RapidsDB
A BORRUI DATA COMPANY

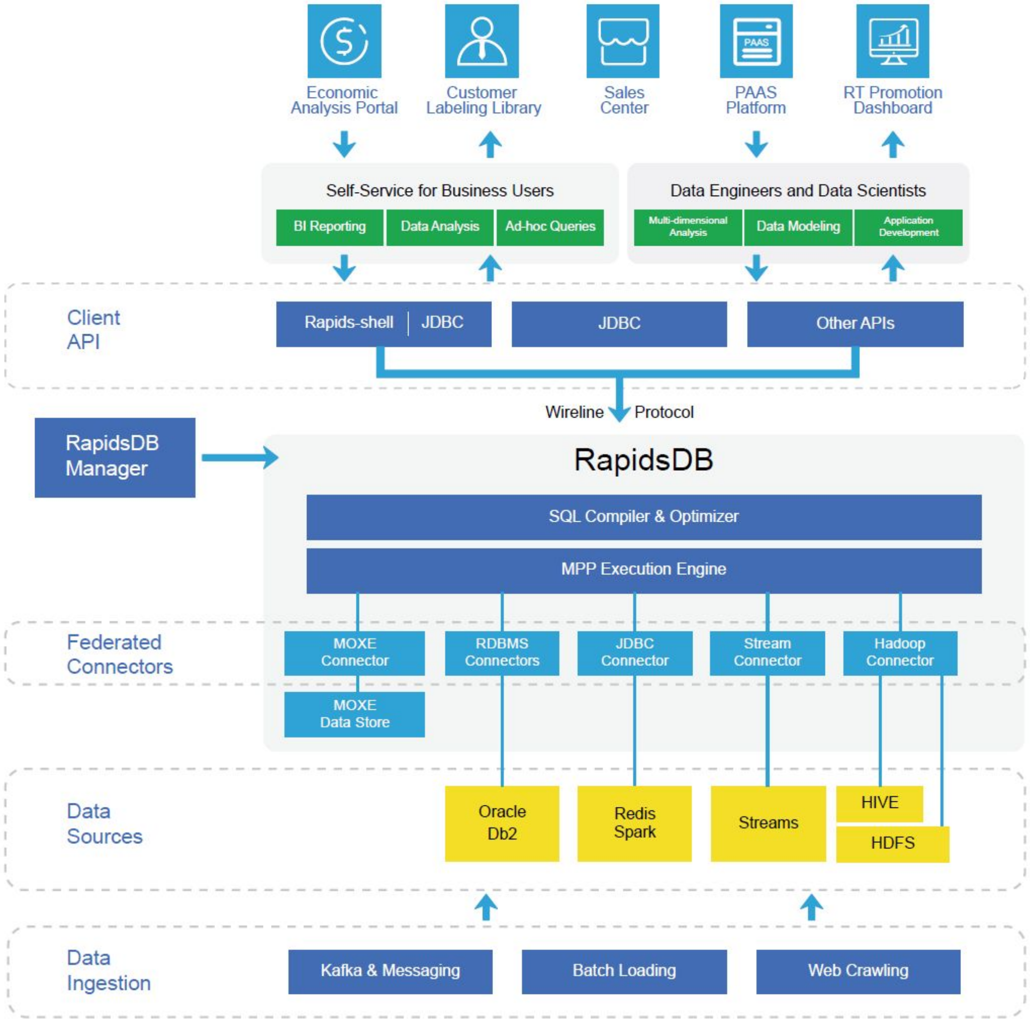
ISO BigData

CODE DEMO

World Leading Real-time Big Data Analytics Platform

OverView

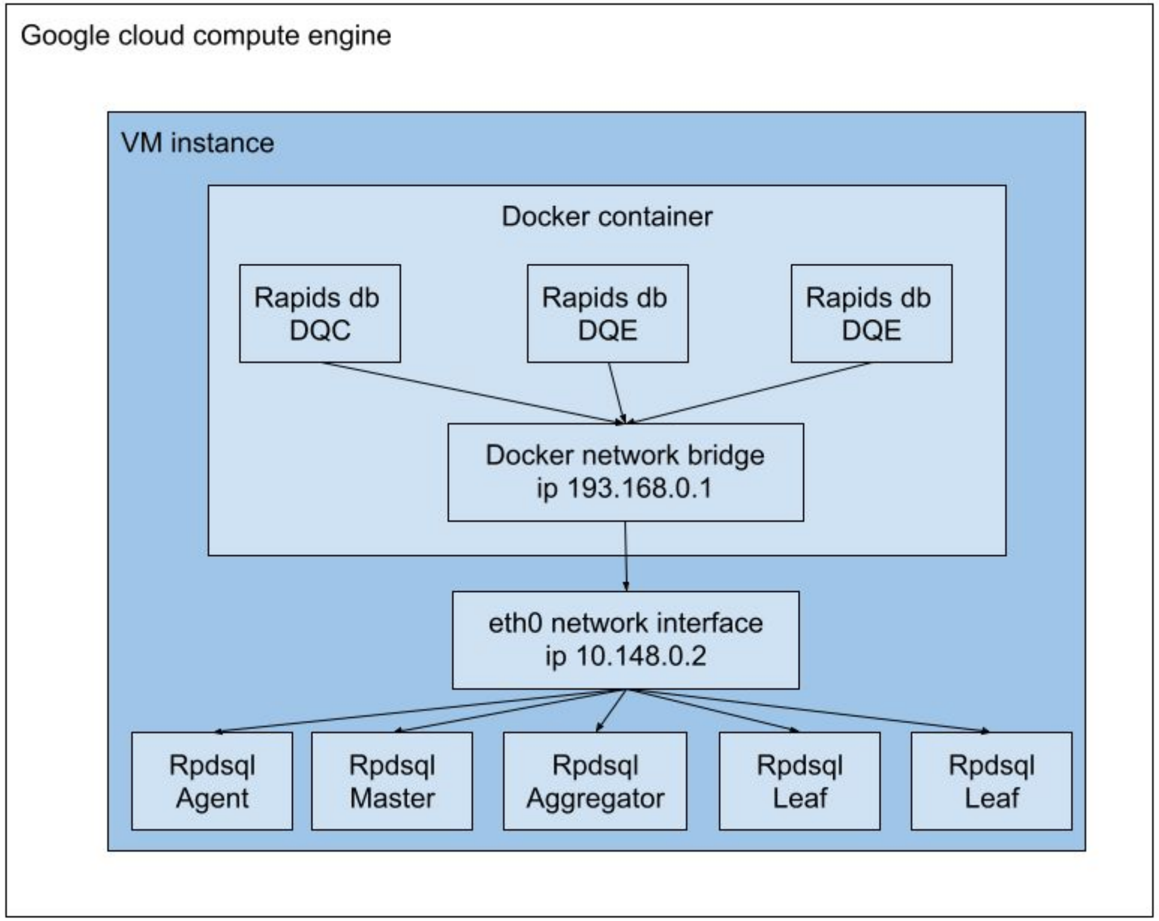
- Rapids Shell Login
- MYSQL / RPDSQL Connector
- HDFS / HIVE Connector
- S3 Data Import
- CSV Data Import
- JSON Data Import
- TPCH Query
- Python Code Example
- GEO Polygon Query
- Create index
- Data distribute and diagnostic
- User management
- Data backup and restore
- Procedure and Transaction
- Federation table join
- High availability



DEMO Architecture

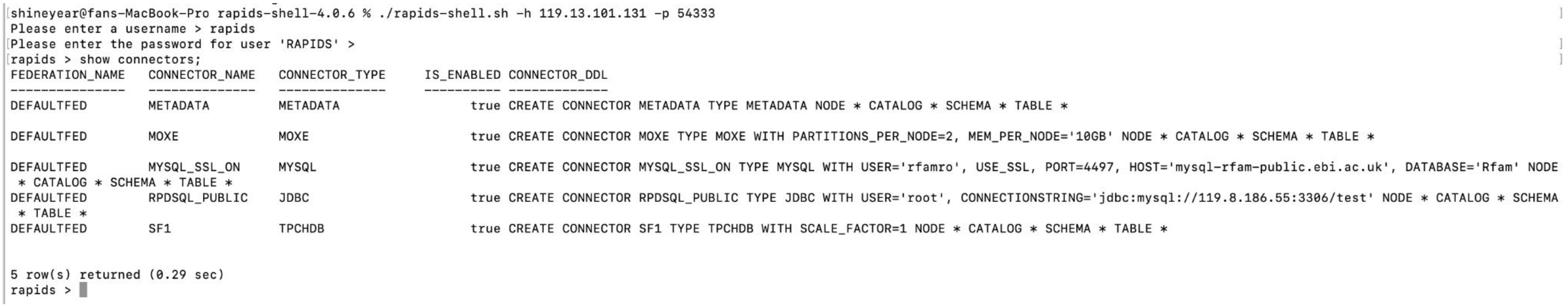
VM + Docker ALL IN ONE

```
/home/sbsadmin/rapidsdb

/home/sbsadmin/rapidsdb/rapids-shell-4.0.6

/opt/rdp/

/opt/rdp/rpdsql-ops
```



Rapids Shell Login

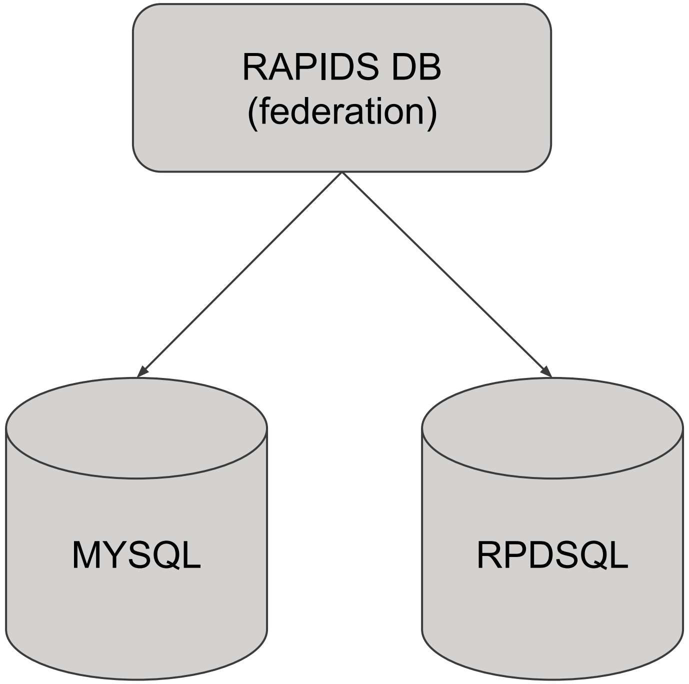
```
rapids-shell.sh -h ip -p port ##then input username and password

show connectors;

use connector CONNECTOR_NAME;

use connector rapids; (back to top level of connectors)
```



```
[shineyear@fans-MacBook-Pro rapids-shell-4.0.6 % ./rapids-shell.sh -h 119.13.101.131 -p 54333
Please enter a username > rapids
[Please enter the password for user 'RAPIDS' >
[rapids > show connectors;
FEDERATION_NAME   CONNECTOR_NAME   CONNECTOR_TYPE     IS_ENABLED CONNECTOR_DDL
---------------   --------------   --------------     ---------- -------------
DEFAULTFED        METADATA         METADATA                 true CREATE CONNECTOR METADATA TYPE METADATA NODE * CATALOG * SCHEMA * TABLE *

DEFAULTFED        MOXE             MOXE                     true CREATE CONNECTOR MOXE TYPE MOXE WITH PARTITIONS_PER_NODE=2, MEM_PER_NODE='10GB' NODE * CATALOG * SCHEMA * TABLE *

DEFAULTFED        MYSQL_SSL_ON     MYSQL                    true CREATE CONNECTOR MYSQL_SSL_ON TYPE MYSQL WITH USER='rfamro', USE_SSL, PORT=4497, HOST='mysql-rfam-public.ebi.ac.uk', DATABASE='Rfam' NODE
 * CATALOG * SCHEMA * TABLE *
DEFAULTFED        RPDSQL_PUBLIC    JDBC                     true CREATE CONNECTOR RPDSQL_PUBLIC TYPE JDBC WITH USER='root', CONNECTIONSTRING='jdbc:mysql://119.8.186.55:3306/test' NODE * CATALOG * SCHEMA
 * TABLE *
DEFAULTFED        SF1              TPCHDB                   true CREATE CONNECTOR SF1 TYPE TPCHDB WITH SCALE_FACTOR=1 NODE * CATALOG * SCHEMA * TABLE *


5 row(s) returned (0.29 sec)
rapids >
```

MYSQL /MSSQL/ RPDSQL Connector

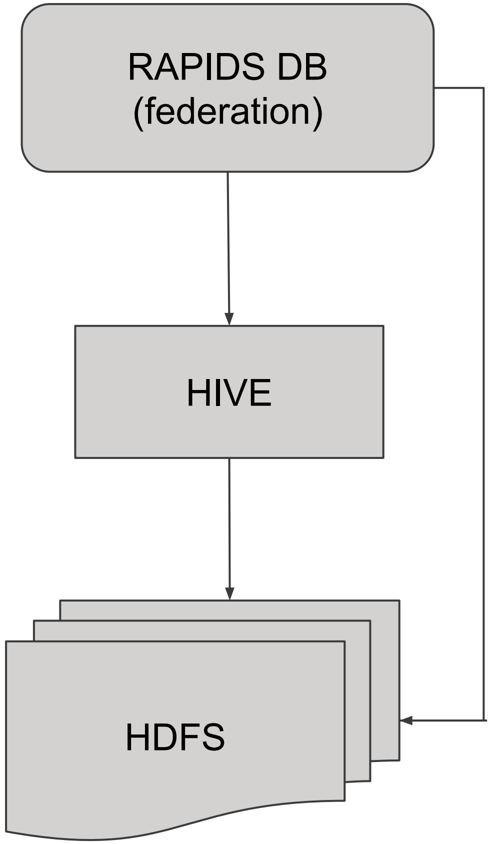
```
CREATE CONNECTOR MYSQL_SSL_ON
TYPE MYSQL WITH
DATABASE='Rfam',
USER='rfamro',
PORT=4497,
USE_SSL='TRUE',
HOST='mysql-rfam-public.ebi.ac.uk'
NODE * CATALOG * SCHEMA * TABLE *;


CREATE CONNECTOR RPDSQL_PUBLIC
TYPE JDBC WITH CONNECTIONSTRING='jdbc:mysql://119.8.186.55:3306/test',
USER='root'
NODE * CATALOG * SCHEMA * TABLE *;

CREATE CONNECTOR MSSQL TYPE JDBC WITH
CONNECTIONSTRING='jdbc:sqlserver://172.26.58.19:1433;databaseName=ifrs9_et
l_arc;trustServerCertificate=true', USER='rapidsdb_etl',
PASSWORD='P@ssw0rd123' NODE * CATALOG ifrs9_etl_arc SCHEMA ifrs9_etl_arc
TABLE *;

refresh;

#can replace mysql to db2, sqlserver or anyother jdbc support database
```



HDFS / HIVE Connector

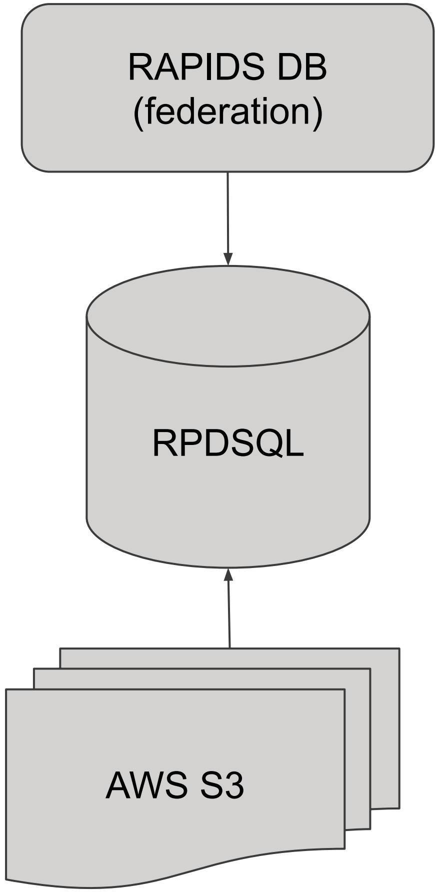
```
CREATE CONNECTOR HDFS5
TYPE HADOOP
WITH hdfs='hdfs://159.138.83.27:9000',
format='delimited',
delimiter=',',
user='hadoop',
partitions_per_node = '1'
CATALOG * SCHEMA *
TABLE TEST5 (C1 INTEGER, C2 VARCHAR)
WITH PATH='/test5';
```

```
CREATE CONNECTOR HIVE2
TYPE JDBC
WITH CONNECTIONSTRING='jdbc:hive2://127.0.0.1:10000/default'
NODE * CATALOG * SCHEMA * TABLE *;
```



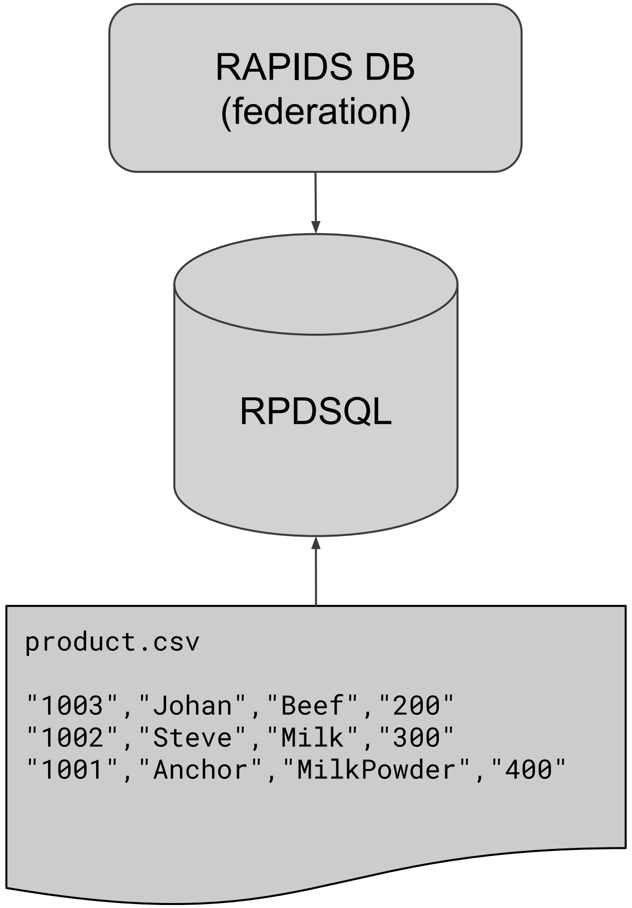
S3 Data Import

```
use connector RPDSQL_PUBLIC;

create table beta (num VARCHAR(20), price VARCHAR(20));


CREATE PIPELINE p AS LOAD DATA S3 'bucket-name'
CONFIG '{"region": "region-name"}'
CREDENTIALS '{
"aws_access_key_id": "key-id",
"aws_secret_access_key": "secret-key"
}'
SKIP DUPLICATE KEY ERRORS
INTO TABLE beta
FIELDS TERMINATED BY ',' ENCLOSED BY '' ESCAPED BY '\\'
LINES TERMINATED BY '\n' STARTING BY '';

test pipeline p;

show pipelines;

start pipeline p;
```



CSV Data Import

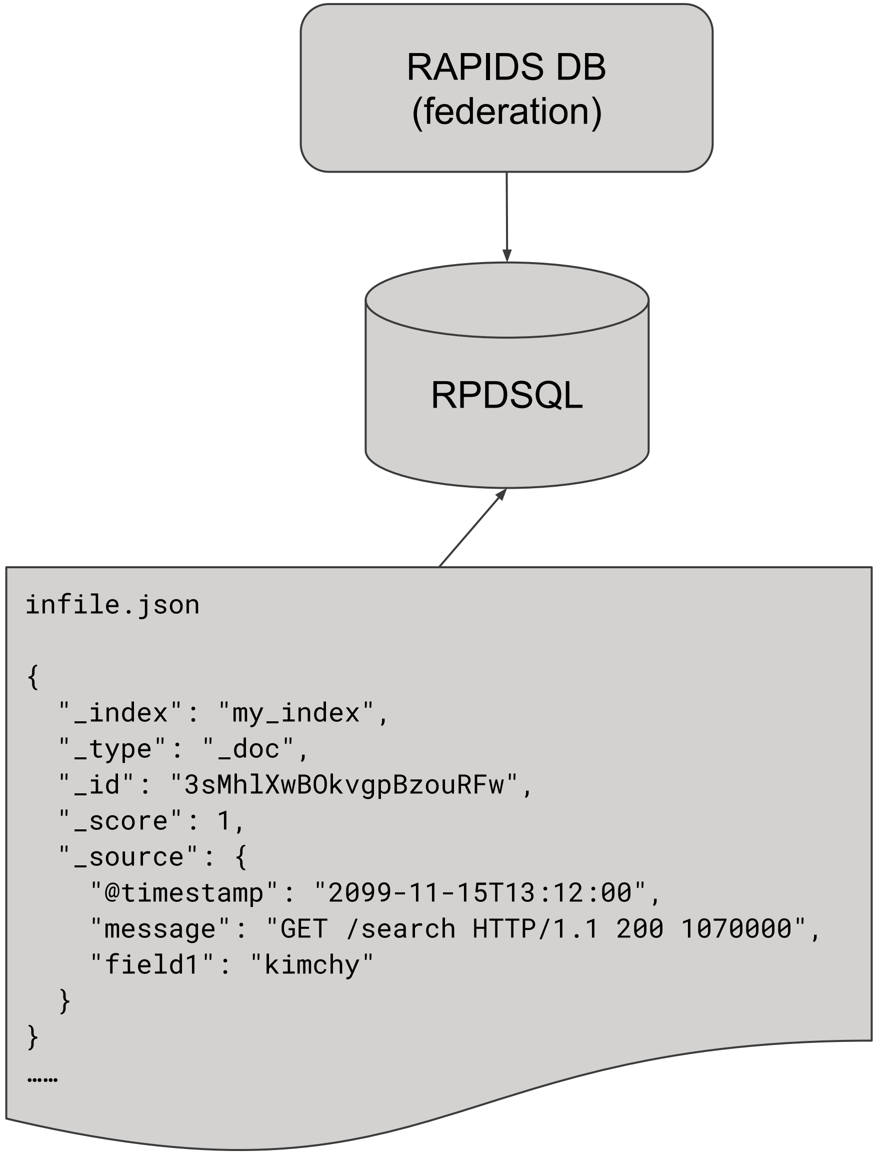
```
use connector RPDSQL_PUBLIC;

create table csv (
id VARCHAR(100) NOT NULL,
name VARCHAR(100) NOT NULL,
category VARCHAR(100) NOT NULL,
price VARCHAR(100) NOT NULL,
PRIMARY KEY ( name )
);

LOAD DATA LOCAL  INFILE 'product.csv' INTO
TABLE csv COLUMNS TERMINATED BY ',' ENCLOSED
BY '"';

select * from csv;
```



```
product.csv

"1003","Johan","Beef","200"
"1002","Steve","Milk","300"
"1001","Anchor","MilkPowder","400"
```

JSON Data Import

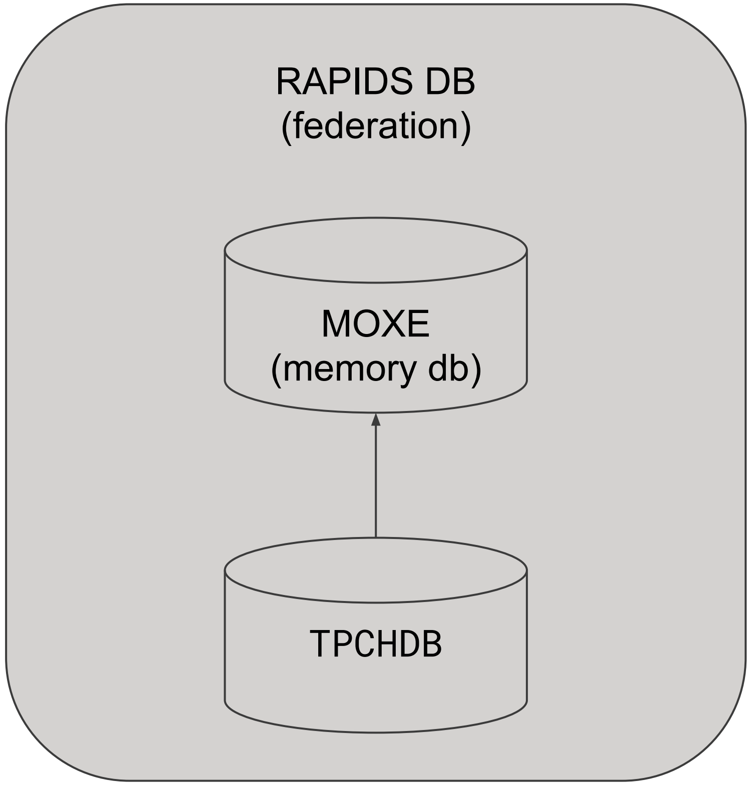
```
create table test (
_id varchar(10),
_index varchar(20),
_score int,
field varchar(100),
message varchar(50),
timest datetime(6),
raw json,
KEY(_id),
KEY(timest)
);

LOAD DATA LOCAL INFILE "infile.json" INTO TABLE test FORMAT JSON (
_id <- _id default NULL,
_index <- _index default NULL,
_score <- _score DEFAULT 0,
@avar <- _source default NULL,
raw <- % default NULL
)
SET
field = json_extract_string(@avar, 'field1'),
message = json_extract_string(@avar, 'message'),
timest = to_date(json_extract_string(@avar, '@timestamp'), 'YYYY-MM-DDTHH24:MM:SS');

select json_extract_string(raw, "_id") from test;
select json_extract_string(raw::_source, "message") from test;

update test set raw = JSON_SET_STRING(raw::_source, "field1", "rice") where _id =
'3sMhlXwBOkvgpBzouRFw';
```



```
infile.json

{
  "_index": "my_index",
  "_type": "_doc",
  "_id": "3sMhlXwBOkvgpBzouRFw",
  "_score": 1,
  "_source": {
    "@timestamp": "2099-11-15T13:12:00",
    "message": "GET /search HTTP/1.1 200 1070000",
    "field1": "kimchy"
  }
}
……
```

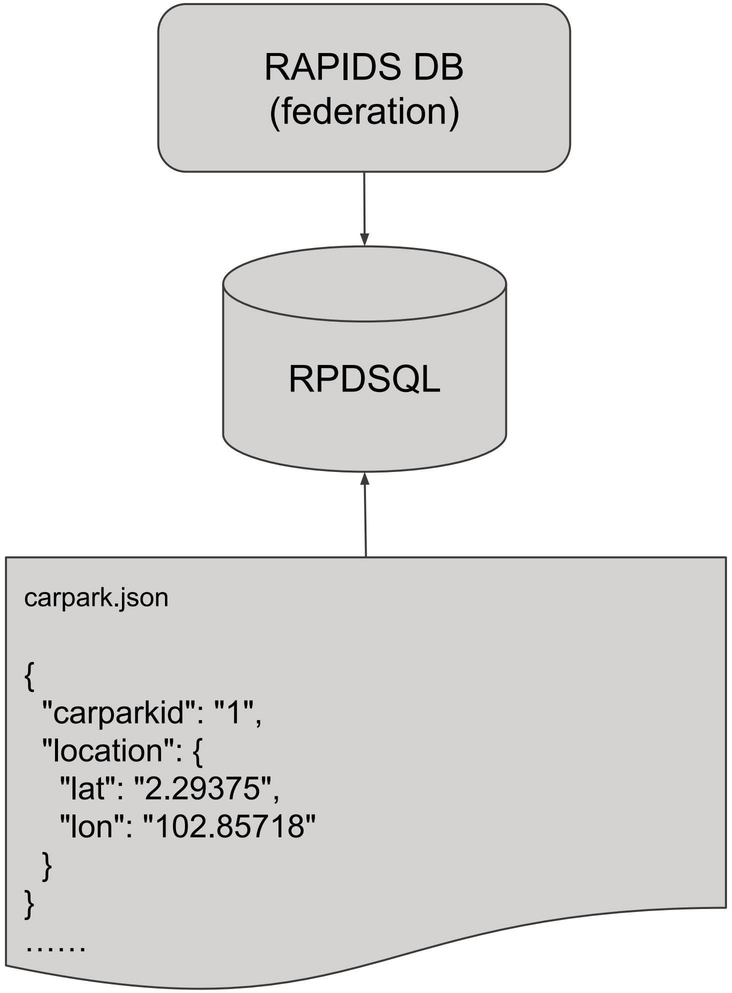
TPCH Query

```
CREATE CONNECTOR SF1 TYPE TPCHDB WITH SCALE_FACTOR='1';

CREATE CONNECTOR MOXE TYPE MOXE WITH
PARTITIONS_PER_NODE='2', MEM_PER_NODE='10GB';

#CREATE TABLES

insert into moxe.lineitem select * from sf1.lineitem;
insert into moxe.orders select * from sf1.orders;
insert into moxe.partsupp select * from sf1.partsupp;
insert into moxe.part select * from sf1.part;
insert into moxe.supplier select * from sf1.supplier;
insert into moxe.customer select * from sf1.customer;
insert into moxe.nation select * from sf1.nation;
insert into moxe.region select * from sf1.region;

#RUN SQL QUERY
```



Python Code Example

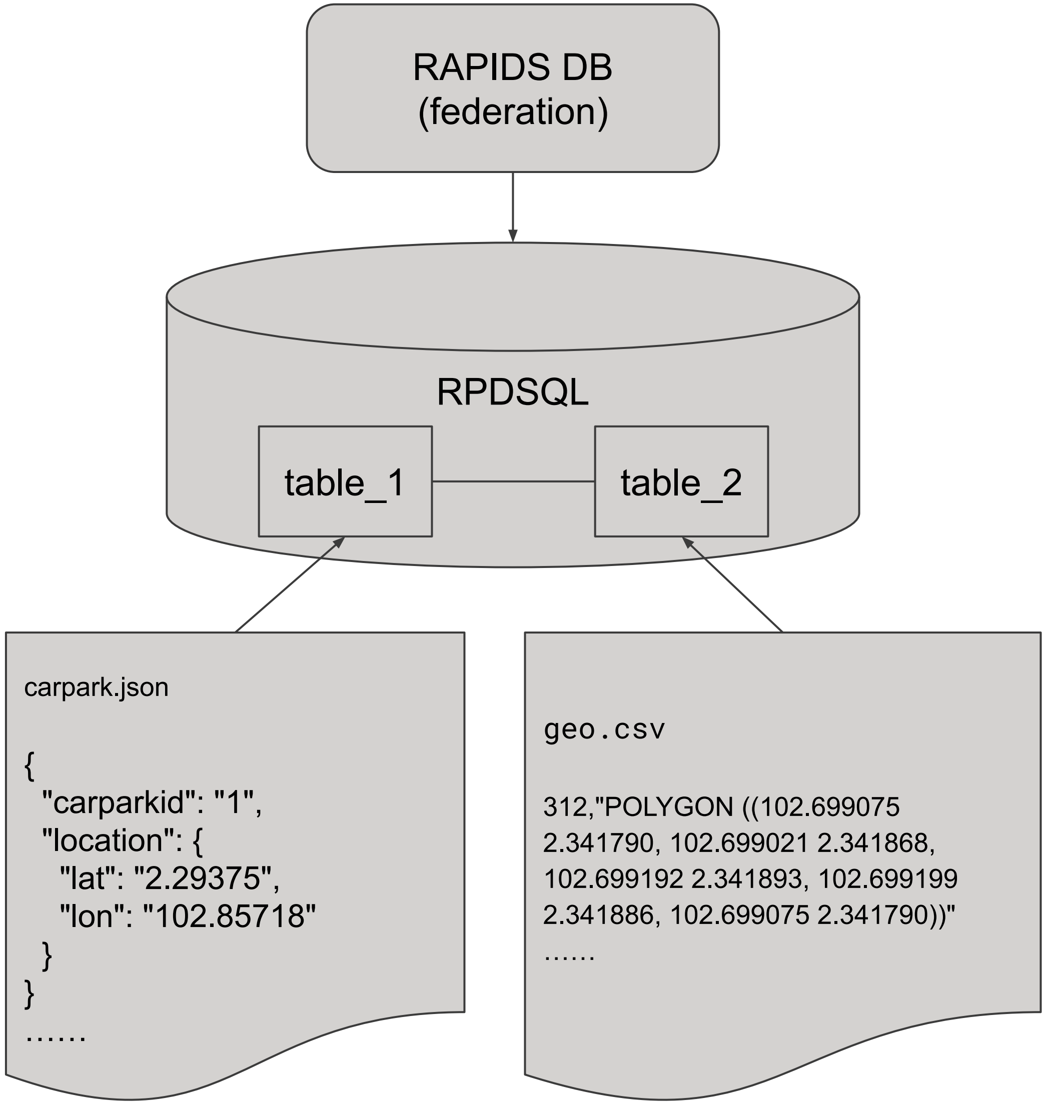
```
#import pyRDP
import pyRDP as pyRDP
import json

#open json file
with open('carpark.json') as f:
  data = json.load(f)

#open connections
conn = pyRDP.connect(host = "domain or ip", port = 4333, user = 'RAPIDS',
password = 'rapids', catalog = 'connector_name', schema = "database_name")
cursor = conn.cursor()

#process data
for i in data:
    carparkid = i["carparkid"]
    lat = i["location"]["lat"]
    lon = i["location"]["lon"]

#execute SQL
sql = "INSERT INTO table_name (carparkid, location, lat, lon) VALUES
('"+carparkid+"', 'POINT("+lon+" "+lat+")', "+lat+", "+lon+")"
cursor.execute(sql)

#close connection
conn.close()
```



```
{
  "carparkid": "1",
  "location": {
    "lat": "2.29375",
    "lon": "102.85718"
  }
}
……
```

GEO Polygon Query

```
create table table_1 (
  carparkid varchar(10),
  location GEOGRAPHYPOINT,
  lat double,
  lon double,
  index(location)
);

create table table_2 (
  FEATID varchar(100) default null,
  GEOMETRY GEOGRAPHY default null,
  index (GEOMETRY) with (resolution = 8),
  index (FEATID)
);

load data local infile "geo.csv" into table table_2 FIELDS TERMINATED BY
'|' ENCLOSED BY '"';

SELECT carparkid, location FROM table_1 WHERE
ROUND(GEOGRAPHY_DISTANCE("POINT(2.85718 3.29375)", location), 0) < 5000;

SELECT c.carparkid, h.FEATID FROM table_1 c, table_2 h WHERE
GEOGRAPHY_CONTAINS(h.GEOMETRY, c.location);
```



carpark.json

```
{
  "carparkid": "1",
  "location": {
    "lat": "2.29375",
    "lon": "102.85718"
  }
}
……
```

geo.csv

```
312,"POLYGON ((102.699075
2.341790, 102.699021 2.341868,
102.699192 2.341893, 102.699199
2.341886, 102.699075 2.341790))"
……
```

Create index

```
CREATE INDEX city_index ON
Persons(city(255));
```

```
ALTER table Persons ADD INDEX
address_index(Address);
```

```
CREATE TABLE mytable(
  ID INT NOT NULL,
  username VARCHAR(16) NOT NULL,
  INDEX username_index (username(16))
);
```

```
CREATE TABLE articles (
  id INT ,
  year INT,
  KEY (year) USING HASH
);
```

```
CREATE TABLE articles2 (
  id INT ,
  year INT,
  KEY (id) USING BTREE
);
```

```
CREATE TABLE articles3(
  id INT ,
  year INT,
  KEY (id) USING CLUSTERED COLUMNSTORE,
  UNIQUE KEY (id) UNENFORCED RELY
);
```

```
CREATE TABLE articles4 (
  id INT ,
  year INT ,
  time DATE,
  PRIMARY KEY (id)
);
```

Data distribute and diagnostic

```
create database web partitions = 100;

CREATE TABLE urls (
  id BIGINT,
  domain_id BIGINT,
  path VARCHAR(8192),
  first_seen INT UNSIGNED NOT NULL,
  crawl count INT UNSIGNED NOT NULL,
  SHARD KEY (domain_id)
);


show partitions on tpch;

rebalance partitions on tpch;
```

```
EXPLAIN SELECT * FROM table;

stats select count(*) as number from moxe.nation;

SHOW VARIABLES LIKE "maximum_%";

SHOW STATUS EXTENDED LIKE "Alloc_table_memory";


SELECT DATABASE_NAME, TABLE_NAME, ORDINAL AS PARTITION_ID, ROWS,
MEMORY_USE FROM INFORMATION_SCHEMA.TABLE_STATISTICS WHERE
TABLE_NAME = 'table';
```

User management

rapidsdb

```
create user bigdata password 'bigdata';

create user bigdata password 'bigdata';

drop user bigdata ;
```

rpdsql

```
create user foo identified by '123456';

show users;

GRANT ALL ON dbname.* TO foo@'10.8.0.5' IDENTIFIED BY 'my_passwd';
```

Data backup and restore

rapidsdb

```
escape connector moxe unload with filter='MOXE.MOXE.SUPPLIER',
name='supplier_2019_10_31', path='/home/rapids/moxe';

escape connector moxe reload with name='supplier_2019_10_31', path='/home/rapids/moxe';
```

rpdsql

```
BACKUP [DATABASE] db_name [WITH { INIT | DIFFERENTIAL | SPLIT PARTITIONS [BY
2] }] TO "backup_path" [(OPTION resource_pool = resource_pool_name)]


RESTORE [DATABASE] db_name FROM "backup_path" [WITH FILE incr_backup_id]
[sync_options] [(OPTION resource_pool = resource_pool_name)]
```

Procedure and Transaction

```
CREATE TABLE t(
  id INT PRIMARY KEY,
  str VARCHAR(255)
);

CREATE TABLE t_errors(
  dt DATETIME,
  id INT,
  str VARCHAR(255)
);

CALL insert_handle_exception(1, "baz");
```

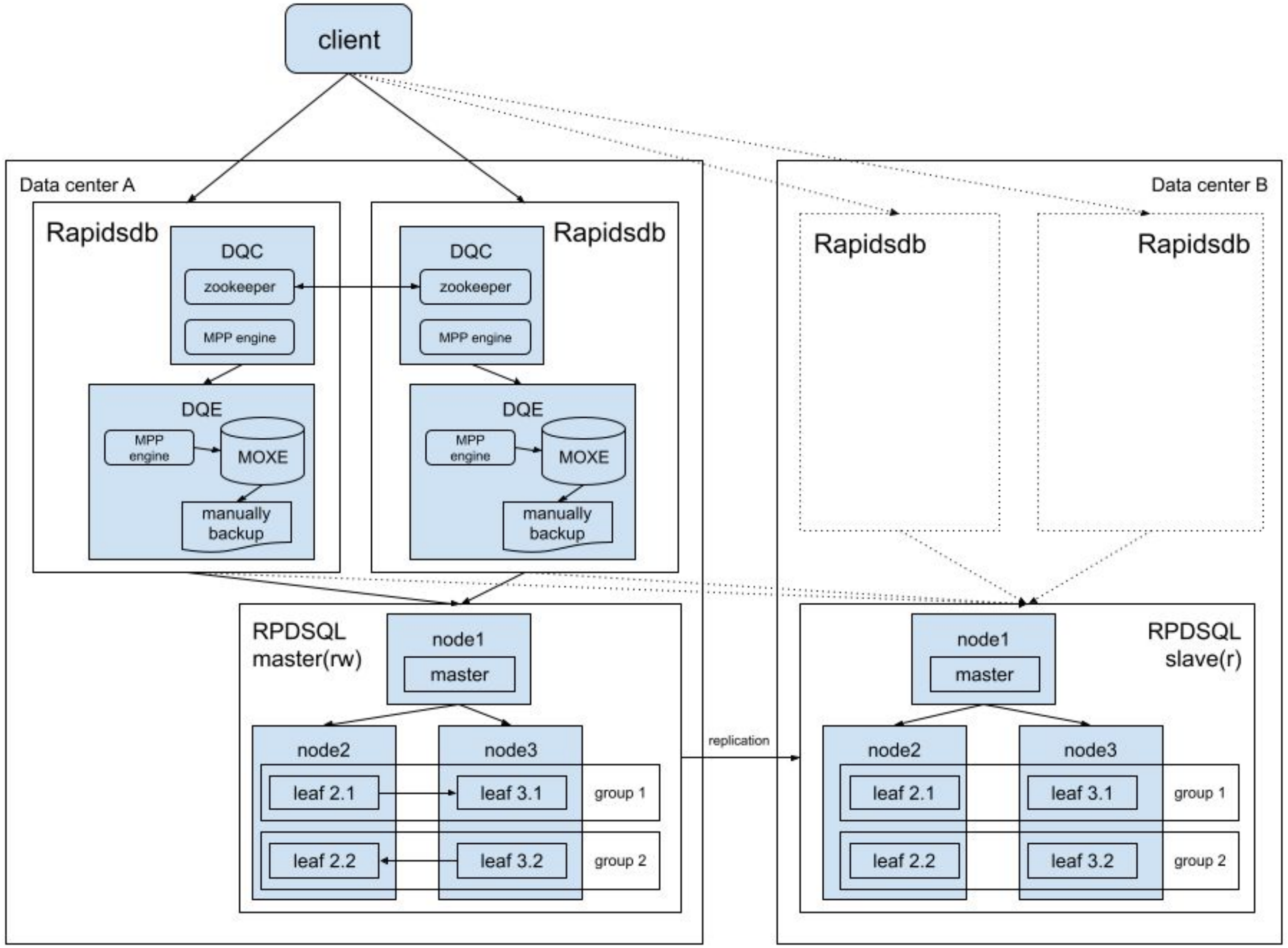
```
DELIMITER //
CREATE OR REPLACE PROCEDURE insert_handle_exception(
  id INT, str VARCHAR(255)) AS
  BEGIN
    START TRANSACTION;
    INSERT INTO t VALUES (1, str);
    COMMIT;

    EXCEPTION
      WHEN ER_DUP_ENTRY THEN
        ROLLBACK;
        START TRANSACTION;
        INSERT INTO t_errors VALUES (current_timestamp(), id, str);
        COMMIT;
  END //
DELIMITER ;
```

Federation table join

```
use connector RPDSQL_TEST;

CREATE TABLE Orders (OrderID INT UNSIGNED NOT NULL DEFAULT 0, CustomerID INT UNSIGNED NOT NULL DEFAULT 0, OrderDate DATE);

insert into Orders (OrderID, CustomerID, OrderDate) VALUES (10308, 2, '1996-09-18');
insert into Orders (OrderID, CustomerID, OrderDate) VALUES (10309, 37, '1996-09-19');
insert into Orders (OrderID, CustomerID, OrderDate) VALUES (10310, 77, '1996-09-20');
select * from Orders;

use connector rapids;

CREATE TABLE Customers (CustomerID INT UNSIGNED NOT NULL DEFAULT 0, CustomerName VARCHAR(50) NOT NULL, ContactName VARCHAR(50) NOT NULL,
Country VARCHAR(50) NOT NULL);

insert into Customers (CustomerID, CustomerName, ContactName, Country) VALUES (1, 'Alfreds', 'Maria Anders', 'Germany');
insert into Customers (CustomerID, CustomerName, ContactName, Country) VALUES (2, 'Ana', 'Ana Trujillo', 'Mexico');
insert into Customers (CustomerID, CustomerName, ContactName, Country) VALUES (3, 'Antonio', 'Antonio Moreno', 'Mexico');

use connector rapids;
refresh;
show tables;

select o.CustomerID, o.CustomerName,c.CustomerID from AI.demo.Customers o join RPDSQL_TEST.rpdsql_test.Orders c on o.CustomerID=c.CustomerID;
```

High availability

```
show leaves;

remove leaf '192.168.0.88':3308;

SET @@GLOBAL.redundancy_level = 2;

ADD LEAF root@'192.168.0.89':3308 INTO GROUP 2;
```



Thank You !

RapidsDB

A BORRUI DATA COMPANY

Intelligent Data, Enabling Future !

doc.rapidsdb.sg

Please email your questions to: yylai@rapidsdb.sg

@RapidsDB