**Using Amazon Redshift and Amazon S3 to deploy a Data Warehouse**

IAM Role ARN:arn:aws:iam::434051408026:role/myRedshiftRole

**COPY IN REDSHIFT:**

copy users from 's3://bucket-viriginia-1/allusers\_pipe.txt'

credentials 'aws\_iam\_role=arn:aws:iam::434051408026:role/myRedshiftRole'

delimiter '|' region 'us-east-1';

copy venue from 's3://bucket-viriginia-1/venue\_pipe.txt'

credentials 'aws\_iam\_role=arn:aws:iam::434051408026:role/myRedshiftRole'

delimiter '|' region 'us-east-1';

copy category from 's3://bucket-viriginia-1/category\_pipe.txt'

credentials 'aws\_iam\_role=arn:aws:iam::434051408026:role/myRedshiftRole'

delimiter '|' region 'us-east-1';

copy date from 's3://bucket-viriginia-1/date2008\_pipe.txt'

credentials 'aws\_iam\_role=arn:aws:iam::434051408026:role/myRedshiftRole'

delimiter '|' region 'us-east-1';

copy event from 's3://bucket-viriginia-1/allevents\_pipe.txt'

credentials 'aws\_iam\_role=arn:aws:iam::434051408026:role/myRedshiftRole'

delimiter '|' timeformat 'YYYY-MM-DD HH:MI:SS' region 'us-east-1';

copy listing from 's3://bucket-viriginia-1/listings\_pipe.txt'

credentials 'aws\_iam\_role=arn:aws:iam::434051408026:role/myRedshiftRole'

delimiter '|' region 'us-east-1';

copy sales from 's3://bucket-viriginia-1/sales\_tab.txt'

credentials 'aws\_iam\_role=arn:aws:iam::434051408026:role/myRedshiftRole'

delimiter '\t' timeformat 'MM/DD/YYYY HH:MI:SS' region 'us-east-1';

**CREATE IN REDSHIFT:**

create table users(

userid integer not null distkey sortkey,

username char(8),

firstname varchar(30),

lastname varchar(30),

city varchar(30),

state char(2),

email varchar(100),

phone char(14),

likesports boolean,

liketheatre boolean,

likeconcerts boolean,

likejazz boolean,

likeclassical boolean,

likeopera boolean,

likerock boolean,

likevegas boolean,

likebroadway boolean,

likemusicals boolean);

create table venue(

venueid smallint not null distkey sortkey,

venuename varchar(100),

venuecity varchar(30),

venuestate char(2),

venueseats integer);

create table category(

catid smallint not null distkey sortkey,

catgroup varchar(10),

catname varchar(10),

catdesc varchar(50));

create table date(

dateid smallint not null distkey sortkey,

caldate date not null,

day character(3) not null,

week smallint not null,

month character(5) not null,

qtr character(5) not null,

year smallint not null,

holiday boolean default('N'));

create table event(

eventid integer not null distkey,

venueid smallint not null,

catid smallint not null,

dateid smallint not null sortkey,

eventname varchar(200),

starttime timestamp);

create table listing(

listid integer not null distkey,

sellerid integer not null,

eventid integer not null,

dateid smallint not null sortkey,

numtickets smallint not null,

priceperticket decimal(8,2),

totalprice decimal(8,2),

listtime timestamp);

create table sales(

salesid integer not null,

listid integer not null distkey,

sellerid integer not null,

buyerid integer not null,

eventid integer not null,

dateid smallint not null sortkey,

qtysold smallint not null,

pricepaid decimal(8,2),

commission decimal(8,2),

saletime timestamp);

**INSERT IN REDSHIFT:**

insert into shoes values

('loafers', 'brown'),

('sandals', 'black');

**SELECT IN REDSHIFT:**

-- Get definition for the sales table.

SELECT \*

FROM pg\_table\_def

WHERE tablename = 'sales';

-- Find total sales on a given calendar date.

SELECT sum(qtysold)

FROM sales, date

WHERE sales.dateid = date.dateid

AND caldate = '2008-01-05';

-- Find top 10 buyers by quantity.

SELECT firstname, lastname, total\_quantity

FROM (SELECT buyerid, sum(qtysold) total\_quantity

FROM sales

GROUP BY buyerid

ORDER BY total\_quantity desc limit 10) Q, users

WHERE Q.buyerid = userid

ORDER BY Q.total\_quantity desc;

-- Find events in the 99.9 percentile in terms of all time gross sales.

SELECT eventname, total\_price

FROM (SELECT eventid, total\_price, ntile(1000) over(order by total\_price desc) as percentile

FROM (SELECT eventid, sum(pricepaid) total\_price

FROM sales

GROUP BY eventid)) Q, event E

WHERE Q.eventid = E.eventid

AND percentile = 1

ORDER BY total\_price desc;