DB2 with BLU Acceleration:
Der Quantensprung in der Informationstechnologie

Berni Schiefer
Distinguished Engineer
IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM’s sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.
DB2 with BLU Acceleration?

- Extreme performance out-of-the-box
- Massive storage savings
  - No indexes required
- Lower cost of operational analytics

- Built seamlessly into DB2
- Consistent SQL, interfaces, administration
- Dramatic simplification
  - Less to design
  - Less to tune
  - Just Load and Go

- In Memory Optimized
  - Compressed in memory
- Modern CPU Exploitation
- I/O Optimized
  - Only read columns of interest
What is DB2 with BLU Acceleration?

- **New technology for analytic queries in DB2 LUW**
  - DB2 column-organized tables add columnar capabilities to DB2 databases
    - Table data is stored column organized rather than row organized
    - Using a vector processing engine
    - Using this table format with star schema data marts provides **significant improvements to storage, query performance, ease of use and time-to-value**
  - New unique runtime technology which leverages the CPU architecture and is **built directly into the DB2 kernel**
  - New unique encoding for **speed and compression**
    - This new capability is both main-memory optimized, CPU optimized and I/O optimized
# How fast is it?

## Results from the DB2 10.5 Beta

<table>
<thead>
<tr>
<th>Customer</th>
<th>Speedup over DB2 10.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large Financial Services Company</td>
<td>46.8x</td>
</tr>
<tr>
<td>Global ISV Mart Workload</td>
<td>37.4x</td>
</tr>
<tr>
<td>Analytics Reporting Vendor</td>
<td>13.0x</td>
</tr>
<tr>
<td>Global Retailer</td>
<td>6.1x</td>
</tr>
<tr>
<td>Large European Bank</td>
<td>5.6x</td>
</tr>
</tbody>
</table>

8x-25x improvement is common

“It was amazing to see the faster query times compared to the performance results with our row-organized tables. The performance of four of our queries improved by over 100-fold! The best outcome was a query that finished 137x faster by using BLU Acceleration.”

- Kent Collins, Database Solutions Architect, BNSF Railway
**BLU Acceleration Performance**

- **POPS (Proof of Performance and Scalability)**
  - Derived from Redbrick performance test
  - Classic sales analytics
  - 5.5 years of data (2000 days) for 63 stores
    - ~4TB of raw data
    - 2 fact tables
    - 5 dimension tables
  - Broad range of queries with varying selectivity / aggregation

- **Substantial Storage Savings with BLU Acceleration**
  - 2.5x less space than DB2 10.1

- **Massive Performance Gains**
  - 25x speedup over DB2 10.1
  - Maximum query speed up over 400x

![Diagram showing performance comparison between DB2 10.1 and BLU Acceleration with Intel® Xeon® Processor E5-4650]
Storage Savings

- **Multiple examples of data requiring substantially less storage**
  - 5% of the uncompressed size
  - Fewer objects required
- **Multiple compression techniques**
  - combined to create a nearly optimal compression strategy
- **Compression algorithm adapts to the data**
Seamless Integration into DB2

- **Built seamlessly into DB2 – integration and coexistence**
  - Column-organized tables can coexist with existing, traditional, tables
    - Same schema, same storage, same memory
  - Integrated tooling support
    - Optim Query Workload Tuner recommends BLU Acceleration deployments

- **Same SQL, language interfaces, administration**
  - Column-organized tables or the combinations of column-organized and row-organized tables can be accessed within the same SQL statements

- **Dramatic simplification – Just “Load and Go”**
  - Faster deployment
    - Fewer database objects required to achieve same outcome
  - Requires less ongoing management due to it’s optimized query processing and less database object required
  - Simple migration
    - Conversion from traditional row table to BLU Acceleration is easy
    - DB2 Workload Manager identifies workloads to tune
    - Optim Query Workload Tuner recommends BLU Acceleration table transformations
    - Users only notice speed up/DBA’s only notice less work!
  - Management of single server solutions less expensive than clustered solutions
Simple to Deploy and Operate

- **Operations**
  - Simply load and go
  - Installation to business value in ~2 days
  - Ease of evaluation and performs as advertised

- **BI developers and DBAs – faster delivery**
  - No configuration or physical modeling
  - No indexes or tuning – out of the box performance
  - Data Architects / DBA focus on business value, not physical design

- **ETL developers**
  - No aggregate tables needed – simpler ETL logic
  - Faster load and transformation times

- **Business analysts**
  - Train of thought analysis – 5x to 100x faster
  - True ad hoc queries – no tuning, no indexes
  - Ask complex queries against large datasets
Analytics Data Mart
*From Transactional Database*

ERP or other transactional system

Transaction Database

Easily create and load a BLU in-memory mart

Create tables, load and go!
- Instant performance boost
- Handles terabytes of data
- No indexes/aggregates to create & tune
- Multi-platform software flexibility

Line of Business Analytics Data Mart

Multi-platform software

Analytic Data Mart (BLU Tables)
Enterprise Data Warehouse Offload

Data Mart Acceleration

EDW Application

Poor Performing Oracle or Teradata Warehouse

Create tables, load and go!
- Instant performance boost
- Handles terabytes of data
- No indexes/aggregates to create & tune
- Multi-platform software flexibility

Cognos BI

with BLU Acceleration

Easily create and load a BLU in-memory mart

OLAP Application

Cognos BI

with BLU Acceleration

Multi-platform software

Analytic Data Mart (BLU Tables)

EDW Application

OLAP Application

Cognos BI

with BLU Acceleration

Multi-platform software

Analytic Data Mart (BLU Tables)
Cognos with BLU Acceleration

- **Cognos BI 10.2**
  - Dynamic Cubes (ROLAP)
  - Extends Dynamic Query with in-memory caching of members, data, expressions, results, and aggregates
  - 963GB of raw data
    - 7 fact tables
    - 17 dimension tables
  - Workload consists of both loading the cache and running ad-hoc reports not satisfied in the cache

- **In-Memory Aggregate Cache Load**
  - 18x faster than DB2 10.1

- **Ad-hoc Cognos Reports**
  - 14x faster than DB2 10.1
The Seven Big Ideas of DB2 with BLU Acceleration
7 Big Ideas: Simple to Implement and Use

- LOAD and then... run queries
  - No indexes
  - No REORG (it’s automated)
  - No RUNSTATS (it’s automated)
  - No MDC or MQTs or Materialized Views
  - No statistical views
  - No optimizer hints

- It is just DB2!
  - Same SQL, language interfaces, administration
  - Reuse DB2 process model, storage, utilities

“...The BLU Acceleration technology has some obvious benefits: It makes our analytical queries run 4-15x faster and decreases the size of our tables by a factor of 10x. But it’s when I think about all the things I don't have to do with BLU, it made me appreciate the technology even more: no tuning, no partitioning, no indexes, no aggregates.”

-Andrew Juarez, Lead SAP Basis and DBA
7 Big Ideas: Simple to Implement and Use

- **One setting optimized the system for BLU Acceleration**
  - Set `DB2_WORKLOAD=ANALYTICS`
  - Informs DB2 that the database will be used for analytic workloads

- **Automatically configures DB2 for optimal analytics performance**
  - 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

- **Simple Table Creation**
  - If `DB2_WORKLOAD=ANALYTICS`, tables will be created column organized automatically
  - For mixed table types can define tables as `ORGANIZE BY COLUMN` or `ROW`
  - Compression is always on – No options

- **Easily convert tables from row-organized to column-organized**
  - `db2convert` utility
7 Big Ideas: 2. Compute Friendly Encoding and Compression

- **Massive compression with approximate Huffman encoding**
  - More frequent the value, the fewer bits it takes

- **Register-friendly encoding dramatically improves efficiency**
  - Encoded values packed into bits matching the register width of the CPU
  - Fewer I/Os, better memory utilization, fewer CPU cycles to process

<table>
<thead>
<tr>
<th>LAST_NAME</th>
<th>Encoding</th>
<th>Packed into register length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brown</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Brown</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gilligan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wong</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
7 Big Ideas: 2 Data Remains Compressed During Evaluation

- Encoded values do not need to be decompressed during evaluation
  - Predicates (=, <, >, >=, <=, Between, etc), joins, aggregations and more work directly on encoded values

```sql
SELECT COUNT(*) FROM T1 WHERE LAST_NAME = 'Johnson'
```

<table>
<thead>
<tr>
<th>LAST_NAME</th>
<th>Encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brown</td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
</tr>
<tr>
<td>Brown</td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
</tr>
<tr>
<td>Gilligan</td>
<td></td>
</tr>
<tr>
<td>Wong</td>
<td></td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
</tr>
</tbody>
</table>
7 Big Ideas: Multiply the Power of the CPU

- Performance increase with Single Instruction Multiple Data (SIMD)
- Using hardware instructions, DB2 with BLU Acceleration can apply a single instruction to many data elements simultaneously
  - Predicate evaluation, joins, grouping, arithmetic

“Intel is excited to see a 25x improvement in query processing performance using DB2 10.5 with BLU acceleration over DB2 10.1. To achieve these amazing gains, IBM has taken advantage of the Advanced Vector Extensions (AVX) instruction set on Intel® Xeon® processor E5-based systems.”

- Pauline Nist, GM, Enterprise Software Alliances, Datacenter & Connected Systems Group
7 Big Ideas: 4 Core-Friendly Parallelism

- Careful attention to physical attributes of the server
  - Queries on BLU Acceleration tables automatically parallelized

- **Maximizes** CPU cache, cacheline efficiency

“During our testing, we couldn’t help but notice that DB2 10.5 with BLU Acceleration is excellent at utilizing our hardware resources. The core-friendly parallelism that IBM talks about was clearly evident and I didn’t even have to partition the data across multiple servers.”

- Kent Collins, Database Solutions Architect, BNSF Railway
7 Big Ideas: 5 Column Store

- **Minimal I/O**
  - Only perform I/O on the columns and values that match query
  - As queries progresses through a pipeline the working set of pages is reduced

- **Work performed directly on columns**
  - Predicates, joins, scans, etc. all work on individual columns
  - Rows are not materialized until absolutely necessary to build result set

- **Improved memory density**
  - Columnar data kept compressed in memory

- **Extreme compression**
  - Packing more data values into very small amount of memory or disk

- **Cache efficiency**
  - Data packed into cache friendly structures
7 Big Ideas: 6 Scan-Friendly Memory Caching

- New algorithms cache in RAM effectively

- **High percent of interesting data fits in memory**
  - We leave the interesting data in memory with the new algorithms

- **Data can be larger than RAM**
  - No need to ensure all data fits in memory
  - Optimization for in memory and I/O efficiency
7 Big Ideas: 7 Data skipping

- Automatic detection of large sections of data that do not qualify for a query and can be ignored

- Order of magnitude savings in all of I/O, RAM, and CPU

- No DBA action to define or use – truly invisible
  - Persistent storage of min and max values for sections of data values
## Optimize the Entire Hardware Stack

### In-Memory Optimized
- **Memory latency optimized for**
  - Scans
  - Joins
  - Aggregation
- **More useful data in memory**
  - Data stays compressed
  - Scan friendly caching
- **Less to put in memory**
  - Columnar access
  - Late materialization
  - Data skipping

### CPU Optimized
- **CPU acceleration**
  - SIMD processing for
    - Scans
    - Joins
    - Grouping
    - Arithmetic
- **Keeping the CPUs busy**
  - Core friendly parallelism
- **Less CPU processing**
  - Operate on compressed data
  - Late materialization
  - Data skipping

### I/O Optimized
- **Less to read**
  - Columnar I/O
  - Data skipping
  - Late materialization
- **Read less often**
  - Scan friendly caching
- **Efficient I/O**
  - Specialized columnar prefetching algorithm
The system – 32 cores, 10TB table with 100 columns, 10 years of data
The query: `SELECT COUNT(*) from MYTABLE where YEAR = '2010'`
The optimistic result: sub second 10TB query! Each CPU core examines the equivalent of just 8MB of data
Unlimited concurrency with “Automatic WLM”

- DB2 10.5 has built-in and automated query resource consumption control
- Every additional query that runs naturally consumes more memory, locks, CPU and memory bandwidth. In other database products more queries means more contention.
- DB2 10.5 automatically allows a high level of concurrent queries to be submitted, but limits the number that consume resources at any point in time
- Enabled automatically when DB2_WORKLOAD=ANALYTICS
Automatic Space Reclaim

- **Automatic Space Reclamation**
  - Frees extents with no active values
  - The storage can be subsequently reused by any table in the tablespace
- **No need for costly DBA space management, and reorg utility**
- **Enabled out-of-the-box for column-organized tables when DB2_WORKLOAD=ANALYTICS**
- **Space is freed online while work continues**
- **Regular space management can result in increased performance of RUNSTATS and some queries**

![Diagram showing automatic space reclaim process]

DELETE * FROM MyTable WHERE Year = 2012

These extents hold only deleted data
Informational Uniqueness

- **DB2 10.5 introduces informational uniqueness constraints**
  - Enforced uniqueness remains the default
  - Informational (i.e., **NOT ENFORCED**) constraints do not enforce uniqueness
  - Valuable when data is coming from a trusted source

- **Benefits**
  - Less storage required! No index is created to enforce the constraint.
  - No runtime overhead to maintain unique indexes during LOAD, INSERT, UPDATE, or DELETE
  - The uniqueness definition informs the query compiler of unique data, enabling opportunities for superior query execution plans

```sql
CREATE TABLE t1 (c1 INTEGER NOT NULL, c2 INTEGER,
                 PRIMARY KEY (c1) NOT ENFORCED);

ALTER TABLE t1 ADD CONSTRAINT uniqu1 UNIQUE (c2) NOT ENFORCED;
```
What Does BLU Acceleration Give Us?

- Order of magnitude improvements
  - Consumability
  - Speed
  - Storage savings

- Breakthrough technology
  - Combines and extends the very best technologies
  - Over 25 patents filed and pending
  - Leveraging years of IBM R&D spanning 10 laboratories in 7 countries worldwide

- Typical experience
  - Simple to implement and use
  - 8x-25x performance gains
  - 10x storage savings vs. uncompressed data with indexes

Request a beta nomination form
  db2beta@ca.ibm.com