**DB server**

* Access the server box
  + As the installation README instructed Postgres Server was installed with ownership of ‘postgres’, a linux user created without password requirement, to access the server the following procedure needs to be followed:

1. ssh to the server box as user romserver
2. su postgres, to switch to user postgres

* Shutdown/Restart the server process (need to run as postgres!)
  + Native Commands
    - Shutdown: /usr/local/pgsql/bin/pg\_ctl –D /usr/local/pgsql/data stop
    - Start: /usr/local/pgsql/bin/pg\_ctl -D /usr/local/pgsql/data start > /usr/local/pgsql/data/logfile 2>&1
  + The configuration file: /usr/local/pgsql/data/postgresql.conf. For details of the available parameters and their default values checkout the postgresql document and the comments in postgresql.conf .
  + A script named pgsql\_start(‘su postgres’ is already taken care of) used for starting pgsql server is included in *“/etc/init.d”* directory so that pgsql server will be automatically started when the box is rebooted. In the case pgsql server has to be started manually just run pgsql\_start at command line.
* Do the work
  + Sql query: psql ‘targeted database’ to get the sql shell prompt. Every sql statement should be terminated by ‘;’ before you hit enter. To leave the psql shell simply type \q then enter.
  + Dump, Restore…: run pg\_dump, pg\_restore … from linux command line with proper options. For details regarding the available options you may either checkout the postgresql online document or the man page

**Purge Process**

* The criteria for orders to be purged
  + dest\_session table defines the session duration for every exchange. Any exchange destination not found in this table will following the entry with dest\_id=-1
  + Any noneGTC/GTC orders and dead GTC/GTD orders entered before the corresponding current exchange destination session defined in dest\_session table will be purged.
  + Current cron job for running purge script is set for 11:50pm eastern time every day.
  + The purge logic does not assume when the script can be run.
* The scripts
  + /home/postgres/script/purge/**purge.sql** contains all the sql statements which do the job
  + /home/postgres/script/purge/**purge.sh** sets the locations for the data to be saved, invokes purge.sql and sends out an email to report the status of the purge.
  + The historic data file is saved as /home/postgres/db/OrderHistory/YYYY/YYYYMMDD/OrderHistory\_YYYYMMDD.out.gz. where db=//chicagonas1/dart.
  + If copying the above .gz file failed the file will be saved locally in /home/postgres/orderhistory and a note will be added to the email

**Backup**

* rom database:
  + The script /home/postgres/script/backup\_restore/**backup\_rom.sh** is scheduled to run at 11:58pm eastern time every day
  + The output file has the schemas with data. It is saved as /home/postgres/db/rom\_backup/YYYYMMDD.data
  + Currently the script cleans up the output files older than 30 days.
* The whole DB(schemas/data from all databases on that cluster)
  + The script /home/postgres/script/backup\_restore/**backup\_all.sh** is scheduled to run at 11:40pm eastern time every Saturday.
  + The output file has the schemas with data. It is saved as /home/postgres/db/all\_backup/YYYYMMDD.gz

**Restore**

* rom database
  + The script: /home/postgres/script/backup\_restore/restore\_rom.sh. It takes the date in YYYYMMDD format as its first input argument and table name as its second input argument. When no table name is provided the whole rom database will be restored.
    - Examples
      * To restore orders table in rom database from 20180415: restore\_rom.sh 20180415 orders
      * To restore the rom database from 20181020: restore\_rom.sh 20181020
  + After this script is run the rom database will have the schemas of the date YYYYMMDD right before backup\_rom.sh was run and the data of the same date right after purge.sh was run on that day.
* The whole DB
  + The script: /home/postgres/script/backup\_restore/restore\_all.sh. It takes the date in YYYYMMDD format as its input.
  + After this script is run the DB cluster will have the schemas of the date YYYYMMDD right before backup\_all.sh was run and the data of the same date right before purge.sh was run on that day.

**Build a New DB server**

* Before install DB server
  + Create a linux user ‘postgres’ with command: adduser postgres. Note that no password should be required for postgres!
  + Mount network driver
    - Create directory /home/postgres/db
    - Add the following one line in /etc/fstab(sudo needed): //chicagonas1/dart/DART\_Administration/Dart\_Operations/Operations/DB/ /home/postgres/db cifs defaults,credentials=/home/postgres/.smbcredentials,rw,uid=/postgres,mfsymlinks
    - Run(sudo needed): mount –a
  + Install scripts for purge/backup/restore/load\_hist…
    - sudo su postgres to switch to user postgres
    - cp ~/db/script.gz ~/.
    - cd ~/
    - tar xpvzf ./script.gz
* Install postgresql DB server
  + Method1
    - Get the installation file /home/postgres/db/image/postgresql-9.5.5.tar.bz2 and unzip it
    - Step by step following the instructions in the file INSTALL
    - Start the DB server
    - Restore the latest whole DB
    - Restore the latest rom database
    - Restore or truncate the orders table in rom database
  + Method2
    - tar the directory tree of the postgres server installation from the DB server to be replaced if this has not been done yet
      * logon to the DB server as user romserver
      * cd /usr/local
      * sudo tar cvpzf /home/postgres/db/image/psql\_server.gz ./pgsql
      * Actually in order to handle the failover case we should always keep a copy of psql\_server.gz
    - Install DB server
      * Logon to the new DB server as user romserver
      * cd /usr/local
      * sudo tar xpvzf /home/postgres/db/image/psql\_server.gz
    - Start DB server
    - Restore the latest whole DB
    - Restore the latest rom database
    - Restore or truncate the orders table in rom database

**Load History**

* A separate database orderhistory has been created on dev DB server for constructing order history data to be analyzed. The table to hold the data is orders\_hist
* The script /home/postgres/script/load\_hist/load\_hist.sh reads the load\_hist.dat in the same directory to load the date range. For example

20181005

20181007

In the load\_hist.dat. After load\_hist.sh is run the orders\_hist table

will have the data from 20181005 to 20181007.

**\***The above YYYYMMDD is in the time zone where the purge.sh, backup\_rom.sh or backup\_all.sh are run