DB2 LUW 10.1: Cool Stuff No One Else is Talking About

Iqbal Goralwalla
DB2 RUG BeLux
June 6, 2013
About Me

• Head of DB2 Midrange (DB2 on LUW) Service Delivery at Triton Consulting
• Principal Consultant on DB2 LUW
• Experience of DB2 LUW since DB2 Common Server
• IBM Champion for Data Management
• Tendency to talk too much!
DB2 LUW 10.1: Cool Stuff No One Else is Talking About

- Overview of Big Ticket Items
- Gems in an Edition?
- Reclaimable Storage – A Revisit
  - Insert Time Clustered Tables
  - Reclaiming space from Indexes
- Ingest
- Tracking Changes on Database Server
DB2 LUW 10.1: Cool Stuff No One Else is Talking About – more!

- Event Monitors
- Usage Lists
- Backup & Recovery – Cool Enhancements
- Storage Groups
- Monitoring Routines
- Other cool stuff
DB2 LUW 10.1: Big Ticket Items

Low Operational Costs
- Faster Query Response
- Improved Index Mgmt
- Adaptive Compression
- Multi-Temperature Data Management
- Real-time Data Warehousing

Ease of Development
- SQL Compatibility enhancements
- Graph Store
- Row and Column Access Control
- Temporal Capabilities

Reliability
- DB2 pureScale enhancements
- Workload Management Enhancements
- HADR Supports Multiple Standby Servers
Gems in an Edition?

<table>
<thead>
<tr>
<th>Functionality</th>
<th>DB2 Personal</th>
<th>DB2 Express-C</th>
<th>Express (Incl. FTL)</th>
<th>Workgroup Server</th>
<th>Enterprise Server</th>
<th>Advanced Enterprise Server</th>
<th>Enterprise Developer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced Copy Services</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Adaptive Compression and classic row compression</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Compression: backup</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>DB2 Storage Optimization Feature</td>
<td>Yes</td>
</tr>
<tr>
<td>Continuous Data Ingest</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>DB2 pureScale functionality</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes (10 licenses)</td>
<td>Yes</td>
</tr>
<tr>
<td>Federation with DB2 LUW and Informix Data Server</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Federation with DB2 LUW and Oracle</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>High availability disaster recovery (HADR)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>IBM Data Studio</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IBM InfoSphere® Optim™ Configuration Manager</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>IBM InfoSphere Optim Performance Manager Extended</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>IBM InfoSphere Data Architect</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>IBM InfoSphere Optim Query Workload Tuner</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>LBAC / RCAC</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Materialized query tables (MQTs)</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Multidimensional clustering (MDC) tables</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Multi-Temperature Storage</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Online reorganization</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Oracle Compatibility</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>purXML® storage</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Q Replication with two other DB2 LUW servers</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Query parallelism</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Replication tools</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>SQL Replication between DB2 LUW and Informix</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Time Travel Query</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Table partitioning</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Tivoli® System Automation</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Workload management</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

• The use of DB2 pureScale feature is included in DB2 WSE.
  ▪ It is restricted to a total of 16 processor cores and 64 GB of memory across all servers in the cluster.
  ▪ These limits do not apply to the Cluster Caching Facility (CF).

• Data Federation with DB2 AESE Restricted Use InfoSphere Federation Server includes access to Oracle!
  ▪ For example, you can use this Oracle access capability to ease migration of the Oracle data into DB2. Phased migration.
  ▪ New applications can be coded as if all of the data already resided in DB2 whether you had moved it all there yet or not.
  ▪ In addition, DB2’s load from cursor can quickly populate the new DB2 database tables from the Oracle data.

• Restricted use Q-replication functionality to replicate DB2 LUW to 2 DB2 LUW servers for Active-Active environments with DB2 AESE
Deleted thousands of rows using batch delete

Free unused space cannot be used by any other object in the same table space after the batch deletes are complete

Need to reorg the table in order to release space to the tablespace
Consider – Insert Time Clustered Tables

• Rows clustered by ‘insert time’
• Very predominant usage pattern: rows inserted together are often deleted together
• Results in many extents naturally becoming empty after deletions
• ITC tables cluster data by using a virtual column which clusters rows that are inserted at similar time, together:
  ▪ CREATE TABLE ... ORGANIZE BY INSERT TIME ...
ITC Tables – Reclaiming Space

• Monitor amount of reclaimable space of *data portion* using the ADMIN_GET_TAB_INFO_V97 function

  SELECT DATA_OBJECT_L_SIZE, DATA_OBJECT_P_SIZE, RECLAIMABLE_SPACE
  FROM TABLE (SYSPROC.ADMIN_GET_TAB_INFO_V97('IG', 'T1'))

• Reclaim data space using:

  REORG TABLE IG.T1 RECLAIM EXTENTS ALLOW WRITE ACCESS
ITC Tables – Reclaiming Space

• Monitor amount of reclaimable space of *index portion* using the ADMIN_GET_INDEX_INFO function

  SELECT INDNAME, IID, INDEX_OBJECT_L_SIZE, INDEX_OBJECT_P_SIZE, RECLAIMABLE_SPACE
  FROM TABLE(SYSPROC.ADMIN_GET_INDEX_INFO(‘’, ‘IG’,’T1‘)) AS INDEXINFO WHERE INDNAME='IDX1'

• Reclaim index space using:

  REORG INDEXES ALL FOR TABLE IG.T1 ALLOW WRITE ACCESS
  CLEANUP ALL RECLAIM EXTENTS
ITC Tables – Usage Considerations

• Typical scenario/script:
  1. INSERTS ...
  2. DELETE WHERE ...
  3. REORG TABLE ... RECLAIM EXTENTS ...
  4. REORG INDEXES ALL FOR TABLE ... RECLAIM EXTENTS ...
  5. INSERTS ...

• Existing tables cannot be ALTERED to become ITC tables
  ▪ Export/Import or Load from a table can be used to convert existing tables to ITC tables
  ▪ Or ...

The Information Management Specialists
DB2 Geek T-shirt moment!

Win a limited edition 30th Anniversary DB2 Geek T-Shirt for answering this question!

- ? can be used to convert existing tables to ITC tables
  - Hint: Think Swiss-Army Knife!
Consider – Reclaiming unused space in indexes for DMS/AS tablespaces

• Use the ADMIN_GET_INDEX_INFO table function to check indexes for reclaimable space

• RECLAIMABLE_SPACE column provides an estimate of disk space (in KB) that can be reclaimed from the entire index object
Consider – Reclaiming unused space in indexes for DMS/AS tablespaces

• Use new online index reorg functionality to reclaim unused index space in DMS table spaces:
  ▪ `REORG INDEX FOR TABLE ... RECLAIM EXTENTS ...`  
  ▪ `REORG INDEXES ALL FOR TABLE ... RECLAIM EXTENTS ...`

• The space in question is reclaimed back to the table space for reuse by other objects.

• Use `ADMIN_GET_INDEX_INFO` again
DB2 Geek T-shirt moment!

Win a limited edition 30th Anniversary DB2 Geek T-Shirt for answering this question!

• Is the reclaimed space in the index tablespace always returned to the system for reuse?
Consider – Lowering HWM of index tablespace

```
select substr(TBSP_NAME,1,14) as TS_NAME ,
        TBSP_TOTAL_PAGES as Total_pages,
        TBSP_USED_PAGES as Used_pages,
        TBSP_PAGE_TOP as High_water_MARK ,
        reclaimer_space_enabled
from table ( MON_GET_TABLESPACE ( NULL , -1 ) ) as ts
where TBSP_NAME NOT LIKE ('SYS%') and TBSP_TYPE = 'DMS'
```

<table>
<thead>
<tr>
<th>TS_NAME</th>
<th>TOTAL_PAGES</th>
<th>USED_PAGES</th>
<th>HIGH_WATER_MARK</th>
<th>RECLAIMABLE SPACE_ENABLED</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERSPACE1</td>
<td>8192</td>
<td>3264</td>
<td>3264</td>
<td>0</td>
</tr>
<tr>
<td>TP1DMSH</td>
<td>4608</td>
<td>3120</td>
<td>3120</td>
<td>0</td>
</tr>
<tr>
<td>TP1DMSAD</td>
<td>12000</td>
<td>6400</td>
<td>6400</td>
<td>0</td>
</tr>
<tr>
<td>TP1DMSAI</td>
<td>20000</td>
<td>7040</td>
<td>13984</td>
<td>1</td>
</tr>
<tr>
<td>MDCSP1</td>
<td>8192</td>
<td>5760</td>
<td>5760</td>
<td>1</td>
</tr>
<tr>
<td>MDCSP2</td>
<td>8192</td>
<td>2048</td>
<td>2048</td>
<td>1</td>
</tr>
</tbody>
</table>

6 record(s) selected.

- Use the MON_GET_TABLESPACE monitoring table function
- Look for table spaces with a HWM that is much higher than the number of used pages
- Look for the reclaimable space attribute of ‘1’
Consider – Lowering HWM of index tablespace

• For Automatic Storage table spaces, the REDUCE MAX option of ALTER TABLESPACE can be used to release all unused spaced.

• For DMS, two steps are required to release the unused extents:
  ▪ The High Water can be reduced to match the number of used extents with the LOWER HIGH WATER MARK option.
  ▪ The REDUCE, RESIZE or DROP options can then be used.

• Reclaiming storage is an online operation.
INGEST

• Data Warehouses are becoming more “real-time”

→ New client-side tool for high speed, continuous data ingest

• Continuously pumps data into DB2 tables using SQL arrays until the source is exhausted

• Data can be transformed using SQL expressions (lightweight ETL).

→ Can be used against DB2 10.1, 9.8, 9.7, and 9.5 servers

• Etc.
INGEST – Get started right away!

• Download and install DB2 10.1 client
• Catalog remote DB2 9.8, 9.7, or 9.5 database
• Connect to down-level database from DB2 10.1 client
INGEST – Get started right away!

- db2 "ingest from file myfile.txt format delimited insert into ingest_test"
  - Oops! SQL0805N  Package "NULLID.SQLC2J23 0X4141414141564962" was not found. SQLSTATE=51002
  - db2 "bind @db2ubind.lst blocking all grant public"
  - db2 "bind @db2cli.lst blocking all grant public"
INGEST – Get started right away!

• db2 "ingest from file myfile.txt format delimited insert into ingest_test"
  ▪ Oops! SQL2957N  The ingest utility could not find the restart log table "SYSTOOLS.INGESTRESTART". SQLSTATE=42704
  ▪ Need to manually create the INGESTRESTART table
    ▶ Note: For DB2 10.1 servers, simply call the SYSPROC.SYSINSTALLOBJECTS procedure
INGEST – Get started right away!

CREATE TABLE SYSTOOLS.INGESTRESTART (
    JOBID          VARCHAR(256)   NOT NULL,
    APPLICATIONID  VARCHAR(256)   NOT NULL,
    FLUSHERID      INT             NOT NULL,
    FLUSHERDISTID  INT             NOT NULL,
    TRANSPORTERID  INT             NOT NULL,
    BUFFERID       BIGINT          NOT NULL,
    BYTEPOS        BIGINT          NOT NULL,
    ROWSPROCESSED INT             NOT NULL,
    PRIMARY KEY (JOBID, FLUSHERID, TRANSPORTERID, FLUSHERDISTID))
DISTRIBUTE BY (FLUSHERDISTID);

GRANT SELECT, INSERT, UPDATE, DELETE
    ON TABLE SYSTOOLS.INGESTRESTART TO PUBLIC;
INGEST – Get started right away!

- `db2 "ingest from file myfile.txt format delimited insert into ingest_test"
  - Success this time!
    - SQL2979I The ingest utility is starting at "28/12/2012 15:19:22.927111".
    - SQL2914I The ingest utility has started the following ingest job: "DB21001:20121228.151922.927111:00002:00007".
    - Number of rows read = 10
    - Number of rows inserted = 10
    - Number of rows rejected = 0
Tracking Changes on DB2 Server

• Manually tracking changes to the database server can be daunting (db2diag.log, STMM logs, History file, etc.)

• New History Event Monitor that tracks:
  ▪ DBM and DB parameters changes
  ▪ DB2 Registry variables changes
  ▪ DDL statement execution (what happened to *that* index?)
  ▪ Utility execution
  ▪ Captures initial values (when enabled)
  ▪ Autonomic changes as well!
    ▶ STMM
    ▶ Automatic REORGs/RUNSTATS
Tracking Changes on DB2 Server – Usage

• Decide which change history events are of interest.
• To monitor configuration changes, index changes, and LOAD operations with a change history event monitor:

```
CREATE EVENT MONITOR HIST_CFG
FOR CHANGE HISTORY
WHERE EVENT IN (DBCFG, DBMCFG, DBCFGVALUES, DBMCFGVALUES, REGVAR, REGVARVALUES, DDLDATA, LOAD)
WRITE TO TABLE
```

• Activate event monitor

```
SET EVENT MONITOR HIST_CFG STATE=1
```
Tracking Changes on DB2 Server – Usage

SELECT EVENT_TYPE
FROM
CHANGESUMMARY_HIST_CFG
WHERE EVENT_TIMESTAMP > CURRENT TIMESTAMP - 24 HOURS

SELECT EVENT_TIMESTAMP, CFG_NAME, CFG_VALUE, CFG_OLD_VALUE
FROM DBDBMCFG_HIST_CFG
Event Monitors can now write to formatted tables!

- DB2 9.7 introduced 3 new event monitors
  - Locking
  - Package Cache
  - Unit of Work

- These wrote to Binary “Unformatted Events” (UE) tables
  - Complex post-processing headaches!

- These (plus new Change History) can now also write to standard (formatted) tables – WRITE TO TABLE target
Upgrade event monitor output tables

- EVMON_UPGRADE_TABLES stored procedure
  - upgrades definitions of existing tables along with all data
  - automatically deactivates/reactivates event monitor before/after upgrade
  - Call EVMON_UPGRADE_TABLES (null, null, null, ?, ?, ?)
Usage Lists

• Captures information about DML statement sections which affect a table or index
• Meant for helping identify costly SQL against a table or index
• Create using
  ▪ CREATE USAGE LIST FOR TABLE/INDEX...
• Read using table functions
  ▪ MON_GET_TABLE_USAGE_LIST()
  ▪ MON_GET_INDEX_USAGE_LIST()
Using (!) Usage Lists

1. Identify table that you want to track costly SQL on
   - You might have seen an unusual value for a monitor element on a table using the MON_GET_TABLE table function (e.g., High number of Rows Read)

2. Set the MON_OBJ_METRICS configuration parameter to EXTENDED
   
   ```sql
   UPDATE DATABASE CONFIGURATION USING MON_OBJ_METRICS EXTENDED
   ```

3. Create a usage list for the table
   
   ```sql
   CREATE USAGE LIST T1UL FOR TABLE IG.T1
   ```

4. Activate the usage list
   
   ```sql
   SET USAGE LIST T1UL STATE = ACTIVE
   ```
Using (!) Usage Lists

5. Deactivate the usage list (after collection period)
   
   SET USAGE LIST T1UL STATE = INACTIVE

6. View the information that you collected by using the 
   MON_GET_TABLE_USAGE_LIST function

   SELECT MEMBER, EXECUTABLE_ID, ROWS_READ
   FROM TABLE(MON_GET_TABLE_USAGE_LIST('IG, 'T1UL', -2)) ORDER BY
   ROWS_READ DESC
   FETCH FIRST 10 ROWS ONLY

7. View the text of a statement that affected the table, using 
   EXECUTABLE_ID element as input for the 
   MON_GET_PKG_CACHE_STMT table function.

   SELECT STMT_TEXT
   FROM TABLE
   (MON_GET_PKG_CACHE_STMT(NULL, < EXECUTABLE_ID>, NULL, -2))
Backup & Recovery – Cool Enhancements

• DMS and AS page validation of backup images is now supported during the execution of the BACKUP DATABASE command.
  ▪ Set DB2 registry variable DB2_BCKP_PAGE_VALIDATION to TRUE

• Backups can now succeed even if they do not contain all of the necessary log files
  ▪ Online backups that do not explicitly specify either the INCLUDE LOGS or the EXCLUDE LOGS option fail if all of the logs are not successfully included
  ▪ Set DB2 registry variable DB2_BCKP_INCLUDE_LOGS_WARNING to TRUE
Backup & Recovery – Cool Enhancements

• Backup image file can now be directly written to disk, bypassing the file cache
  ▪ Set DB2 registry variable DB2_BACKUP_USE_DIO to TRUE
  ▪ Note: Although memory utilisation will be better, backup may be a little slower.

• Backup and restore statistics are now automatically generated (db2diag.log) after the successful completion of the operation.
  ▪ Total time of Backup EDU, Time spent in R/W I/O, Time taken for compression, Quantity of uncompressed data, Etc.
Storage paths for Data & Index Tablespaces – the DBA dilemma

In DMS tablespaces, data and index can reside in separate tablespaces and DBA has control on container numbers and location.

In AS tablespaces, data and index can reside in separate tablespaces. However, DBA has no control on container numbers and location.
Storage paths for Data & Index Tablespaces

pre-DB2 10.1 – the DBA dilemma

db2inst1@grampa:~$ db2 "create database IG_AS on /db2data, /home/db2inst1"
DB20000I The CREATE DATABASE command completed successfully.
db2inst1@grampa:~$ db2 connect to IG_AS

      Database Connection Information

Database server       = DB2/LINUXX8664 9.7.5
SQL authorization ID  = DB2INST1
Local database alias  = IG_AS

db2inst1@grampa:~$ db2 "create tablespace data_tbsp"
DB200000I The SQL command completed successfully.
db2inst1@grampa:~$ db2 "create tablespace index_tbsp"
DB200000I The SQL command completed successfully.
db2inst1@grampa:~$ db2pd -db IG_AS -tablespaces
Storage paths for Data & Index Tablespaces pre-DB2 10.1 – the DBA dilemma

<table>
<thead>
<tr>
<th>Address</th>
<th>Id</th>
<th>Type</th>
<th>Content</th>
<th>PageSz</th>
<th>ExtentSz</th>
<th>Auto</th>
<th>Prefetch</th>
<th>BufID</th>
<th>BufIDDisk</th>
<th>FSC</th>
<th>NumCntrs</th>
<th>MaxStripe</th>
<th>LastConsecPg</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x00007F7F9C1A1AA0 3</td>
<td>DMS</td>
<td>Large</td>
<td>4096</td>
<td>32</td>
<td>Yes 384</td>
<td>1</td>
<td>1</td>
<td>Off 2</td>
<td>0</td>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td>DATA_TBSP</td>
</tr>
<tr>
<td>0x00007F7F9C1AB120 4</td>
<td>DMS</td>
<td>Large</td>
<td>4096</td>
<td>32</td>
<td>Yes 384</td>
<td>1</td>
<td>1</td>
<td>Off 2</td>
<td>0</td>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td>INDEX_TBSP</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Containers:

<table>
<thead>
<tr>
<th>Address</th>
<th>TspId</th>
<th>ContainNum</th>
<th>Type</th>
<th>TotalPgs</th>
<th>UseablePgs</th>
<th>PathID</th>
<th>StripeSet</th>
<th>Container</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x00007F7F9C1A3C20 3</td>
<td>0</td>
<td>File</td>
<td>4096</td>
<td>4064</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>/home/db2inst1/db2inst1/NODE0000/IG_AS/T0000003/C0000000.LRG</td>
</tr>
<tr>
<td>0x00007F7F9C1A3E30 3</td>
<td>1</td>
<td>File</td>
<td>4096</td>
<td>4064</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>/db2data/db2inst1/NODE0000/IG_AS/T0000003/C0000001.LRG</td>
</tr>
<tr>
<td>0x00007F7F9C1AD240 4</td>
<td>0</td>
<td>File</td>
<td>4096</td>
<td>4064</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>/home/db2inst1/db2inst1/NODE0000/IG_AS/T0000004/C0000000.LRG</td>
</tr>
<tr>
<td>0x00007F7F9C1AD450 4</td>
<td>1</td>
<td>File</td>
<td>4096</td>
<td>4064</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>/db2data/db2inst1/NODE0000/IG_AS/T0000004/C0000001.LRG</td>
</tr>
</tbody>
</table>
DB2 10.1 – here come Storage Groups!

Partitioned Table Sales

Partition
- 2010Q1
- 2009Q4
- 2009Q3
- 2009Q2
- 2009Q1
- 2008Q4 ...
- 2006Q3

Automatic Storage Table space

Storage Group

New

Table Space
- 14
- 13
- 12
- 11
- 10
- 9 ...
- 1

spath: /hot/fs1

spath: /warm/fs1
spath: /warm/fs2

spath: /cold/fs1
spath: /cold/fs2
spath: /cold/fs3

SG_HOT

SG_WARM

SG_COLD

Physical Disk

SSD RAID Array

FC/SAS RAID Array

SATA RAID Array

The Information Management Specialists
Storage paths for Data & Index Tablespaces – DB2 10.1

```
db2ins10@grampa:/db2data> db2 "create database IG_AS_10"
DB20000I  The CREATE DATABASE command completed successfully.
db2ins10@grampa:/db2data> db2 "connect to IG_AS_10"

Database Connection Information

Database server = DB2/LINUXX8664 10.1.0
SQL authorization ID = DB2INS10
Local database alias = IG_AS_10

db2ins10@grampa:/db2data> db2 "create stogroup sto_data on '/db2data'"
DB20000I  The SQL command completed successfully.
db2ins10@grampa:/db2data> db2 "create stogroup sto_idx on '/home/db2ins10'"
DB20000I  The SQL command completed successfully.
db2ins10@grampa:/db2data> db2 "create tablespace data_tcbsp using stogroup sto_data"
DB20000I  The SQL command completed successfully.
db2ins10@grampa:/db2data> db2 "create tablespace idx_tcbsp using stogroup sto_idx"
DB20000I  The SQL command completed successfully.
```

The Information Management Specialists
Storage paths for Data & Index Tablespaces – DB2 10.1

db2ins10@grampa:/iqbal> db2pd -db IG_AS_10 -storagepaths

Database Member 0 -- Database IG_AS_10 -- Active -- Up 0 days 00:04:25 -- Date 10/01/2013 16:11:51

Storage Group Configuration:

<table>
<thead>
<tr>
<th>Address</th>
<th>SGID</th>
<th>Default</th>
<th>DataTag</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00007F2F6E27A080</td>
<td>0</td>
<td>Yes</td>
<td>0</td>
<td>IBMSTOGROUP</td>
</tr>
<tr>
<td>0x00007F2F751DF940</td>
<td>1</td>
<td>No</td>
<td>0</td>
<td>STO_DATA</td>
</tr>
<tr>
<td>0x00007F2F751DFA60</td>
<td>2</td>
<td>No</td>
<td>0</td>
<td>STO_IDK</td>
</tr>
</tbody>
</table>

Storage Group Statistics:

<table>
<thead>
<tr>
<th>Address</th>
<th>SGID</th>
<th>State</th>
<th>Numpaths</th>
<th>NumDropPen</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00007F2F6E27A080</td>
<td>0</td>
<td>0x00000000</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0x00007F2F751DF940</td>
<td>1</td>
<td>0x00000000</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0x00007F2F751DFA60</td>
<td>2</td>
<td>0x00000000</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Storage Group Paths:

<table>
<thead>
<tr>
<th>Address</th>
<th>SGID</th>
<th>PathID</th>
<th>PathState</th>
<th>PathName</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00007F2F6E27A1A0</td>
<td>0</td>
<td>0</td>
<td>InUse</td>
<td>/home/db2ins10</td>
</tr>
<tr>
<td>0x00007F2F751E14A0</td>
<td>1</td>
<td>1024</td>
<td>InUse</td>
<td>/db2data</td>
</tr>
<tr>
<td>0x00007F2F751DFB80</td>
<td>2</td>
<td>2048</td>
<td>InUse</td>
<td>/home/db2ins10</td>
</tr>
</tbody>
</table>
Storage paths for Data & Index Tablespaces – DB2 10.1

Database Member 0 -- Database IG_AS_10 -- Active -- Up 0 days 00:07:09 -- Date 10/01/2013 16:14:35

Tablespace Configuration:

<table>
<thead>
<tr>
<th>Address</th>
<th>Id</th>
<th>Type</th>
<th>Content</th>
<th>Pagesz</th>
<th>ExtentSz</th>
<th>Auto</th>
<th>Prefetch</th>
<th>BufID</th>
<th>BufIDDisk</th>
<th>FSC</th>
<th>NumCntrs</th>
<th>MaxStripe</th>
<th>LastConsecPg</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00007F2F7520B660</td>
<td>3</td>
<td>DMS</td>
<td>Large</td>
<td>4096</td>
<td>32</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>DATA_TBSP</td>
</tr>
<tr>
<td>0x00007F2F752C7200</td>
<td>4</td>
<td>DMS</td>
<td>Large</td>
<td>4096</td>
<td>32</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>IDX_TBSP</td>
</tr>
</tbody>
</table>

Containers:

<table>
<thead>
<tr>
<th>Address</th>
<th>TspId</th>
<th>ContainNum</th>
<th>Type</th>
<th>TotalPgs</th>
<th>UseablePgs</th>
<th>PathID</th>
<th>StripeSet</th>
<th>Container</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00007F2F751E58C0</td>
<td>3</td>
<td>0</td>
<td>File</td>
<td>8192</td>
<td>8160</td>
<td>1024</td>
<td>0</td>
<td>/db2data/db2ins10/NODE0000/IG_AS_10/T0000003/C0000000.LRG</td>
</tr>
<tr>
<td>0x00007F2F75217C00</td>
<td>4</td>
<td>0</td>
<td>File</td>
<td>8192</td>
<td>8160</td>
<td>2048</td>
<td>0</td>
<td>/home/db2ins10/db2ins10/NODE0000/IG_AS_10/T0000004/C0000000.LRG</td>
</tr>
</tbody>
</table>

The Information Management Specialists
HADR – Log Spooling

• Log replay on Standby may be slow when:
  ▪ Sudden spike in transaction volume on the primary
  ▪ Large reorgs are replayed on the standby

• This can cause delay in processing transactions on the Primary

• Solution – implement Log Spooling
HADR – Log Spooling

- Log data on Standby is spooled/written to disk on the standby if it falls behind in log replay and log buffer fills up
- Standby can read the log data from disk later on
- **HADR_SPOOL_LIMIT** (db cfg)
  - Determines maximum amount of log data allowed to be spooled to disk on HADR standby
  - 0 (default): log spooling disabled
  - -1: no defined limit
HADR – Log Spooling

• Ensure that adequate disk space is provided to the active log path of the standby database (active log space now includes space required by setting HADR_SPOOL_LIMIT)

• Takeover time may be longer as a result of log spooling
  ▪ Replay of the spooled logs has to complete first
Hot off the Press! Monitoring Routines

• What are the most expensive routines?
• What SQL statements were executed by a routine?
• What are the most time consuming SQL statements executed by a routine?
• What routines are invoked when a particular SQL statement is executed?
Hot off the Press! Monitoring Routines

• MON_GET_ROUTINE
• MON_GET_ROUTINE_DETAILS
• MON_GET_ROUTINE_EXEC_LIST
• Notes
  ▪ MON_RTN_DATA and MON_REQ_METRICS database configuration parameters need to be enabled
  ▪ Any routines that were not executed during the previous 24 hour period are pruned from memory and not returned
Other cool stuff

- **ADMIN_MOVE_TABLE**
  - Now supports moving tables with referential integrity constraints
  - REDIRECT option – forwards changes directly to the target table instead of capturing the changes in the staging table

- **Offline Log Archive Compression**
  - `db2 update db cfg using LOGARCHCOMPR1 ON`
  - `db2 update db cfg using LOGARCHCOMPR2 ON`

- **db2pd**
  - -storagepaths (information on storage groups)
  - -hadr (output has completely changed. Existing fields renamed to match output from new MON_GET_HADR table function)
Summary & Conclusions

• DB2 10.1 has lot’s of cool stuff for DBAs that does not necessarily make the headlines

• Keep track of more cool stuff through Fixpaks
Feedback / Questions

Iqbal Goralwalla
iqbal@triton.co.uk
www.triton.co.uk