DB2 LUW 10.5 with BLU Acceleration: First Impressions

Iqbal Goralwalla
DB2 RUG BeLux
June 6, 2013

The Information Management Specialists
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 10.5 with BLU Acceleration: First Impressions

- Overview of BLU Acceleration and Columnar tables
- DB2 10.5 Packaging Simplification
- My 4 Impressions
- Columnar Tables – Guidance & Monitoring
- Summary
Overview of BLU Acceleration and Columnar tables

- New technology for analytic queries in DB2 LUW
- Embedded directly into the DB2 kernel
- New column-organized tables in DB2 databases
  - Table data is stored column organized rather than row organized
  - Using a vector processing engine
- New unique encoding for speed and compression
- New unique runtime technology which leverages the CPU architecture
# DB2 10.5 Packaging Simplification

**One Set of Editions for Both Transactional and Warehouse Workloads**

<table>
<thead>
<tr>
<th>Departmental Market</th>
<th>Enterprise Market</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DB2 Advanced Workgroup Server Edition</strong></td>
<td><strong>DB2 Advanced Enterprise Server Edition</strong></td>
</tr>
<tr>
<td>- For small OLTP and analytic deployments</td>
<td>- For Enterprise Class OLTP and/or analytic deployments</td>
</tr>
<tr>
<td>- Primarily used in department environments within large enterprises or SMB/MM deployments</td>
<td>- Targeting full enterprise/full data centre requirements</td>
</tr>
<tr>
<td>- Limited by TB, memory, sockets and cores</td>
<td>- No TB, memory, socket or core limit</td>
</tr>
<tr>
<td>- Supports BLU, pureScale and DPF deployment models</td>
<td>- Supports BLU, pureScale and DPF deployment models</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>DB2 Workgroup Server Edition</strong></th>
<th><strong>DB2 Enterprise Server Edition</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Entry level offering</td>
<td>- Entry level offering</td>
</tr>
<tr>
<td>- Single server for less intense workloads</td>
<td>- Single server for enterprise/more intense workloads</td>
</tr>
<tr>
<td>- Limited by TB, memory, sockets and cores</td>
<td>- No TB, memory, socket or core limit</td>
</tr>
<tr>
<td>- No support for BLU, pureScale or DPF deployment models</td>
<td>- No support for BLU, pureScale or DPF deployment models</td>
</tr>
</tbody>
</table>

**Limited capacity**

- Down from 13 to 6 editions
- Down to 1 new optional feature – Advanced Recovery Feature

**Full capacity**

- The Information Management Specialists
DB2 LUW 10.5 with BLU Acceleration: First Impressions

- Overview of BLU Acceleration and Columnar tables
- DB2 10.5 Packaging Simplification

My 4 Impressions

- Columnar Tables – Guidance & Monitoring
- BLU Acceleration First Impressions – Summary
Impression #1
Simplicity

The Information Management Specialists
Analytic Database Tuning Complexity!

- Database design and tuning
  - Decide on partition strategies
  - Select compression strategy
  - Create table
  - Load data
  - Create auxiliary performance structures
    - MQTs
    - MDCs
    - Create indexes
      - B+ indexes
      - Bitmap indexes
  - Tune memory
  - Tune I/O
  - Add optimizer hints
  - REORG
  - Statistics collection

Repeat

Is there an end in sight?

The Information Management Specialists
With BLU Acceleration Comes …

Analytic Database Tuning *Simplicity*!

**Database design and tuning**
- Decide on partition strategies
- Select compression strategy
- Create table
- Load data
- Create auxiliary performance structures
  - MQTs
  - MDCs
  - Create indexes
    - B+ indexes
    - Bitmap indexes
- Tune memory
- Tune I/O
- Add optimizer hints
- REORG
- Statistics collection

Repeat
LOAD and GO!

- No indexes
- No REORG (it's automated)
- No RUNSTATS (it's automated)
- No MDC
- No MQTs
- No partitioning
- No statistical views
- No optimizer hints
Getting Started

• Need to be on AWSE or AESE
• Flick on BLU Acceleration with one switch
  ▪ Set DB2 registry variable DB2_WORKLOAD to ANALYTICS
    ▶ db2set DB2_WORKLOAD=ANALYTICS
  ▪ Automatically configures DB2 database for analytic workloads
    ▶ Makes column-organized tables the default table type
    ▶ Enables automatic workload management
    ▶ Enables automatic space reclaim
    ▶ Page and extent size configured for analytics
    ▶ Memory for caching, sorting and hashing, utilities are automatically initialized based on the server size and available RAM
Getting Started

• Recommend setting DB2_WORKLOAD to ANALYTICS *before* creating a new DB2 database with BLU Acceleration
  • When database is created, it will detect this setting and configure itself for analytic processing
  • Personalized to your server – based on your CPU type, memory, and disks.
  • If for some reason you can't or don't want to set this, no need to worry ...
DB2_WORKLOAD=ANALYTICS – behind the scenes

- Page size set to 32 KB (for a *new* database)
- Extent size set to 4
- Default table organization (DFT_TABLE_ORG) db cfg set to COLUMN
- Memory is divided (roughly) equally between the buffer pool and shared sort heap threshold (SHEAPTHRESH_SHR)
- Sort heap (SORTHEAP) is set to a moderate fraction (for example, 5%) of SHEAPTHRESH_SHR
DB2_WORKLOAD=ANALYTICS – behind the scenes

- Intrapartition parallelism is enabled
- DFT_DEGREE set to ANY
- Automatic statistics collection is enabled
- UTIL_HEAP_SZ set to a sizeable value
- Self-tuning memory manager (STMM), buffer pools, the lock list, and package cache are all enabled
  - Note: Sort memory is not part of STMM
  - SHEAPTHRESH_SHR and SORTHEAP should be set to static values
Creating Columnar Tables

• If DB2_WORKLOAD=ANALYTICS or DFT_TABLE_ORG is set to COLUMN, tables will be created column organized automatically

• For mixed table types, define tables as ORGANIZE BY COLUMN or ROW

  CREATE TABLE T1 (
    C1 INTEGER NOT NULL,
    C2 INTEGER
    ...
    PRIMARY KEY (C1)
) ORGANIZE BY COLUMN;
Creating Columnar Tables

• For BLU tables, do not specify:
  ▪ Compression
  ▪ MDC
  ▪ Partitioning
  ▪ MQTs
  ▪ Indexes
    ▶ Except Primary keys and Unique indexes ....

• BLU tables can only be created in Automatic Storage tablespaces
Informational Constraints – Recap

• Mechanism to *inform* the DB2 optimizer about a constraint without having to enforce it in the database.

• *If the data has come from a trusted source,* you could define this as a NOT ENFORCED informational constraint.
  
  ▪ Bypass the database manager overhead of checking constraint
  ▪ Reduce on-disk footprint associated with enforcing the constraint.

• DB2 optimizer could still take advantage of constraint during query rewrite or access plan generation.
With DB2 10.5 comes ... Informational *Uniqueness*!

- **Unique indexes:**
  - Do not compress well
  - Have a large on-disk footprint
  - Must be maintained

- **Option** of specifying Informational uniqueness for BLU tables
  - Primary keys and unique indexes can be enforced or not enforced
    - Default is still to *enforce* uniqueness
  - Similar to informational constraints, less overhead but optimizer is able to leverage information for access plans
  - Make sure your data is indeed unique!
With DB2 10.5 comes ... Informational Uniqueness!

CREATE TABLE T1 (
  C1 INTEGER NOT NULL,
  C2 INTEGER
  ... 
  PRIMARY KEY (C1) NOT ENFORCED) ORGANIZE BY COLUMN;

ALTER TABLE T1
  ADD CONSTRAINT UNQ_C2 UNIQUE (C2) NOT ENFORCED;

• What about Foreign Key constraints?
  ▪ Can only be non-enforced
Converting Existing Tables

- Easily convert tables from row-organized to column-organized
- `db2convert` utility
  - Online
    - Calls `ADMIN_MOVE_TABLE` behind the scenes (same options and restrictions apply)
  - Convert ALL tables in database
    - `db2convert` –d mydb
  - Convert single table (T1 in schema myschema)
    - `db2convert` –d mydb –z db2105 –t T1
Converting Existing Tables

- db2convert utility – things to be aware of: (may be lifted in GA code)
  - INTRA_PARALLEL needs be ON
  - -z parameter for schema is mandatory
  - Non-unique indexes need to be dropped
  - Space temporarily required for both the source and target table
  - No online process to convert a column-organized table back to a row-organized table
    - Perform a backup before converting tables to column organised
    - Unload the column-organized data and then reload it into a new row-organized table
Converting Existing Tables

• db2convert utility – things to be aware of: (may be lifted in GA code)
  ▪ Can be accessed through Optim Data Studio
    ► Right-click a row-organized table and select Migrate to Column Storage option
  ▪ If db2convert fails, syntax of ADMIN_MOVE_TABLE given with error message:
    ► CALL SYSPROC.ADMIN_MOVE_TABLE('DB2105','T1','','','','ORGANIZE BY COLUMN','','','COPY_USE_LOAD,NOT_ENFORCED','COPY')
  ▪ Use SYSTOOLS.ADMIN_MOVE_TABLE to check status
  ▪ CALL SYSPROC.ADMIN_MOVE_TABLE('DB2105','T1','','','','ORGANIZE BY COLUMN','','','CANCEL')
Which tables are Column-organized?

```sql
SELECT
    SUBSTR(TABNAME, 1, 20) AS TABNAME,
    TABLEORG,
    COMPRESSION
FROM SYSCAT.TABLES
WHERE TABNAME LIKE 'T1%'
```

<table>
<thead>
<tr>
<th>TABNAME</th>
<th>TABLEORG</th>
<th>COMPRESSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1_C</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>T1_R</td>
<td>R</td>
<td>N</td>
</tr>
</tbody>
</table>

Blank because compression is always ON for columnar tables. No way to enable/disable.
“Phantom” Columnar Tables?

- A “synopsis” table is automatically created and maintained for each columnar table
  - Used for data elimination/skipping
  - Very small footprint
    - Approximately 0.1% of the user table
    - 1 row for every 1024 rows in the user table

<table>
<thead>
<tr>
<th>TABNAME</th>
<th>TABLEORG</th>
<th>TABSCHEMA</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1_C</td>
<td>C</td>
<td>DB2105</td>
</tr>
<tr>
<td>SYN130508191807459235_T1</td>
<td>C</td>
<td>SYSIBM</td>
</tr>
</tbody>
</table>

The Information Management Specialists
Impression #2

Extreme Compression

The Information Management Specialists
Storage Optimization in DB2

- Provides storage compression services to optimize the performance and footprint of your data
- Basic compression features included in several editions
- Advanced compression features bundled as DB2 Storage Optimization Feature, included in Advanced Enterprise Server edition
  - Available for DB2 Enterprise Server edition with a PVU license
BLU Tables – Compression and Loading Data

• Multiple compression techniques are combined in DB2 10.5 with BLU Acceleration to create a near optimal compression strategy
  ▪ Column-level dictionaries – 1 per column
  ▪ Page-level dictionaries may also be created
• Compression is always on – no options to enable/disable
• To get good compression – LOAD large representative data
• Increase utility heap
BLU Tables – Compression and Loading Data

• Estimate savings for column tables
  ▪ PCTPAGESSAVED column in SYSCAT.TABLES

• Load – primary key indexes may exist

• Sorting data before load?
  ▪ Yes, that’s an advantage
  ▪ Data skipping works better with sorted data
  ▪ Also compression techniques work better with sorted data
BLU Tables – Calculating Size

- ADMIN_GET_TAB_INFO table function or ADMINTABINFO view can be used to calculate size of column-organised table:
  - COL_OBJECT_P_SIZE + (user data)
  - DATA_OBJECT_P_SIZE + (meta data)
  - INDEX_OBJECT_P_SIZE (unique indexes)
  - + size of Synopsis table if you really want to be accurate!
BLU Tables – Space Reclamation

• Automatic!
  ▪ Background process frees extents with no active values
  ▪ Online
  ▪ The storage can be subsequently reused by any table in the table space

• No need for costly DBA space management and REORG utility

• Enabled for column-organized tables when DB2_WORKLOAD=ANALYTICS

• Synopsis table is cleaned up when you run REORG RECLAIM EXTENTS on its base table
  ▪ Needs more than one extent worth of data per column
DB2 Geek T-shirt moment!

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

- If HWM is > Used Pages for BLU tables after automatic space reclamiation, space can be released to server by:
  
  a) `ALTER TABLE ... REDUCE MAX`
  
  b) `ALTER TABLE ... LOWER HIGH WATER MARK` followed by `ALTER TABLE ... REDUCE MAX`
BLU Tables Compression – Initial Results

• Linux on Intel server
• “Modest” table: ~200GB
• To get a sense on how well the compression strategy in DB2 10.5 works, I wanted to compare it with Adaptive compression that was introduced in DB2 10.1
## BLU Tables Compression – Initial Results

<table>
<thead>
<tr>
<th>Table</th>
<th>Size (GB)</th>
<th>Storage savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1 – Row organized (around 10 columns, mostly VARCHAR) uncompressed</td>
<td>185</td>
<td>N/A</td>
</tr>
<tr>
<td>T2 – T1 Adaptive compressed</td>
<td>57.4</td>
<td>3.23x, 69% compared to T1</td>
</tr>
<tr>
<td>T3 - Convert T1 to column organized using db2convert</td>
<td>12</td>
<td>15.4x, 93.5% compared to T1, 4.78x, 79.1% compared to T2</td>
</tr>
<tr>
<td>T4 - Create new column organized table like T1, and LOAD data</td>
<td>12</td>
<td>Same as T3</td>
</tr>
</tbody>
</table>
Impression #3
Fast Reporting

The Information Management Specialists
Fast Reporting

• Dynamic in-memory columnar data store
  ▪ Work is performed directly on columns
  ▪ Rows are not materialized until absolutely necessary to build the result set
    ▶ Data remains compressed during evaluation
  ▪ Columnar data is kept compressed in memory which means more data can fit in memory
  ▪ Data can be larger than the available RAM
    ▶ Data moved intelligently from storage to memory as needed, thereby delivering in-memory performance without the limitations of an in-memory only system.
Fast Reporting

- Smart data skipping that eliminates unnecessary processing of irrelevant data
  - Results in order of magnitude savings in I/O, RAM, and CPU.
- Hardware optimisation using Parallel Vector Processing (which is a combination of Multi-core and SIMD (Single Instruction Multiple Data) parallelism)
  - Apply a single instruction to many data elements simultaneously by leveraging SIMD parallelism found on the latest chips
  - Careful attention to physical cores of the server so that queries on BLU tables are automatically core-parallelised.
## Fast Reporting - Initial Results

<table>
<thead>
<tr>
<th>Table</th>
<th>Sample Workload Total Response time (seconds)</th>
<th>Sample Workload Response time speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1 – Row organized (around 10 columns, mostly VARCHAR)</td>
<td>1,385</td>
<td>N/A</td>
</tr>
<tr>
<td>T3 - Convert T1 to column organized using db2convert</td>
<td>31</td>
<td>45x faster</td>
</tr>
<tr>
<td>T4 - Create new column organized table like T1, and LOAD data</td>
<td>31</td>
<td>Same as T3</td>
</tr>
</tbody>
</table>
Impression #4
Seamless Integration

The Information Management Specialists
Seamless Integration

• Is BLU Acceleration a new:
  ▪ Engine
  ▪ Analytic Layer on top of DB2?
  ▪ Bolt on feature?
  ▪ Hardware rip and replace strategy?

The Information Management Specialists
Seamless Integration

• DB2 10.5 continues trend set by previous versions (remember pureXML?)

• BLU Acceleration is a simple new format to store table data (column-organized table store in DB2)

• Column-based tables and row-based tables co-exist in harmony
  ▪ Optimiser aware of both table types
  ▪ No change from an application perspective
    ▶ Same SQL and language interfaces
  ▪ No change from an operational perspective
    ▶ Same process model, storage, memory and utilities
Seamless Integration

• Integrated Tooling Support
  • DBA support through Data Studio
  • Workload monitoring support with Optim Performance Manager (OPM)
  • Workload query tuning support with Optim Query Workload Tuner
  • Design support with IBM InfoSphere Data Architect (IDA) and InfoSphere Warehouse (ISW) Design Studio
  • Data movement support with ISW SQW (SQL Warehousing Tool)
  • Cognos Automated Summary Table creation using BLU Acceleration with Cognos ROLAP (Relational online analytical processing)
Seamless Integration

• BLU Acceleration is:
  ▪ Seamlessly embedded in DB2 which is known for it’s scalability, reliability, efficiency.
    ▶ Part and parcel of DB2 engine
    ▶ Same DB2 you have known **without** the tuning complexity behind analytic workloads
    ▶ Mitigates risk
DB2 LUW 10.5 with BLU Acceleration: First Impressions

- Overview of BLU Acceleration and Columnar tables
- DB2 10.5 Packaging Simplification
- My 4 Impressions

Columnar Tables – Guidance & Monitoring

- Summary
Columnar Tables – Guidance & Monitoring

• Are column-organised tables a magic bullet?

• Suited for
  ▪ Analytical workloads
  ▪ Data marts, table scans over star schemas
  ▪ Range scans, grouping, aggregation, joins
  ▪ Queries that touch only a subset of the columns in a table

• Not really suited for
  ▪ OLTP
  ▪ Point queries with highly selective index access
  ▪ Small, frequent write operations
  ▪ Queries touch many or all columns in a table
  ▪ Heavy use of XML, Temporal, LOBs
Columnar Tables – Guidance & Monitoring

• Table organisation
  ▪ MON_GET_TABLE()
    ▶ TAB_ORGANIZATION
      – Column or Row organised?
    ▶ NUM_COLUMNNS REFERENCED
      – Columns referenced in queries
    ▶ SECTION_EXEC_WITH_COL_REFERENCES
      – Queries referencing columns using scan

• Query/Bufferpool/Package cache performance
  ▪ Look for new %COL% monitor elements in
    ▶ MON_GET_DATABASE()
    ▶ MON_GET_SERVICE_SUBCLASS()
    ▶ SYSIBMADM.MON_BP_UTILIZATION
    ▶ ...

The Information Management Specialists
Columnar Tables – Guidance & Monitoring

• Remember the excellent Design Advisor?
• Optim Query Workload Tuner (OQWT) extended for BLU tables
  ▪ Workload Table Organization Advisor
    ▸ Recommends migrating row-organized tables to column-organized tables by examining all tables in a query workload.
    ▸ Shows cardinalities of the tables
    ▸ How many statements in the query workload reference them
    ▸ Cumulative cost of running the statements that reference a table, and more
    ▸ DDL for implementing the changes that are necessary to migrate the tables can be viewed and saved
  ▪ And more ....
Columnar Tables – Guidance & Monitoring

• Memory considerations
  ▪ How much of your table data is “hot”
  ▪ How many table columns are “active”
  ▪ Rule of thumb:
    ▶ Compressed table size * Hot data/Total data * Active columns/Total columns
  ▪ Remember – do not need all data in memory

• Hardware considerations – if you can, go with latest processors
  ▪ Larger CPU caches
  ▪ Facilitate SIMD instructions
DB2 LUW 10.5 with BLU Acceleration: First Impressions

- Overview of BLU Acceleration and Columnar tables
- DB2 10.5 Packaging Simplification
- My 4 Impressions
- Columnar Tables – Guidance & Monitoring

Summary
Summary

• Extreme Compression
  ▪ Massive storage (read I/O) savings
    ▶ Best I/O is no I/O!
  ▪ Addresses Big Data issue of size

• Fast reporting
  ▪ Query response times much, much faster
    ▶ Minimal I/O being performed only on the columns and values that match the query
  ▪ Addresses Big Data issue of response times
Summary

• Simplicity and Seamless Integration
  ▪ DBA tuning time savings!
    ▶ Can we put a number to this?
  ▪ Embedded in DB2 kernel
  ▪ Very easy to create Operational Warehouse
  ▪ Equally easy to create Data Marts
  ▪ Addresses Big Data issue of development time
Feedback / Questions

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