PVSS ONLINE to OFFLINE Streams replication tests for ATLAS

People involved

- James Cook - ATLAS online
- Florbela Tique Aires Viegas - ATLAS offline
- Gancho Dimitrov - ATLAS offline
- Luca Canali - IT/PSS
- Dawid Wojcik - IT/PSS

Installation parameters and postinstallation steps

- preinstall - modified storage parameters in RDB_config.sql:

```
define initial_size        = 1024
define next_size           = 1024
define storage_clause      = '(INITIAL 100M MINEXTENTS 1 MAXEXTENTS UNLIMITED)'
define storage_clause_idx  = '(INITIAL 100M MINEXTENTS 1 MAXEXTENTS UNLIMITED)'
```

- postinstall - sqlplus:

```
update <ACCOUNT_NAME>.ARC_GROUP set MAX_SIZE_MB=102400;
update <ACCOUNT_NAME>.ARC_CONFIG set value=102400 where name='def_max_size_mb';
update <ACCOUNT_NAME>.ARC_GROUP set MAX_ONLINE=2;
update <ACCOUNT_NAME>.ARC_CONFIG set value=2 where name='def_max_online';
```

Tests description

- All tests have been conducted using PVSS 3.6 SP1 Oracle schemas
- Tests involved creating PVSS (3.6 SP1) account on ATONR cluster databse and setting up streams replication to INTR database
- Test data generated by James Cook is inserted into ATLAS_PVSS_ONL @ ATONR and replicated to ATLAS_PVSS_ONL @ INTR

Test plan

- test maximal sustainable throughput and evaluate whether we can achieve 3-5 GB of PVSS data per day - **COMPLETED**
  - ♦ achieved sustained rate was around 1600 LCRs/s that resulted in around 3-4Gb of PVSS data per day
  - ♦ increasing rate up to 2000 LCRs/s caused high CPU consumption on the APPLY side and lag to grow from 0 to 4 min. during 20 min. of accumulation

- test reading from a replicated set of data using PVSS client - **COMPLETED**
  - ♦ all data accessible through PVSS client

- write large amount of data thus forcing tablespace switch to occur - **COMPLETED**
  - ♦ Florbela prepared a ddl_handler for streams to handle tablespace creation
  - ♦ tests successful - tablespace created on the apply side

- test PVSS client against the replicated data again (after the switch has occured) - **COMPLETED**
  - ♦ data accessible through PVSS client

- create additional indexes on the offline replica and repeat performance tests - **IN PROGRESS**

- index created on the APPLY side on atlas_pvss_onl.EVENTHISTORY_00000003 (element_id, value_number,ts)
- one off problem - poor performance on the APPLY side (around 120 LCRs/s) either with parallelism set to 4 and 1, after dropping the index, performance went back to 1600 LCRs/s, index recreated with only 2 columns (element_id, value_number) lead to the same performance issues. It appeared that the problem of the poor APPLY performance was due to the fact that VALUE_NUMBER column was of type BINARY_DOUBLE. After changing it to BINARY_FLOAT the APPLY achieved faster rate - 1000 LCRs/s. Now **the problem has gone**.
- after repeating the test the performance with index on 3 columns the performance is good - around 1000 LCRs/s - sustainable

- conduct streams ramp-up tests of sustained 3GB/day rate with a ramp-up of 6GB/day for a short period of time - **PENDING**
  - ...

- create additional indexes on the offline replica and repeat ramp-up tests

---

This topic: PSSGroup > PVSSStreamsTests
Topic revision: r7 - 2007-07-20 - DawidWojcik