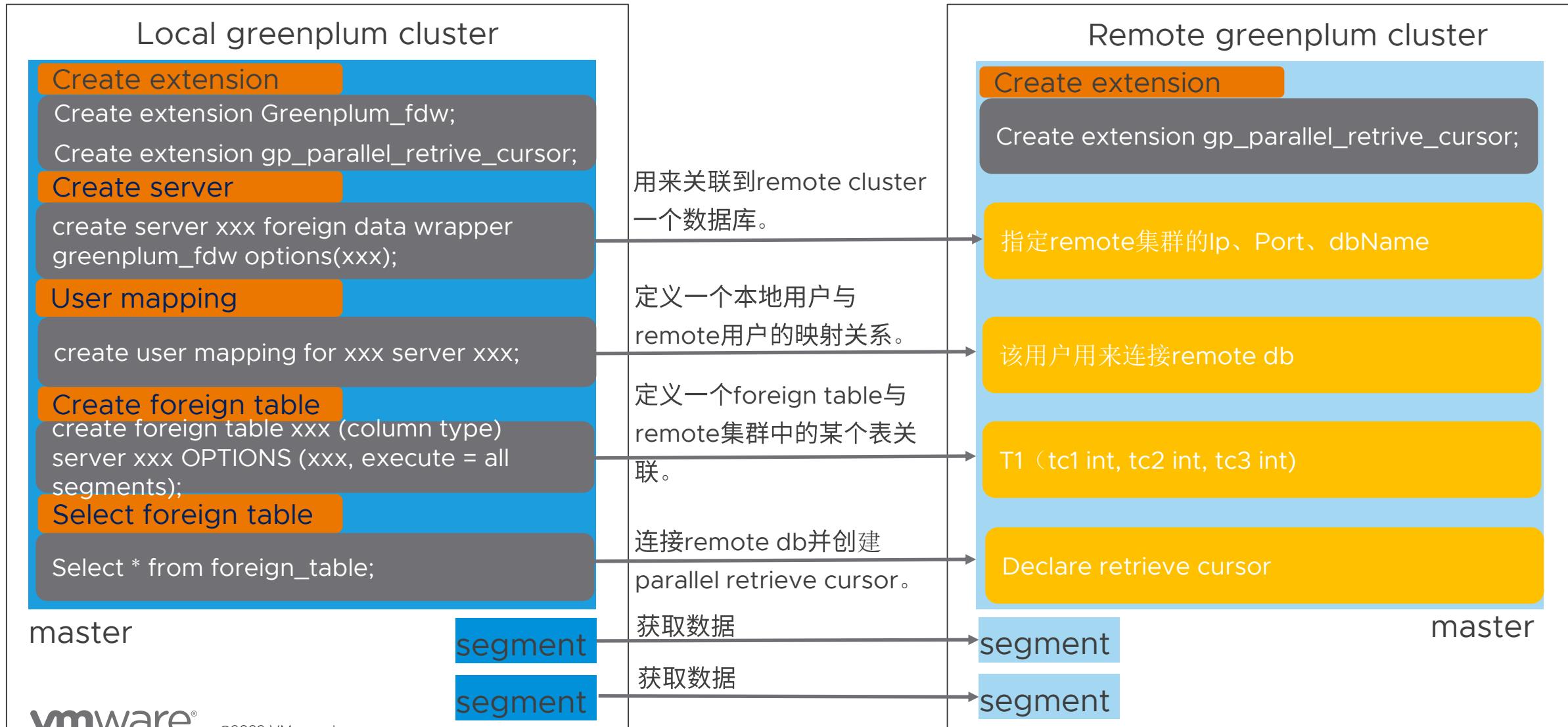
Greenplum_fdw组件

Greenplum_fdw

- PostgreSQL_fdw
是一种外部访问接口，它可以被用来访问存储在该数据库以外其他PostgreSQL数据库中的数据。
- Greenplum_fdw
是PostgreSQL fdw的mpp化版本，借助parallel retrieve cursor从远端集群并行获取数据。

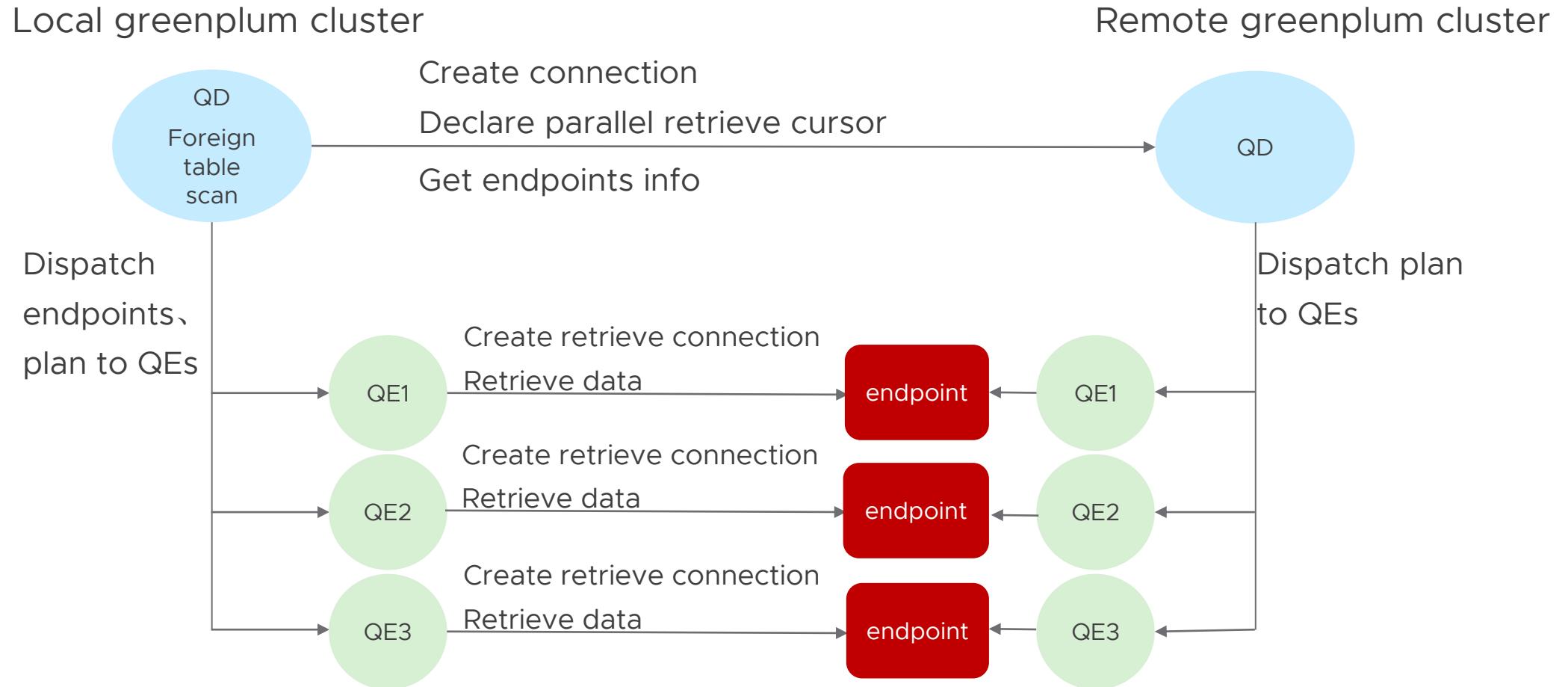
Greenplum_fdw组件

Greenplum_fdw使用



Greenplum_fdw组件

Greenplum fdw并行查询实现



Greenplum_fdw组件

- Show foreign server

\des(+)

```
Select * from pg_foreign_server;
```

- Show user mapping

\deu(+)

```
Select * from pg_user_mapping;
```

- Show foreign table

\det(+)

```
Select * from pg_foreign_table;
```

Greenplum to Greenplum



Greenplum to Greenplum简介

Greenplum to greenplulm使用限制

- Orca优化器不支持对外部表进行查询
- Gp2gp不支持对外部表update、delete操作

Greenplum to Greenplum

相关资料

Greenplum fdw:

<https://github.com/greenplum-db/gpdb/wiki/Greenplum-to-Greenplum>

Parallel retrieve cursor:

<https://github.com/greenplum-db/gpdb/wiki/Parallel-Retrieve-Cursor>

谢谢！

