

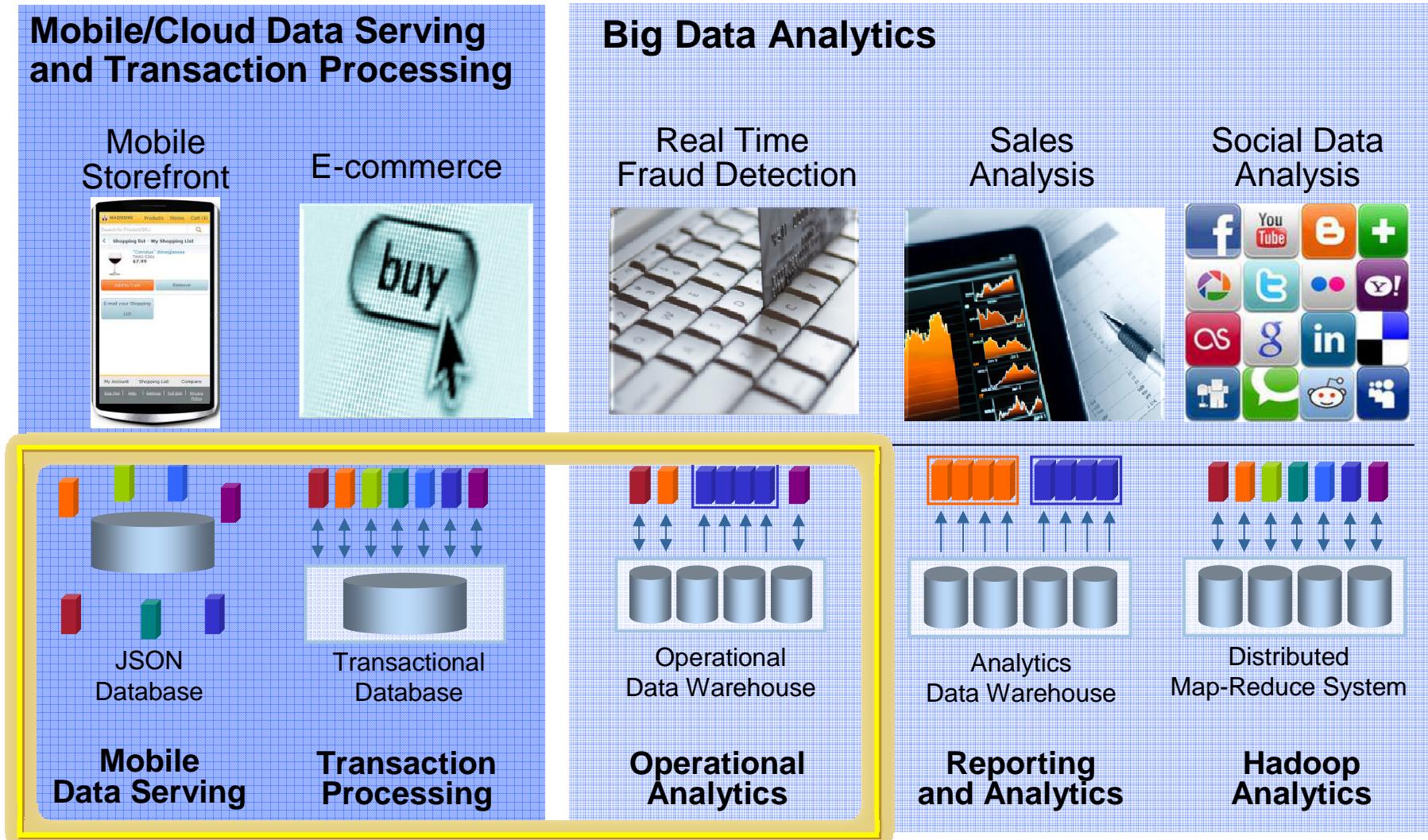
# DB2® 10.5

*with BLU Acceleration  
... and more!*

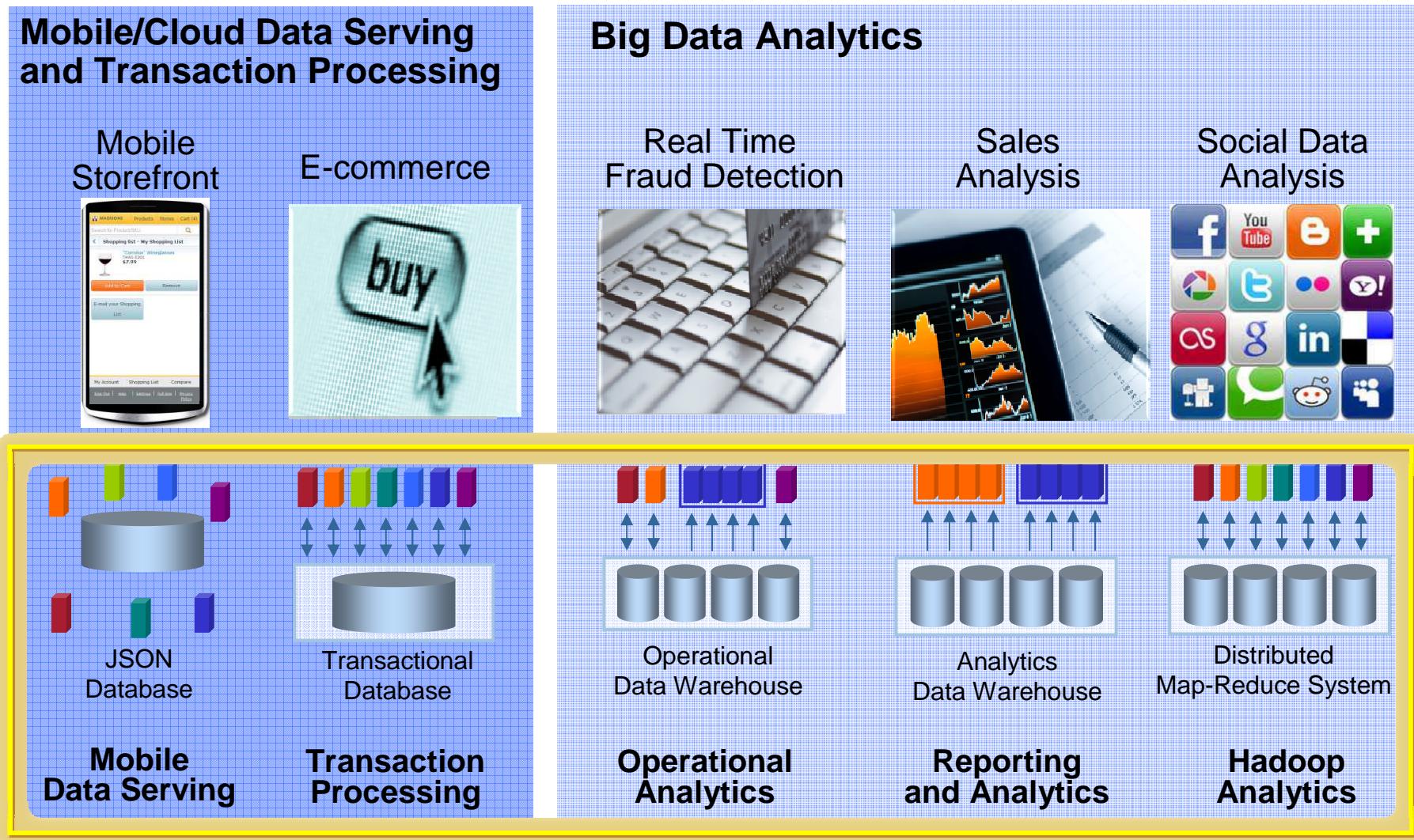


**Bob Harbus**  
**Global IM Technical Sales and Competitive Database**  
**IBM Toronto Laboratory**  
**[rharbus@ca.ibm.com](mailto:rharbus@ca.ibm.com)**

# Different Workloads Require Different Data Systems Where DB2 Plays in an Expert System



# Different Workloads Require Different Data Systems Where DB2 Plays in a Software Solution



## DB2 10.5 with BLU Acceleration



## CIOs must reskill to succeed with in-memory computing, says Gartner

No Reskill  
Required



However Pezzini warned: "Organisations that do not consider adopting in-memory application infrastructure technologies risk being out-innovated by competitors that are early mainstream users of these capabilities."

The second core skill area lies in architectural skills. Pezzini said: "In-memory computing requires application and deployment architectures more complex than those needed for traditional applications."

The Forrester logo, which consists of the word "FORRESTER" in white capital letters inside a dark green oval shape.

FOR APPLICATION DEVELOPMENT & DELIVERY PROFESSIONALS

### **It's The Dawning Of The Age Of BI DBMS**

*Complement Your RDBMS-Based BI With More Agile DBMS Technologies*

## Business Analytics

**Why The Last Decade of BI Best-Practice Architecture is Rapidly Becoming Obsolete**

# Introducing BLU Acceleration

## IBM Research and Development Lab Innovations

**BLU Acceleration includes over 30 new patents and patents pending from our labs!**

- **Dynamic In-Memory**

In-memory columnar processing with dynamic movement of data from storage data



- **Actionable Compression**

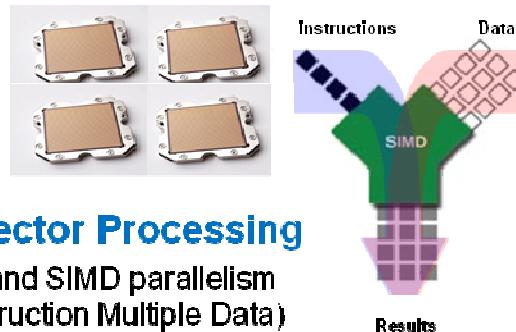
Patented compression technique that preserves order so that the data can be used without decompressing



# Encoded

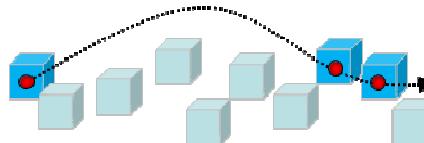
- **Parallel Vector Processing**

Multi-core and SIMD parallelism (Single Instruction Multiple Data)



- **Data Skipping**

Skips unnecessary processing of irrelevant data



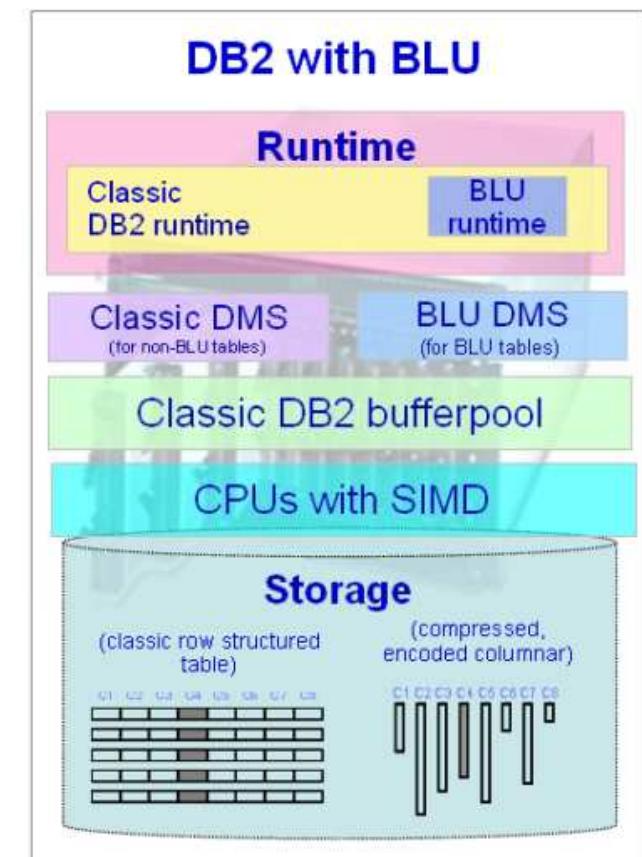
**Super Fast, Super Easy — *Create, Load and Go!***

No Indexes, No Aggregates, No Tuning, No SQL changes, No schema changes

# What is DB2 with BLU Acceleration?

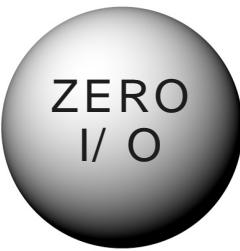
- **New technology for analytic queries in DB2 LUW**

- 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



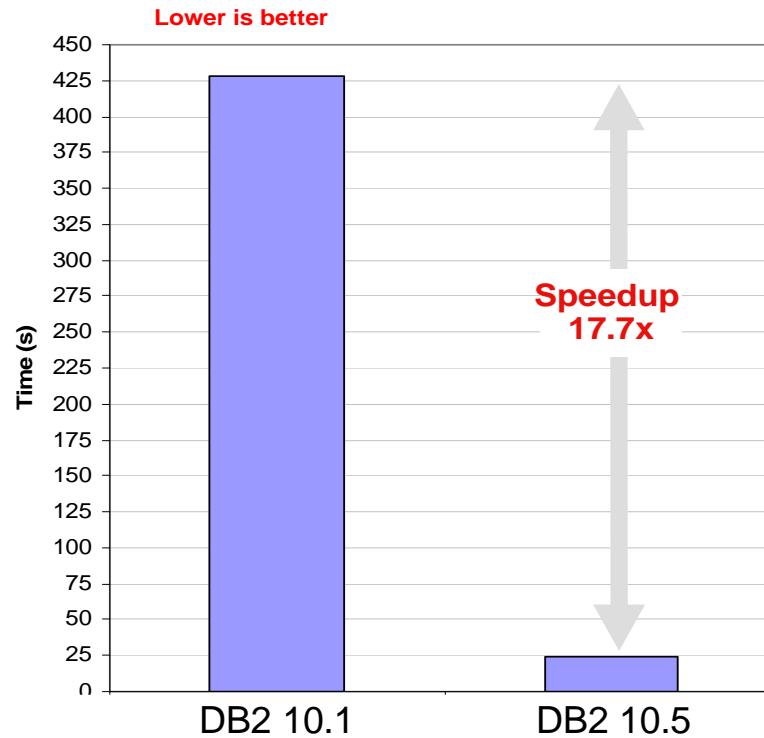
# In-memory Isn't Everything

*Great performance takes a lot more than just “in-memory”*



Performance when all data is cached  
Zero I/O in both DB2 10.1 and DB2 10.5  
with BLU Acceleration<sup>1</sup>

## Cognos 15 queries 100GB no I/O



# How Fast Is BLU Acceleration?

Customer	Performance Gains
BNSF	Up to 137x
Handelsbanken	7x – 100x
Triton Consulting	46x
Yonyou	40x
Coca-Cola Bottling	4x - 15x

10x-25x  
speedup  
is common

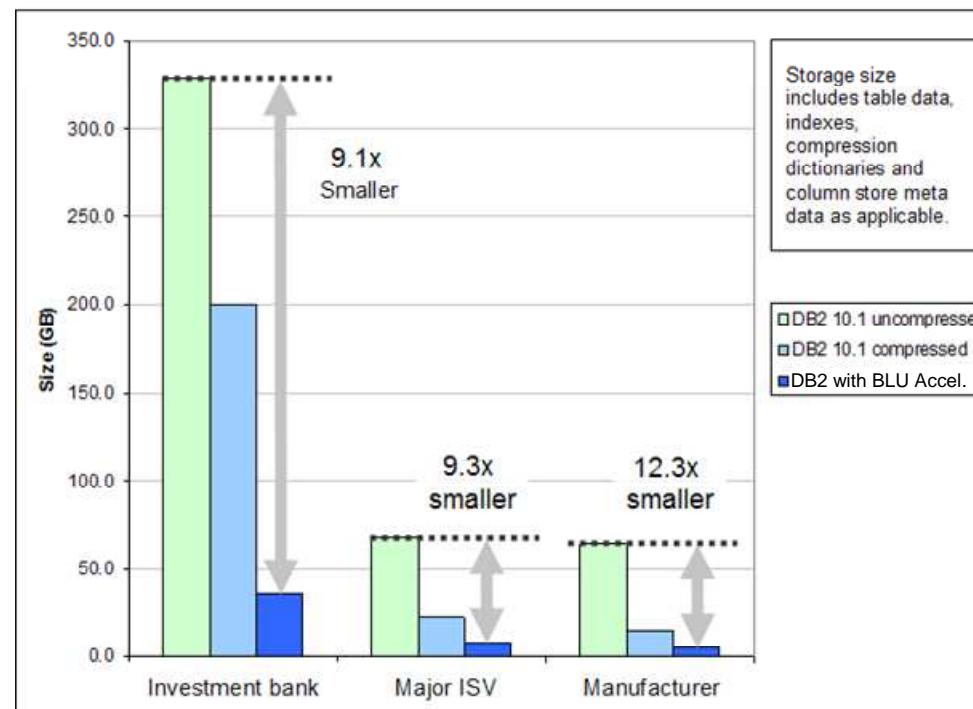


*"It was amazing to see the faster query times compared to the performance results with our row-organized tables. The performance of four of our queries improved by over 100-fold! The best outcome was a query that finished 137x faster by using BLU Acceleration."*

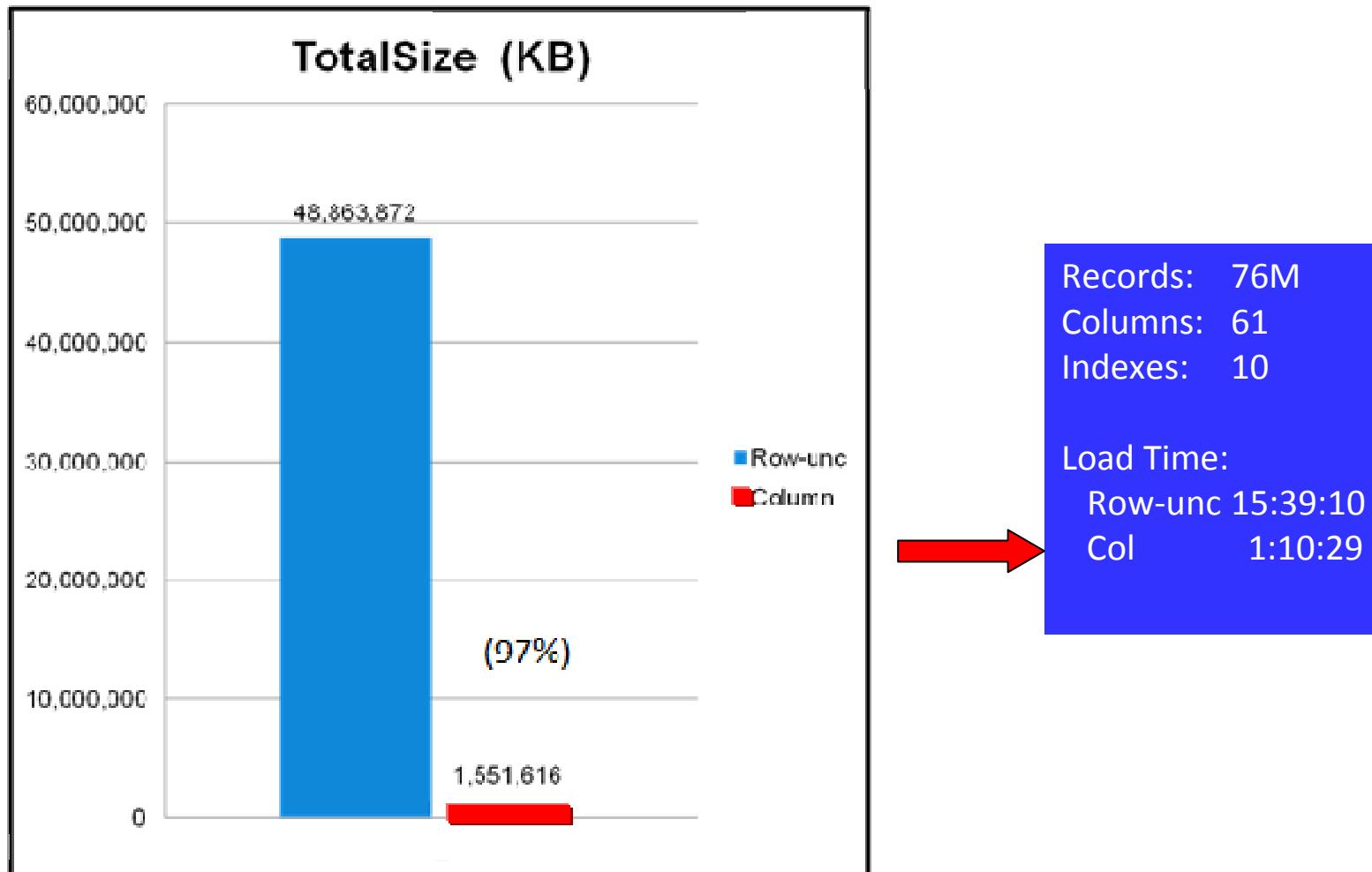
- Kent Collins, Database Solutions Architect, BNSF Railway

# 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 near optimal compression strategy
- **Compression algorithm adapts to the data**



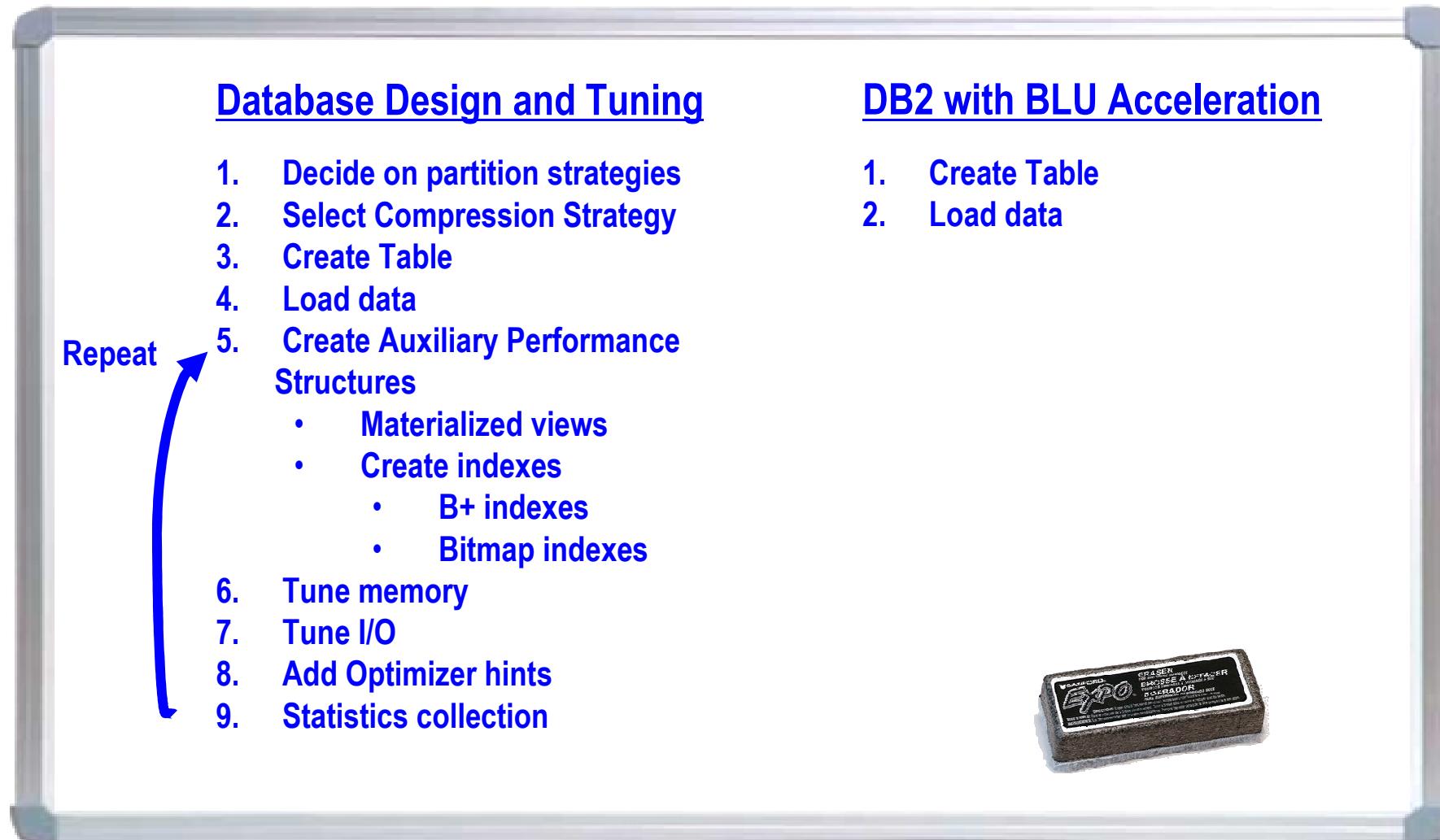
## DB2 10.5 BLU Compression – Customer Example



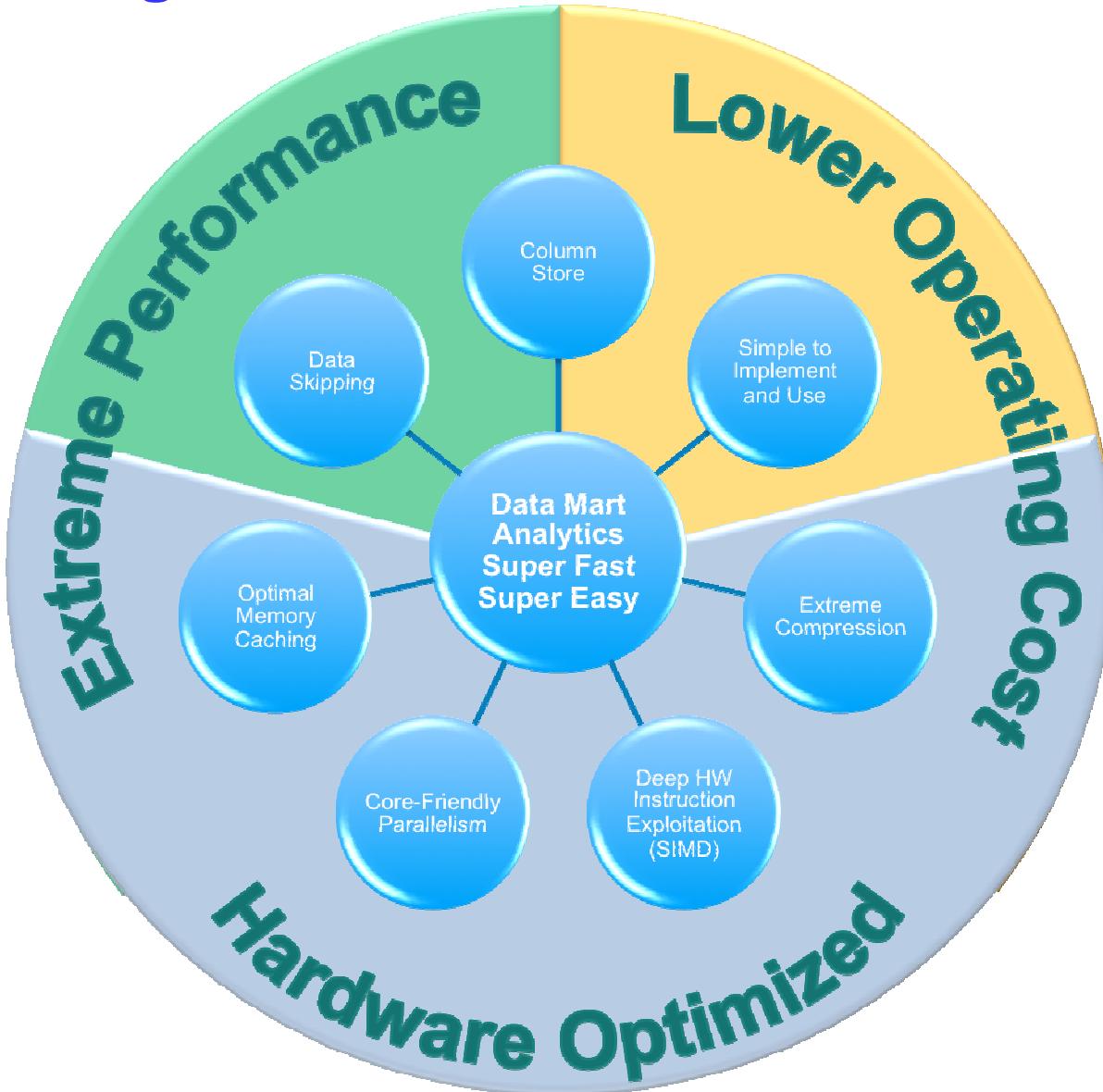
# Seamless Integration into DB2

- **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 (OQWT) recommends BLU Acceleration deployments
- **Same SQL, language interfaces, administration**
  - Column-organized tables or combinations of column-organized and row-organized tables can be accessed within the same SQL statement
- **Dramatic simplification – Just “Load and Go”**
  - Faster deployment
    - Fewer database objects required to achieve same outcome
  - Requires less ongoing management due to its optimized query processing and fewer database objects required
  - Simple migration
    - Conversion from traditional row table to BLU Acceleration is easy
    - DB2 Workload Manager (WLM) 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

# Super Fast, Super Easy – Create, Load, and Go!



# The Seven Big Ideas of DB2 with BLU Acceleration



## 7 Big Ideas: ① Simple to Implement and Use

- **LOAD and then... run queries**

- No indexes
- No REORG (it's automated)
- No RUNSTATS (it's automated)
- No MDC
- No MQTs or Materialized Views
- No partitioning
- No statistical views
- No optimizer hints

- **It is just DB2!**

- Same SQL, language interfaces, administration
- Reuse DB2 process model, storage, utilities

# 7 Big Ideas: ① 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

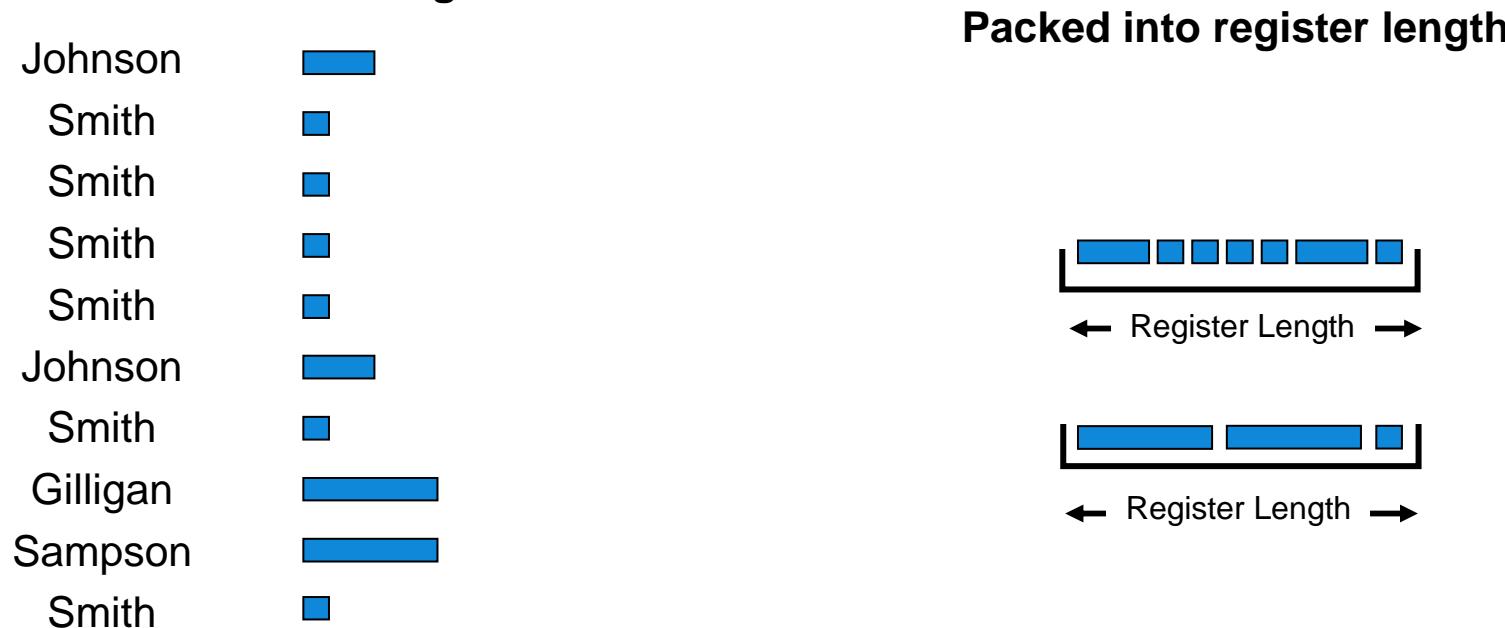
## 7 Big Ideas: ① Simple to Implement and Use

```
db2set DB2_WORKLOAD=ANALYTICS
db2 CREATE DATABASE COLDB
db2 CREATE TABLE "BLU"."FACT_RX" (
    "MONTH_ID"          DECIMAL(6,0) ,
    "DATE_OF_SERVICE"   DATE ,
    "PROVIDER_ID"       DECIMAL(10,0) ,
    "PRODUCT_ID"        DECIMAL(10,0) ,
    "PERSON_ID"         DECIMAL(10,0) ,
    "PERSON_ZIP3_CD"    VARCHAR(3) ,
    "CID"               DECIMAL(14,0) NOT NULL ,
    "FILL_NBR"          DECIMAL(10,0) ,
    "DAYS_SUPPLY"       DECIMAL(10,0) ,
    "QUANTITY_DISPENSED" DECIMAL(10,0) ,
    "PERSON_OPC"        DECIMAL(10,0) ,
    "TOTAL_AMT_PAID"    DECIMAL(14,4) ,
    "PAYER_ID"          DECIMAL(10,0)
) organize by column;
db2 LOAD FROM /tmp/fact_rx.dat OF DEL REPLACE
INTO BLU.FACT_RX
db2 "SELECT COUNT(DISTINCT RX.PROVIDER_ID)
      FROM BLU.FACT_RX RX INNER JOIN BLU.PERSON PT
      ON RX.PERSON_ID=PT.PERSON_ID"
```

## 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 optimizes CPU and memory efficiency**
  - Encoded values packed into bits matching the register width of the CPU
    - Allows for efficient simultaneous evaluation against multiple values
  - Fewer I/Os, better memory utilization, fewer CPU cycles to process

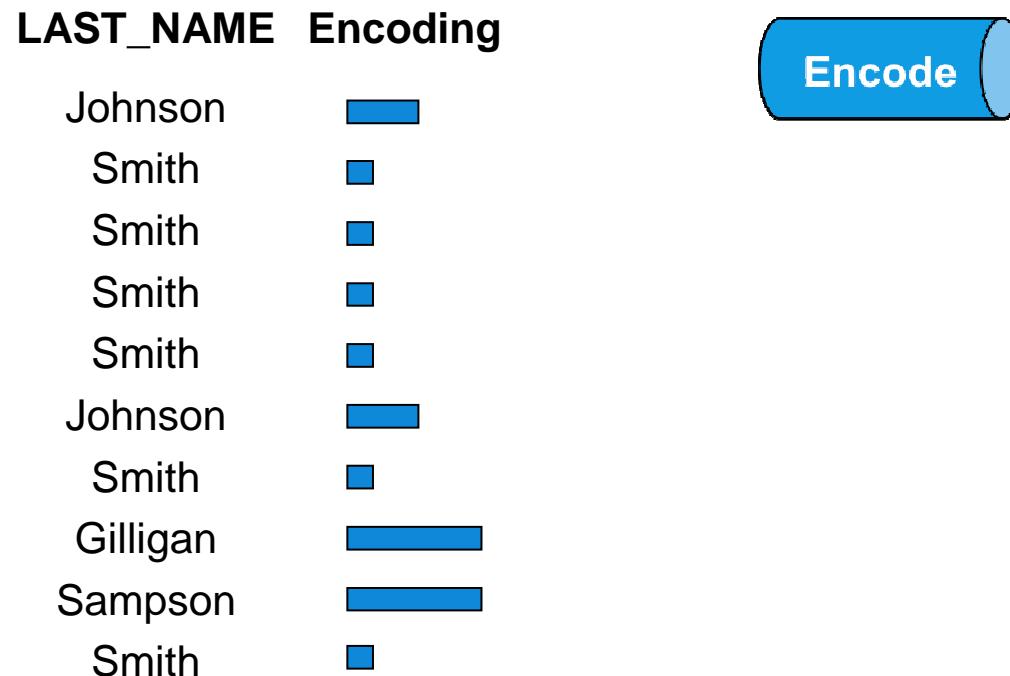
### LAST\_NAME Encoding



## 7 Big Ideas: 2 Data Remains Compressed During Evaluation

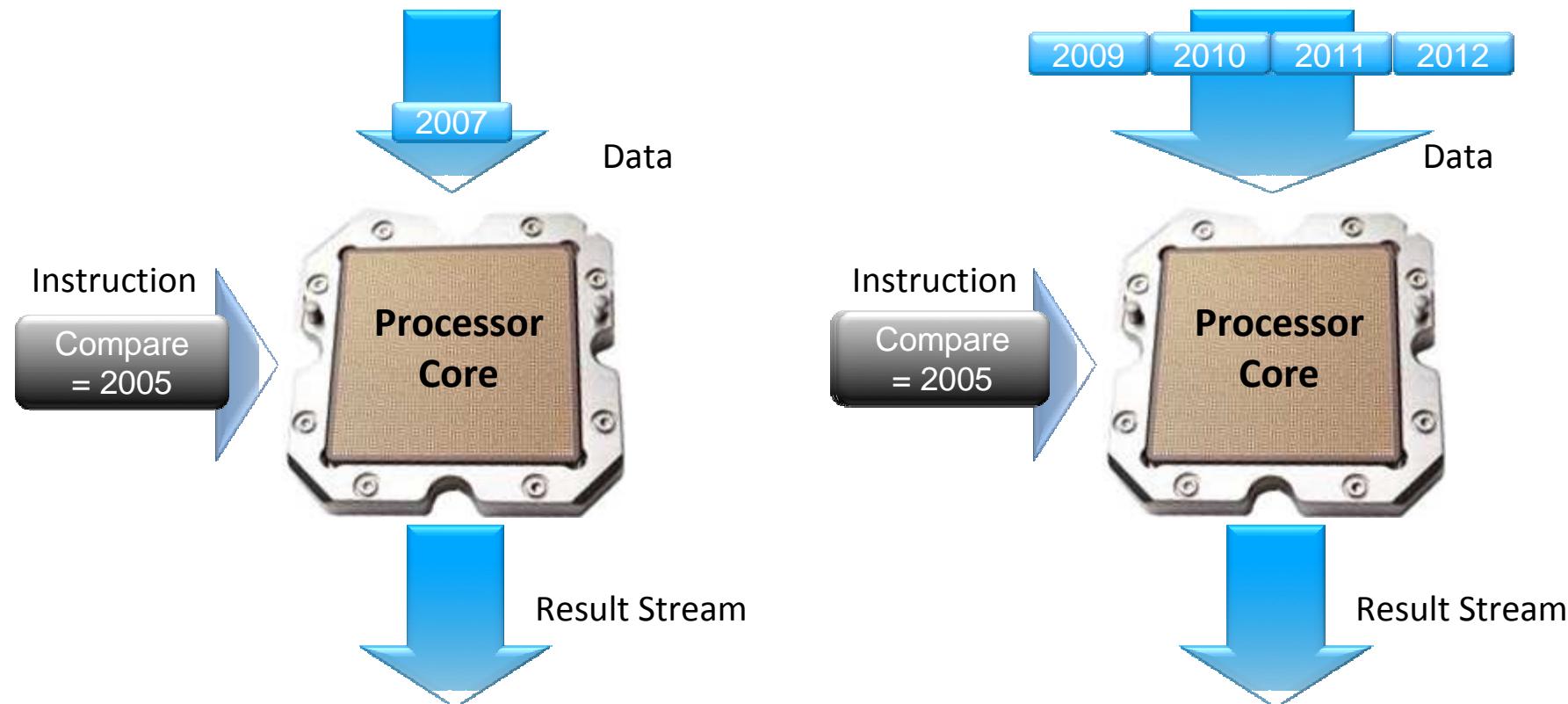
- **Encoded values do not need to be decompressed during evaluation**
  - Predicates and joins work directly on encoded values

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



## 7 Big Ideas: ③ 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

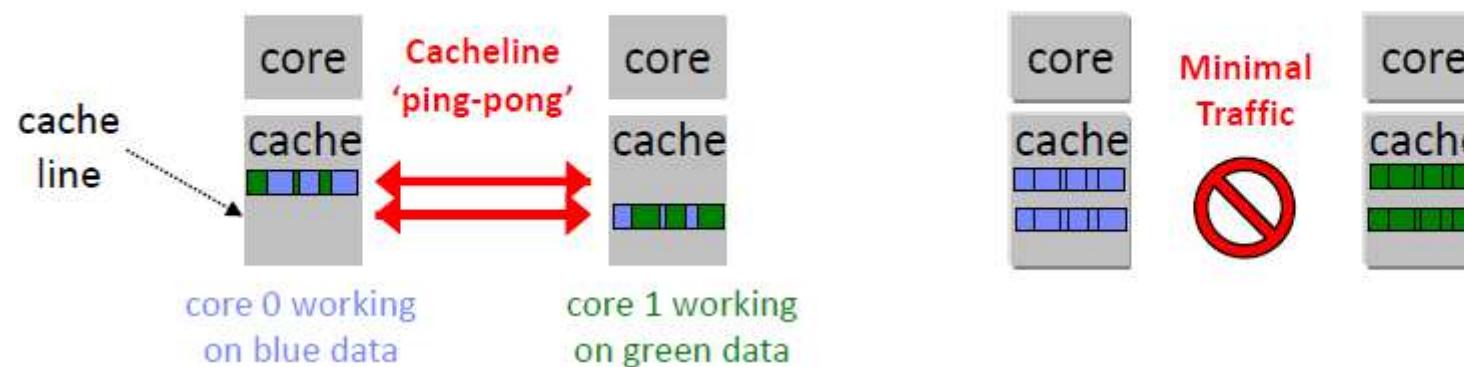


## 7 Big Ideas: 4 Core-Friendly Parallelism

- BLU queries automatically parallelized across cores, and, achieve excellent multi-core scalability via ...
  - Careful data placement and alignment
  - Careful attention to physical attributes of the server
  - ... designed to ...



- Maximize CPU cache, cacheline efficiency



# 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
    - Predicates, joins, scans, etc. all operate on columns packed in memory
    - Rows are not materialized until absolutely necessary to build result set

- **Improved memory density**

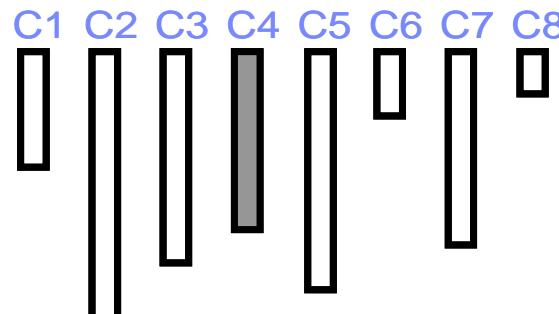
- Columnar data kept compressed in memory
  - No need to consume memory/cache space & bandwidth for unneeded columns

- **Extreme compression**

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

- **Cache efficiency**

- Data packed into cache friendly structures

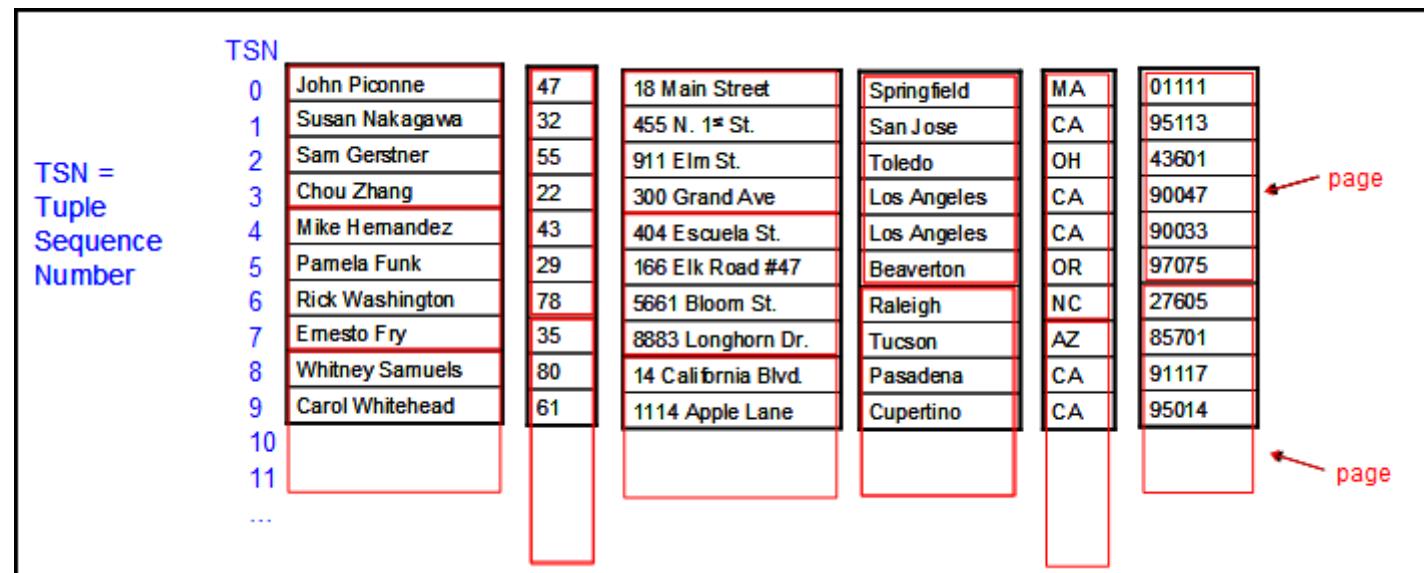
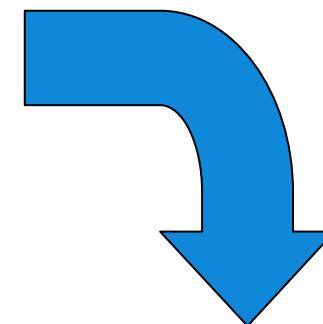


SELECT C4 ... WHERE C4=X

Consumes I/O bandwidth  
memory buffers and memory  
bandwidth only for C4

## 7 Big Ideas: ⑤ Column Store (cont.)

John Piconne	47	18 Main Street	Springfield	MA	01111
Susan Nakagawa	32	455 N. 1 <sup>st</sup> St.	San Jose	CA	95113
Sam Gerstner	55	911 Elm St.	Toledo	OH	43601
Chou Zhang	22	300 Grand Ave	Los Angeles	CA	90047
Mike Hernandez	43	404 Escuela St.	Los Angeles	CA	90033
Pamela Funk	29	166 Elk Road #47	Beaverton	OR	97075
Rick Washington	78	5661 Bloom St.	Raleigh	NC	27605
Ernesto Fry	35	8883 Longhorn Dr.	Tucson	AZ	85701
Whitney Samuels	80	14 California Blvd.	Pasadena	CA	91117
Carol Whitehead	61	1114 Apple Lane	Cupertino	CA	95014

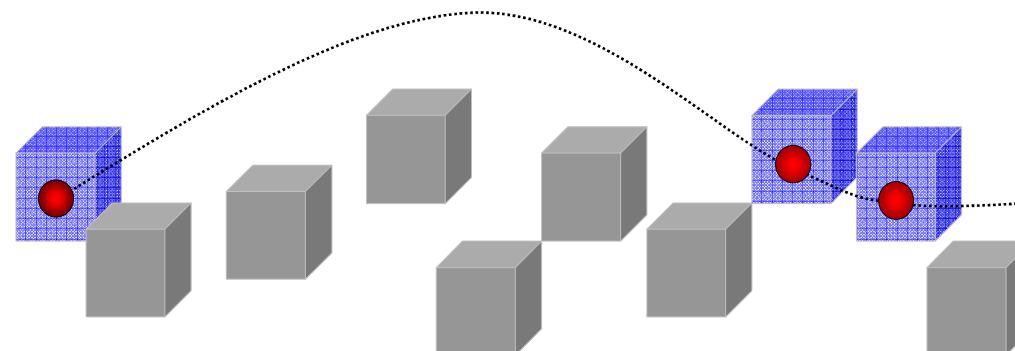


## 7 Big Ideas: ⑥ Scan-Friendly Memory Caching

- **Memory-optimized (not “In-Memory”)**
  - No need to ensure all data fits in memory
  - New algorithms cache in RAM effectively
- **BLU includes new scan-friendly victim selection to keep a near optimal % of pages buffered in memory**
  - A key BLU design point is to run well when all data fits in memory, and when it doesn’t !
  - Even with large scans, BLU prefers selected pages in the bufferpool, using an algorithm that adaptively computes a target hit ratio for the current scan, based on the size of the bufferpool, the frequency of pages being re-accessed in the same scan, and other factors
- Benefit: less I/O!

## 7 Big Ideas: 7 Data Skipping

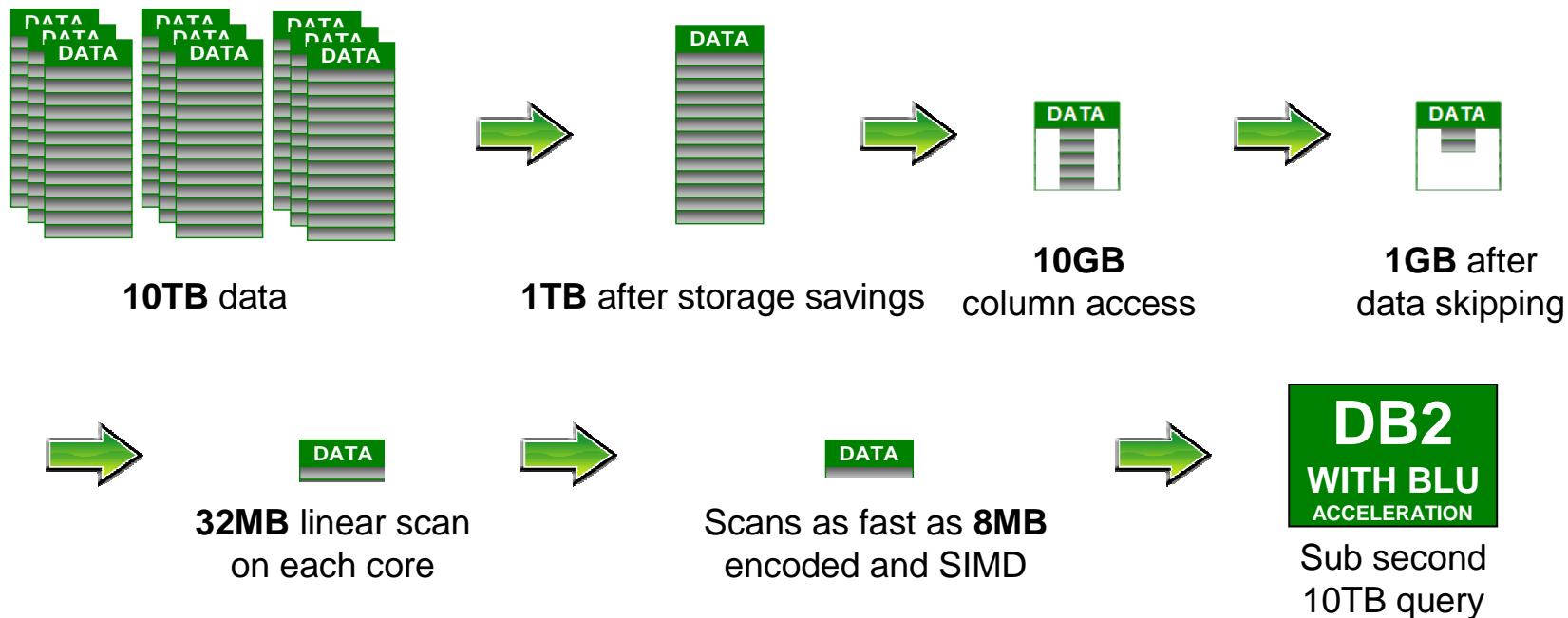
- Automatic detection of large sections of data that do not qualify for a query and can be ignored
- Order of magnitude **savings** in all of I/O, RAM, and CPU
- No DBA action to define or use – truly **invisible**
  - “Synopsis” automatically created and maintained as data is LOADED or INSERTED
  - Persistent storage of min. and max. values for sections of data values



# 7 Big Ideas: How DB2 with BLU Acceleration Helps

## ~Sub second 10TB query – An Optimistic Illustration

- 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



# POPS Benchmark: A Collaboration with Intel

- **POPS (Proof of Performance and Scalability) 10TB benchmark**

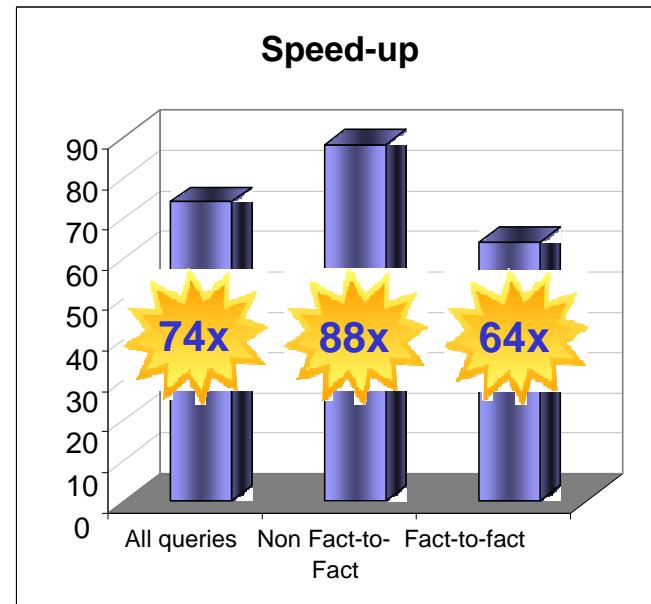
- Broad range of queries with varying selectivity and aggregation
  - ~13.5 years and 63 stores
  - 2 fact tables & 5 dimensions

- **Environment**

- Intel® Xeon® Processor
  - 1TB RAM
  - XIV storage

- **Storage savings**

- 4.5x space consumption reduction on DB2 10.5 with BLU compared to DB2 10.1



**DB2 10.5 with BLU  
vs DB2 10.1**

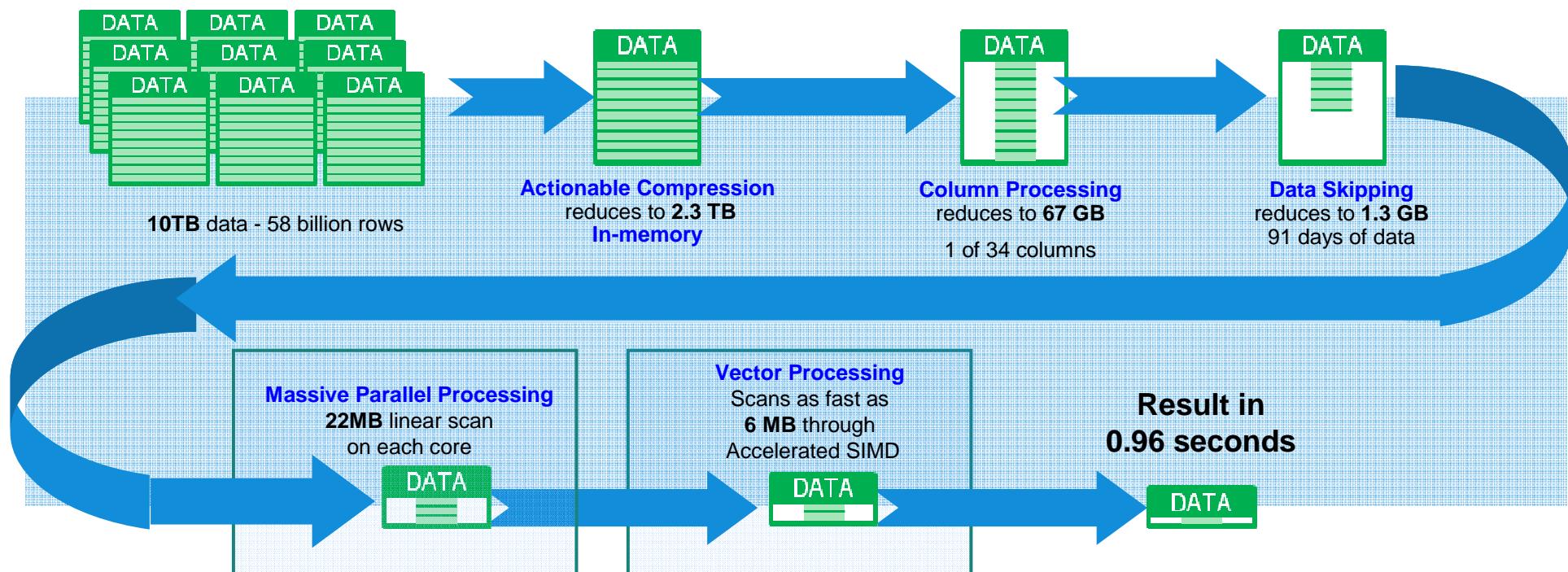


# BLU Acceleration: 10TB Query

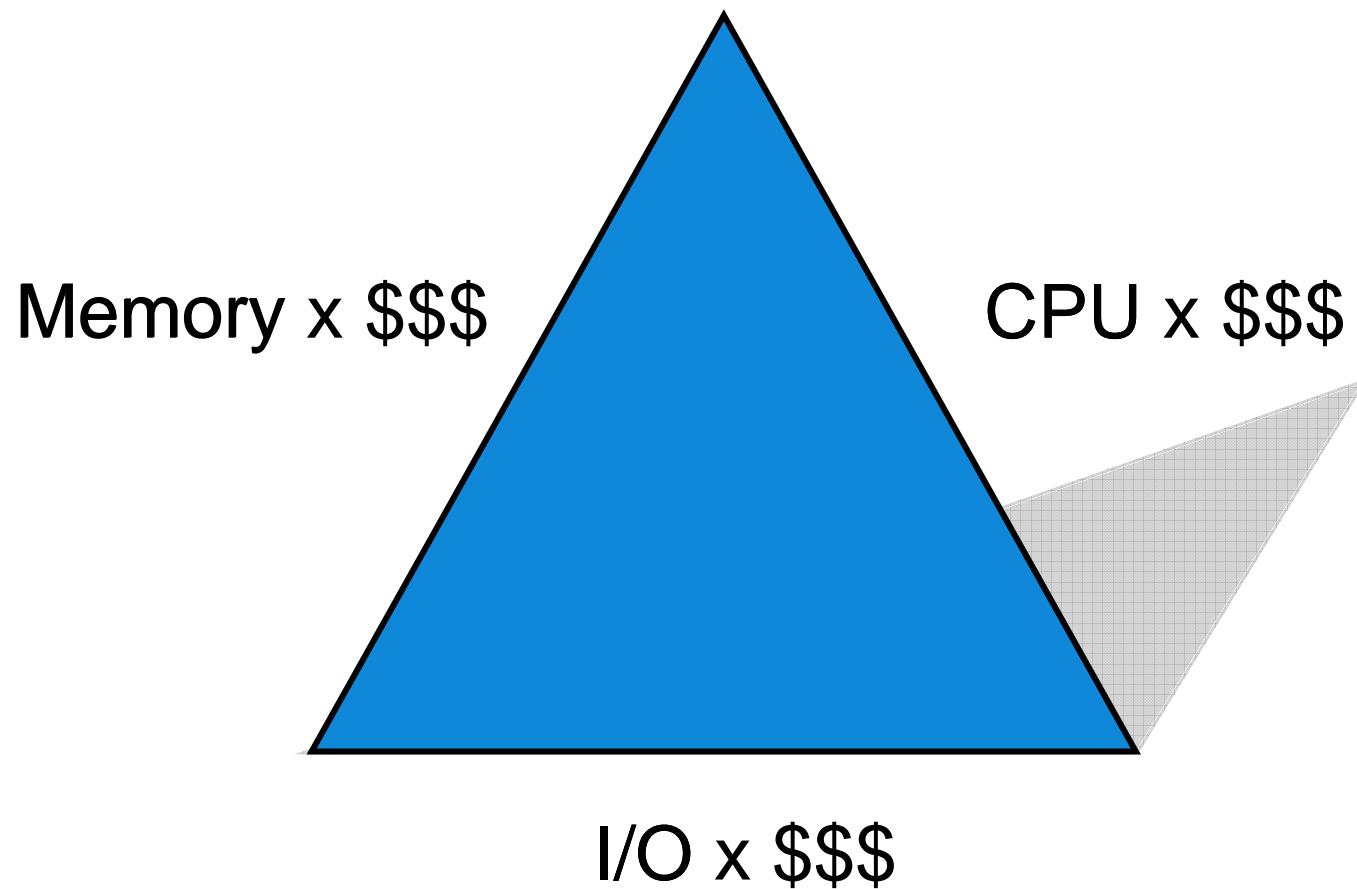


Intel Xeon system  
1TB memory  
10 TB table  
34 columns  
13.5 years data  
No indexes used

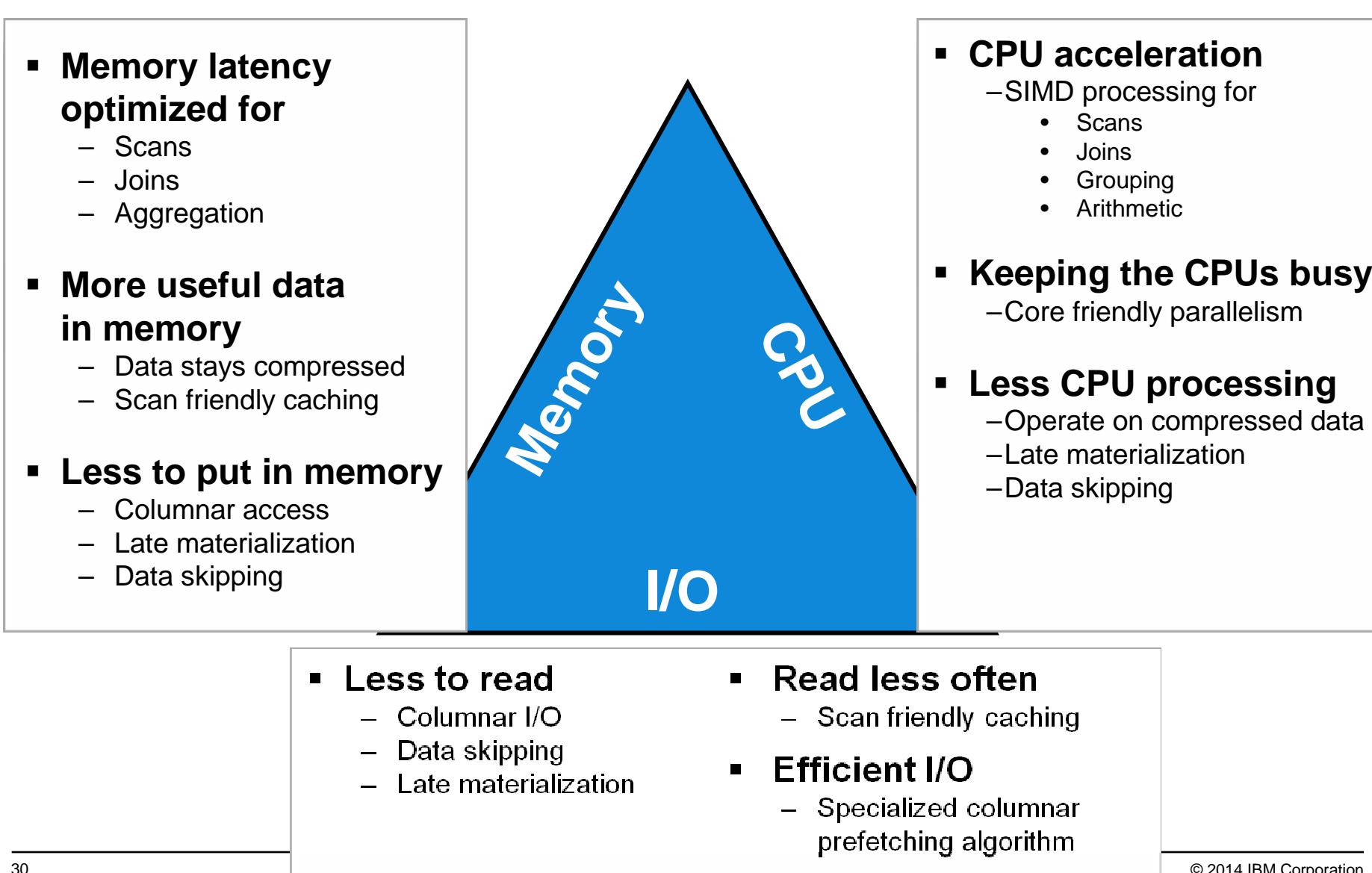
**SELECT COUNT\_BIG(\*) from DAILY\_SALES  
WHERE PERKEY => 1997001 AND PERKEY <= 1997091**



## Performance Triangle



# Optimizing All Sides of the Performance Triangle



---

# BLU Acceleration

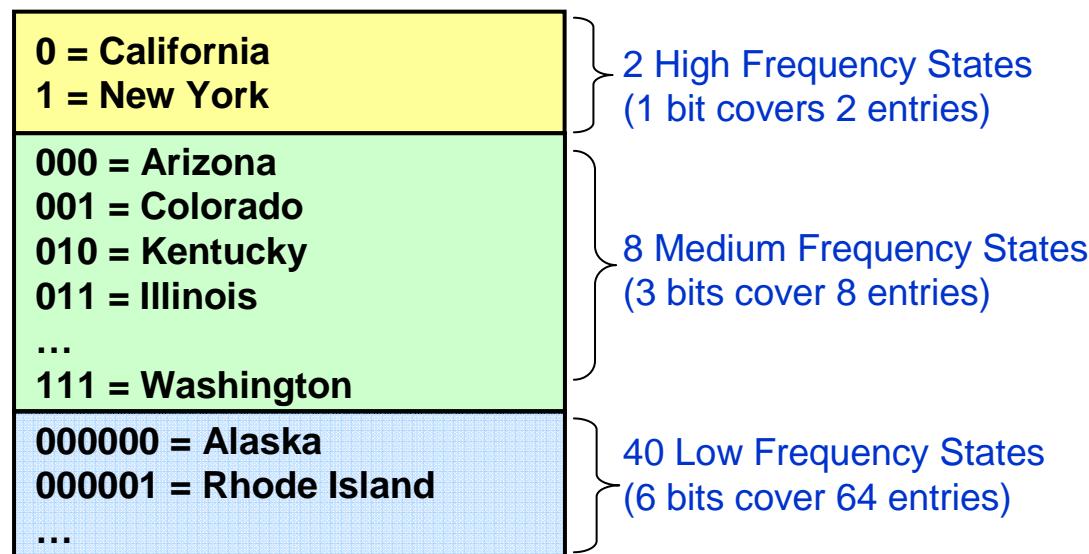
## More on Compression and the Synopsis Table



## Columnar Compression in DB2 10.5 BLU

- Frequency compression: Most common values use fewest bits
- Multiple compression techniques: Approximate Huffman-Encoding, prefix compression, and offset compression

Example showing 3 different code lengths. Code lengths vary depending on the data values.



- Exploiting skew in data distribution improves compression ratio
- Very effective since all values in a column have the same data type
- Maps entire values to dictionary codes

# Column-Level Compression Dictionaries

- **Column-level dictionaries: Always one per column**
  - Dictionary populated during Load Replace, Load Insert into empty table
  - Automatic Dictionary Creation during SQL Insert and Load Insert

**Column 1  
Compression  
Dictionary**

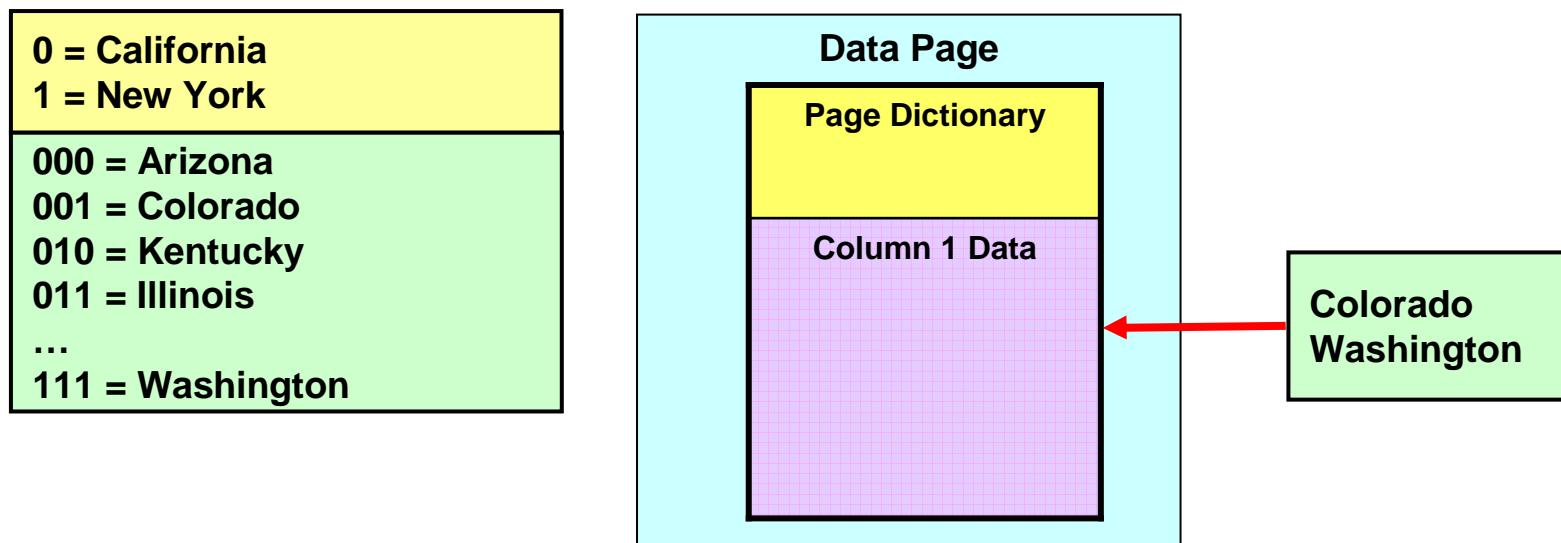
...

**Column N  
Compression  
Dictionary**

# Page-Level Compression Dictionaries

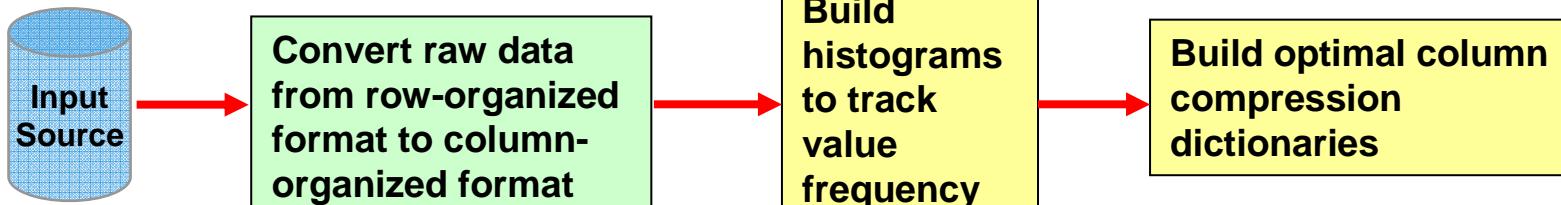
- Page-level dictionaries may also be created if space savings outweighs cost of storing page-level dictionaries
- Exploits local data clustering at page level to compress data even more than using column-level compression alone
- Allows compression to adapt and specialize as data changes over time
  - New values not covered by column-level dictionaries can still be compressed by page-level dictionaries
  - Reduces deteriorating compression ratio over time

## Column-Level Dictionary

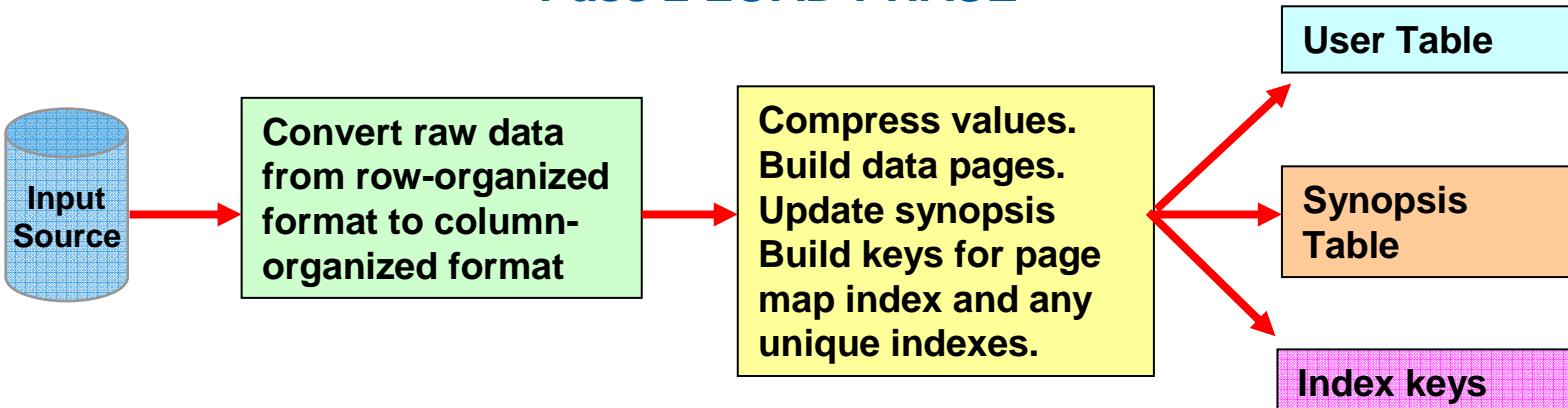


# Load for Column-Organized Tables

## Pass 1 ANALYZE PHASE Only if dictionaries need to be built



## Pass 2 LOAD PHASE



## LOAD Example

```
LOAD FROM /db1/svtdbm1/data.del OF DEL INSERT INTO colTable1;
```

SQL3109N The utility is beginning to load data from file "/db1/svtdbm1/data.del".

SQL3500W The utility is beginning the "**ANALYZE**" phase at time "04/15/2013 14:56:02.272825". SQL3519W Begin Load Consistency Point. Input record count = "0".

SQL3520W Load Consistency Point was successful.

SQL3515W The utility has finished the "**ANALYZE**" phase at time "04/15/2013 14:56:03.327893".

SQL3500W The utility is beginning the "**LOAD**" phase at time "04/15/2013 14:56:03.332048".

SQL3110N The utility has completed processing. "300000" rows were read from the input file.

SQL3519W Begin Load Consistency Point. Input record count = "300000".

SQL3520W Load Consistency Point was successful.

SQL3515W The utility has finished the "**LOAD**" phase at time "04/15/2013 14:56:04.639261".

SQL3500W The utility is beginning the "**BUILD**" phase at time "04/15/2013 14:57:06.848727".

SQL3213I The indexing mode is "REBUILD".

SQL3515W The utility has finished the "**BUILD**" phase at time "04/15/2013 14:59:07.487172".

Number of rows read = 300000 Number of rows skipped = 0 Number of rows loaded = 300000

Number of rows rejected = 0 Number of rows deleted = 0

Number of rows committed = 300000

## What You See in the DB2 Catalog: TABLEORG

- Which tables are column-organized?
  - New column in syscat.tables: TABLEORG

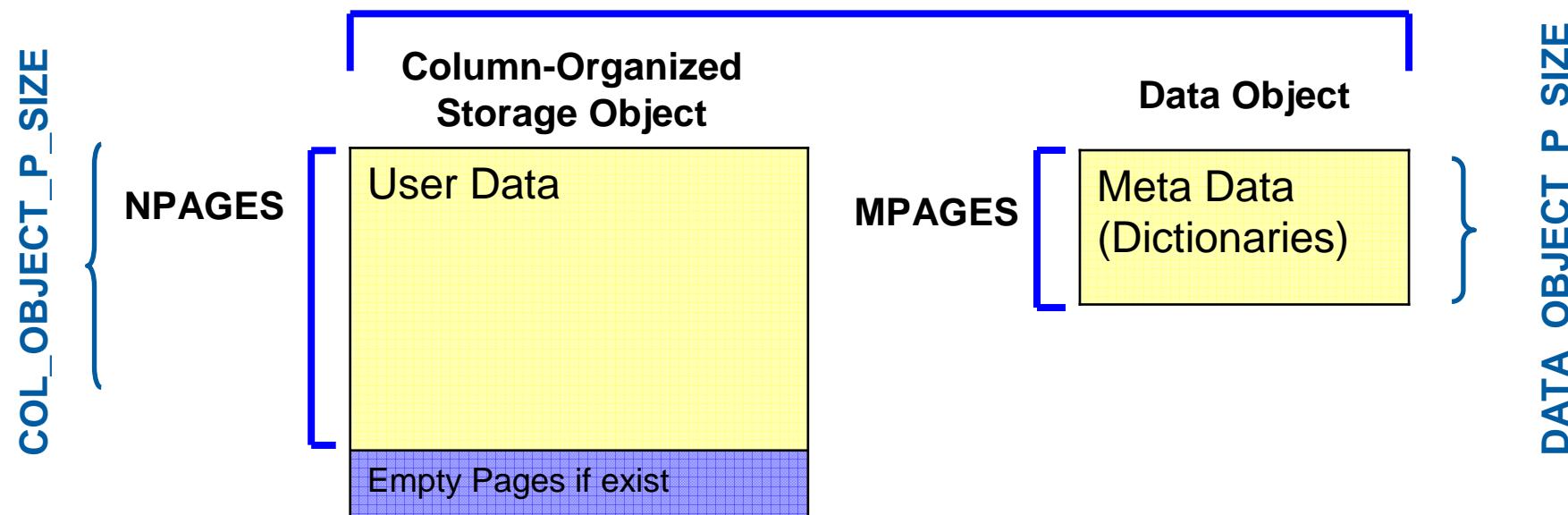
```
SELECT tablename, tableorg, compression
FROM syscat.tables
WHERE tablename like 'SALES%';
```

TABNAME	TABLEORG	COMPRESSION
-----	-----	-----
SALES_COL	C	
SALES_ROW	R	N

2 record(s) selected.

For column-organized tables, COMPRESSION is always blank because you cannot enable/disable compression.

# Measuring Compression



- **Statistics for measuring number of pages in SYSCAT.TABLES**
  - NPAGES: Number of pages in Column-Organized Object minus any empty pages
  - FPAGES: Total number of pages in both objects
  - MPAGES: (M for meta data) Number of pages in Data Object
- **ADMIN\_GET\_TAB\_INFO table function reports**
  - COL\_OBJECT\_P\_SIZE: Physical size of column data object containing user data
  - DATA\_OBJECT\_P\_SIZE: Physical size of data object containing meta data

# Calculating Column-Organized Storage Sizes

User Table	COL_OBJECT_P_SIZE
User Table + Meta Data + Page Map/Unique Indexes	COL_OBJECT_P_SIZE + DATA_OBJECT_P_SIZE + INDEX_OBJECT_P_SIZE

- **Be careful using NPAGES to determine table size**
  - May underestimate actual space usage especially for small tables
  - Doesn't take meta data or empty pages into account
- **Use the table function ADMIN\_GET\_TAB\_INFO or admin view ADMINTABINFO to retrieve**
  - COL\_OBJECT\_P\_SIZE + DATA\_OBJECT\_P\_SIZE + INDEX\_OBJECT\_P\_SIZE

## Table Compression Statistics in SYSCAT.TABLES

Row-Organized Table Statistics	Column-Organized Table Statistics
PCTPAGESAVED	PCTPAGESAVED
AVGCOMPRESSEDROWSIZE	
AVGROWCOMPRESSIONRATIO	
AVGROWSIZE	
PCTROWCOMPRESSED	

- Only **PCTPAGESAVED** applies to column-organized tables too
  - Approximate percentage of pages saved in the table
- Runstats collects **PCTPAGESAVED** by estimating the number of data pages needed to store table in uncompressed row orientation
  - ADMIN\_GET\_COMPRESS\_INFO not supported yet for column-organized tables and will return zero rows

## PCTENCODED Statistic in SYSCAT.COLUMNS

C1	PCTENCODED = 90
C2	PCTENCODED = 75
C3	PCTENCODED = 100

- Percentage of values encoded (compressed) by column-level dictionary
- It measures number of values compressed NOT compression ratio
- Each column could have a different encoded percentage
- PCTENCODED is a lower bound
  - Additional compression possible via page-level dictionaries (even when PCTENCODED=0)
  - Used in heuristic decisions for performing join or group by on either encoded or unencoded data

## Compression Summary

- Storage optimization through DB2 compression can save 60%-75% of your total database storage requirements
- In real customer examples, storage savings are realized along with improved performance
- DB2 9.7 saves even more with compression for indexes, temp tables and XML data
- DB2 10 delivers adaptive data compression capabilities with up to 7x storage savings for tables
- DB2 10.5 BLU Acceleration further improves compression and removes need for indexes and aggregates to save even more space
  - 2-3x storage savings over adaptive compression is common
  - Several customers have reported up to 10-25x storage reduction vs. uncompressed row tables

## What You See in the DB2 Catalog: Synopsis Tables

- For each columnar table, there is a corresponding synopsis table, automatically created and maintained

```
SELECT tabschema, tablename, tableorg
FROM syscat.tables
WHERE tableorg = 'C';
```

TABSCHEMA	TABNAME	TABLEORG
MNICOLA	SALES_COL	C
SYSIBM	SYN130330165216275152_SALES_COL	C

2 record(s) selected.

- Size of the synopsis table: ~0.1% of the user table
- 1 row for every 1024 rows in the user table

# Synopsis Table

- Meta-data that describes which *ranges* of values exist in which parts of the user table

**SYN130330165216275152\_SALES\_COL**

TSNMIN	TSNMAX	S_DATEMIN	S_DATEMAX	...
0	1023	2005-03-01	2006-10-17	...
1024	2047	2006-08-25	2007-09-15	...
...				

TSN = Tuple Sequence Number

User table: **SALES\_COL**

S_DATE	QTY	...
2005-03-01	176	...
2005-03-02	85	...
2005-03-02	267	
2005-03-04	231	
...		
...		
...		
...		
...		
...		
...		

- Enables DB2 to skip portions of table when scanning data during query
- Predicate WHERE S\_DATE = 2007-01-01 would skip first range

---

# BLU Acceleration Query Execution Plans



## Query Execution in DB2 with BLU Acceleration

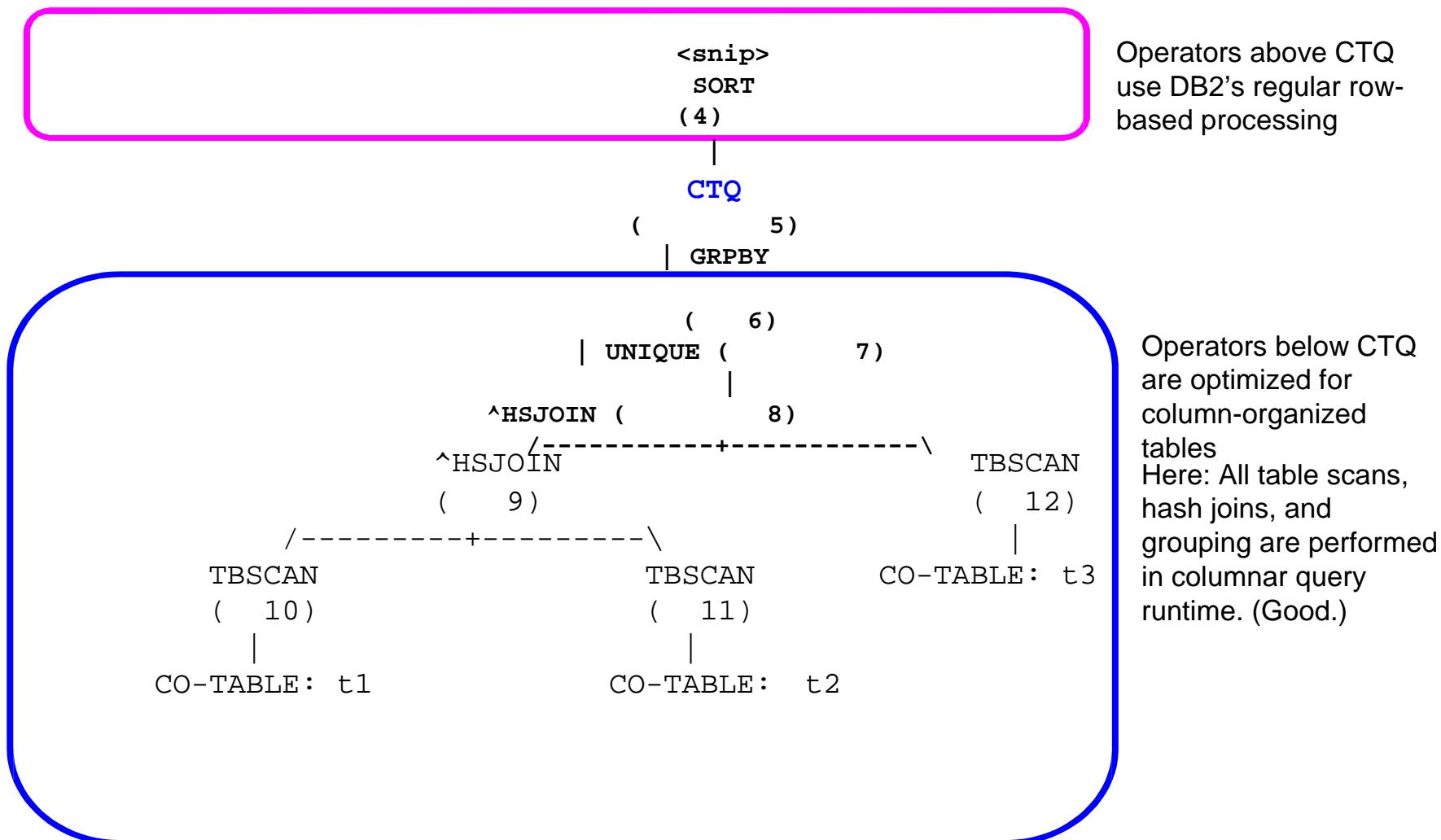
- BLU Acceleration is more than columnar storage!
- BLU Acceleration = columnar data storage
  - + columnar query runtime
  - + many other optimizations
- New operator CTQ in execution plans
- CTQ transfers data between operators specializing in column-organized and row-organized data

## Sample Query

```
SELECT t2.c4  
FROM t1, t2, t3  
WHERE t1.c1 = t2.c1  
      AND t1.c2 = t3.c2  
      AND t1.c3 = 0  
GROUP BY t2.c4  
ORDER BY t2.c4
```

Let's review the execution plan of this query....

## Sample Execution Plan



Operators above CTQ  
use DB2's regular row-based processing

Operators below CTQ  
are optimized for  
column-organized  
tables  
Here: All table scans,  
hash joins, and  
grouping are performed  
in columnar query  
runtime. (Good.)

## Execution Plans

- Runtime operators are optimized for row- and column-organized tables
- CTQ operator transfers data from column- to row-organized processing
- Operators that are optimized for column-org tables below CTQ include
  - Table scan
  - Hash-based join, optionally employing a semi-join
  - Hash-based group by. Potentially faster without the sort
  - Hash-based unique
- Aim is to “push down” most operators below CTQ

# Tooling Assist



# IBM Optim Query Workload Tuner

This page shows the recommendations from the advisors that you ran.

Database connection:  TPCDSANv10.2hotel67 ( DB2 for Linux, UNIX, and Windows V10.5.0 )

▶ Status/Description

Statements Summary Table organization Candidate Table Organization

Estimated performance improvement: 83.44 %

Number of tables referenced in the workload: 11 Number of tables recommended for conversion: 11

Show DDL Script Test Candidate Table Organization Filter by Tables to be converted

Table	Creator	Current Organization	Recommended Organization	Conversion Warning
HOUSEHOLD_DEMOG...	TPCDS	ROW	COLUMN	Indexes will be removed
DATE_DIM	TPCDS	ROW	COLUMN	Indexes will be removed
WEB_SALES	TPCDS	ROW	COLUMN	Indexes will be removed
STORE	TPCDS	ROW	COLUMN	Indexes will be removed
STORE_CALLS	TPCDS	ROW	COLUMN	Indexes will be removed

Advisor identifies candidate tables for conversion to columnar format.

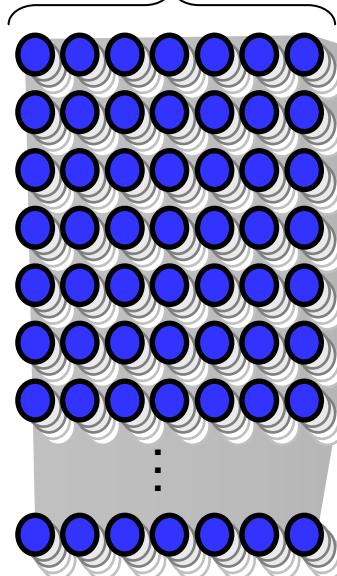
Analyzes SQL workload and estimates execution cost on row- and column-organized tables.

## 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`

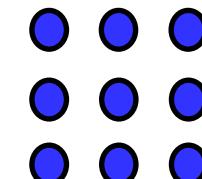
### Applications and Users

Up to tens of thousands of SQL queries at once



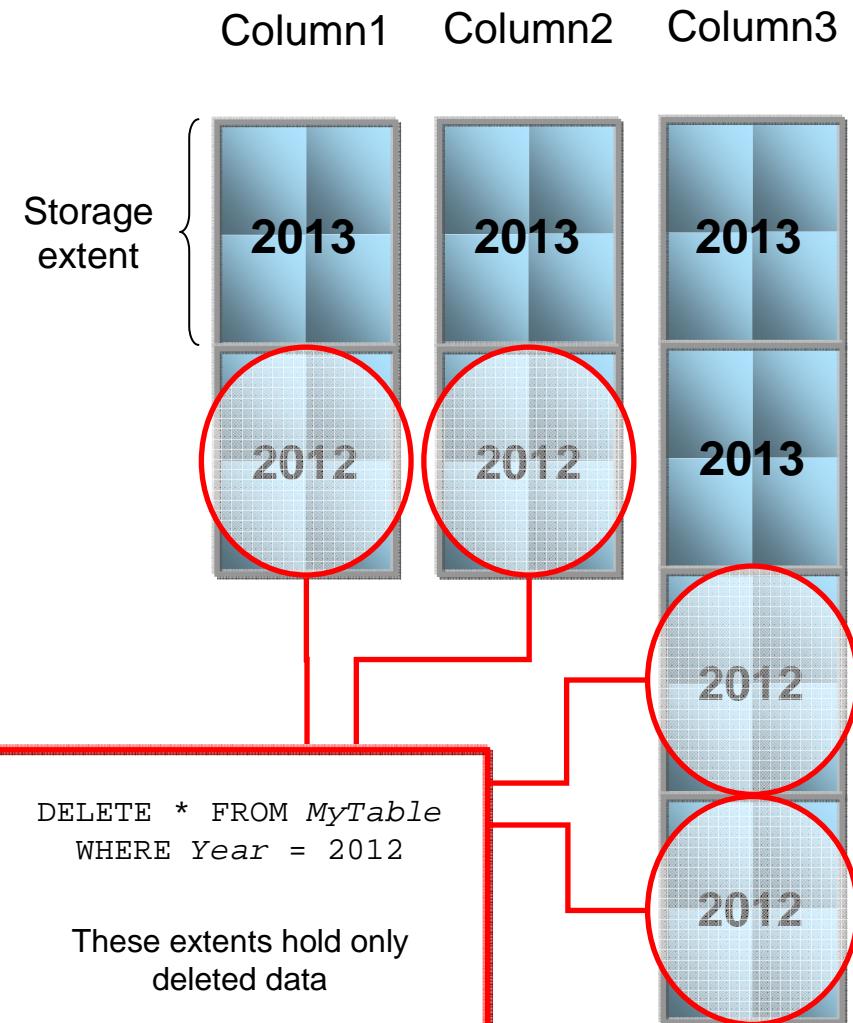
### DB2 DBMS kernel

Moderate number of queries consume resources



# Automatic Space Reclaim

- **Automatic space reclamation**
  - Freed extents with no active values
  - The storage can be subsequently reused by any table in the table space
- **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**



# Full Exploitation: Core Fabrication to Data Delivery

## ▪ Deep Processor Exploitation

- DB2 has deep exploitation of Simultaneous Multi Threading (SMT), NUMAtization, +++
- Key POWER7 value proposition is the ability to dispatch a huge number of threads
  - DB2 moved to fully threaded engine (from process based) to exploit this capabilities at day 1
- DB2 Decimal arithmetic performed directly on the DECFLOAT accelerator



## ▪ Deep Memory Exploitation

- Autonomic detection and exploitation of POWER features such as larger pages, storage keys
- Alternative page cleaning algorithms built into DB2 specifically for AIX performance boost



## ▪ Automatic Storage Exploitation

- Exploits Async I/O, Scatter/Gather I/O, CIO, DIO interfaces, Atomic Logical Volumes, +++

## ▪ Policy based workload management from application to AIX execution

- DB2 WLM exploits AIX WLM within its own workload policies
- Other vendors work in silos (either AIX WLM or Database's WLM, but not both)



## ▪ Result is a decade long run of proven performance leadership

- Apples-to-Apples benchmarks consistently show DB2/POWER 16-35% faster than Oracle
- Consistently best per core performance versus Intel: total days of TPC-C leadership belongs to POWER

## ▪ And now...DB2 with BLU Acceleration, the ONLY in AIX in-memory columnar database (not to mention optimized for POWER)



## DB2 10.5 BLU Optimizations Specifically for Power

- **Columnar encodings of data are stored in chunks that map to the size of the registers**
  - Power7+ has more registers than Intel (64 vs. 16)
  - AIX also has 2 pipes for processing SIMD requests
  - Results in more data being loaded in registers for higher performance
- **BLU leverages SIMD processing instructions for better performance**
  - Leveraging the above registers to perform more comparisons per cycle
  - Better performance by exploiting Power7 SIMD capabilities
- **DB2 runs a separate binary library optimized specifically for Power7+ if DB2 detects Power7+ architecture at installation/upgrade time**
  - Compiler directives that take advantage of specific Power7 capabilities
  - Also using compiler directives that allows instructions to be scheduled so they are optimal for Power7+

# DB2 10.5 with BLU Acceleration with Cognos BI performance is ‘Fast on Fast’



Efficient, affordable, and improved storage savings for Cognos BI customers

**18X**

Faster cube load



Make better and timelier business decisions

**20X**

Faster DB Query



DB2 with BLU Acceleration complements and enhances IBM Cognos BI

**10X- 25X**

Improvement is common\*

\*Client-reported testing results in DB2 10.5 early release program. Individual results will vary depending on individual workloads, configurations and conditions.

---

# DB2 10.5 BLU Acceleration Power 8 Exploitation



# Extreme Performance via Deep Power8 Exploitation

- **Faster performance for financial calculations**

- Decimal arithmetic using new vector based instructions
  - Row based tables benefit from vector processing on decimal data

- **Improved integrity and reliability**

- Leverage new Power8 algorithms for high speed memory integrity checking
  - Increased processing performance while ensuring a higher level of integrity for data pages

- **Optimizations for increased concurrency**

- Power8 will support twice the threading of Power7
  - Can result in software contention if not optimized for
  - DB2 10.5 FP4 exploits low level AIX latching algorithms to improve the concurrency of these extremely highly threaded servers

# Extreme Performance via Deep Power8 Exploitation

- **Cognitive compilation**

- When compiling and optimizing DB2 runtime code, IBM uses special cognitive algorithms that watch DB2 processing BLU acceleration workloads
- This learning is then used to reorder instructions within the product for even faster runtime performance

- **Faster range predicates for BLU tables**

- Power8 has new instructions that can be exploited by SIMD aware applications
- DB2 will leverage these new instructions for range predicates to evaluate many more column values simultaneously compared to Power7 or Intel
- Resulting in even greater performance and faster analytics

# Is BLU Acceleration Unique?

- Competitors exist – Oracle Exadata, SAP HANA, etc.
- No competitor comes close in abilities as BLU Acceleration
- Usability and flexibility
  - Columnar support on AIX
  - Data accessed can be larger than available memory!
  - Flexible deployment with full row- and column-organized tables available
  - Encryption support
  - Security – Full separation of Duties
- Simplicity
  - Load And Go!
  - No indexes for performance
  - No decisions for compression – automatic
  - Automatic Workload Management
  - Full graphical performance monitoring management tooling
- Performance
  - Maintain column-organization in memory – late materialization
  - Actionable compression – no need to uncompress to work on data
  - INSERT/UPDATE done directly into column-organized format
  - Data Skipping – Maintain data value mapping for query data skipping
  - Multiply the power of the processor with SIMD
  - Allow constraints to be defined but not enforced on columnar data for performance
  - Allow uniqueness to be enforced on columnar data

# BLU Acceleration in the Cloud

Visit: <http://bluforcloud.com>



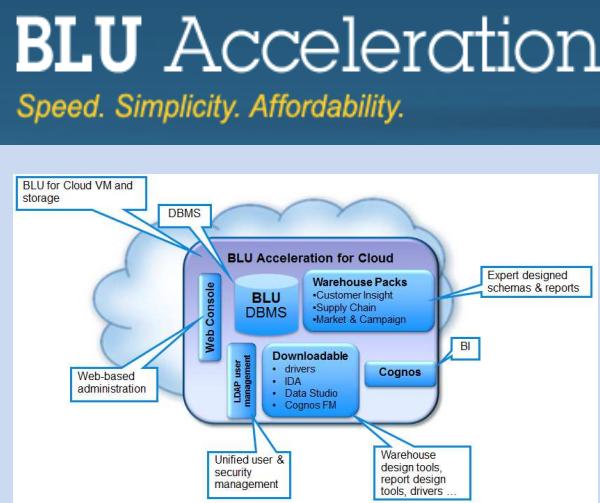
Self-service BI and data warehousing on the cloud

**BLU Acceleration for Cloud – in minutes**

- Purchasing + provisioning + boot: **20-30 minutes** to a fully configured system
- Create your schema and load data

1. In under an hour anyone can access data awesome warehousing and BI for less than a cup of coffee
2. No infrastructure or IT resources

# Offerings and Deployment Models

Pure Systems	Cloud	Software
<p>Pure Application System</p>  <p>PureApplication Application Platform Delivering Platform Services</p>	<p>BLU Acceleration for the Cloud</p>  <p><b>BLU Acceleration</b> <i>Speed. Simplicity. Affordability.</i></p> <p>The diagram illustrates the BLU Acceleration for Cloud architecture. At the center is a cloud icon labeled "BLU Acceleration for Cloud". Surrounding it are various components: "DBMS" (top left), "Web Console" (left), "Warehouse Packs" (right), "Cognos" (bottom right), "Downloadable" (bottom left), and "Unified user &amp; security management" (bottom center). Arrows point from each component to their respective labels. Callouts provide additional details: "BLU for Cloud VM and storage" points to the DBMS, "Expert designed schemas &amp; reports" points to the Warehouse Packs, and "Warehouse design tools, report design tools, drivers ..." points to the Cognos icon.</p>	<p>DB2 10.5 Advanced Workgroup Advanced Enterprise</p> 
<p>IBM Business Intelligence Pattern with BLU Acceleration</p> <p>IBM DB2 Data Mart with BLU Acceleration</p>	<p>Pay by the hour for 1TB or 10TB Use your credit card Bring your own license</p>	<p>Cognos BI 10.2</p>  <p>10 COGNOS</p> <p>DB2 10.5 Advanced Editions include 5 user licenses of Cognos</p>

# IBM BLU Acceleration



## ✓ **SPEED: Dramatically Faster Reporting and Analytics**

- In-memory processing eliminates disk scan
- Column store retrieves relevant data
- Maximized CPU power speeds processing
- Data skipping for more efficient data retrieval



## ✓ **SIMPLICITY: Fast time to value**

- Requires only a database software upgrade
- Create tables, load data, run applications
- No application or schema changes required
- No data modeling, indexes, tuning or MQTs required



## ✓ **EFFICIENCY: Unprecedented Affordability**

- Exploits infrastructure you already have
  - No special hardware appliance required
- Smarter memory management
  - Data still compressed while in memory
  - Only relevant data in memory, not ALL data

## DB2 10.5 pureScale



# DB2 10.5 pureScale Enhancements

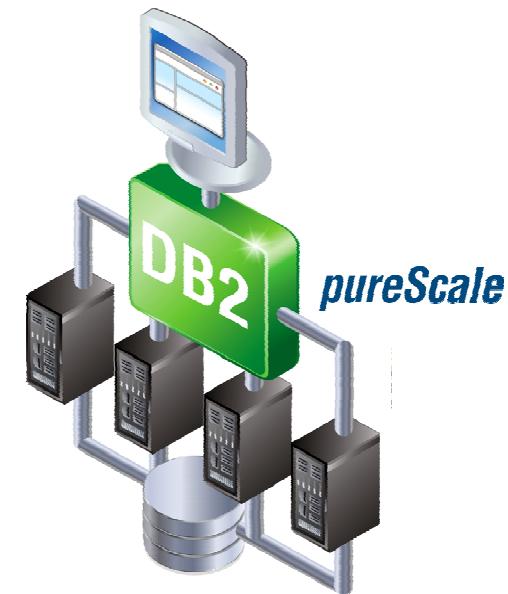
## *Enhanced availability, optimized for OLTP Workloads*

- **DB2 pureScale**

- Robust infrastructure for OLTP workloads
- Provides improved availability, performance, and scalability
- Transparent scalability beyond 100 nodes<sup>1</sup>
- Leverages z/OS cluster technology

- **NEW pureScale enhancements**

- Online member add
- HADR designed to failover in seconds
- Multi-tenancy: Member subsets
- Multi-tenancy: Explicit Hierarchical Locking (FP1)
- Topology changing backup and restore



1. Available with DB2 Advanced Enterprise Server Edition.
2. Based on IBM design for normal operation with rolling maintenance updates of DB2 server software on a pureScale cluster. Individual results will vary depending on individual workloads, configurations and conditions, network availability and bandwidth.
3. Based on IBM design for normal operation under typical workload using HADR and pureScale clusters. Individual results will vary depending on individual workloads, configurations, and conditions, network availability and bandwidth.

---

# DB2 10.5 Oracle Compatibility



# Oracle Compatibility Built into DB2

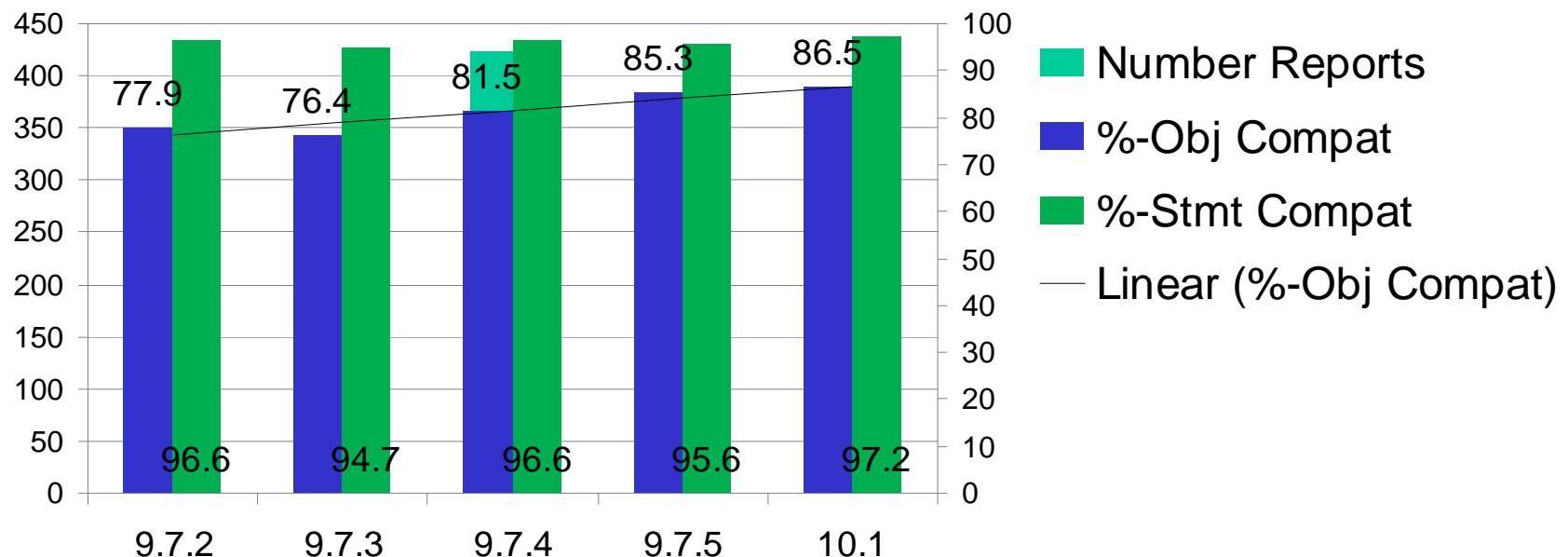
*Lower Transition Cost and Less Risk*

---

<b>Concurrency Control</b>	→ Native support
<b>Oracle SQL dialect</b>	→ Native support
<b>PL/SQL</b>	→ Native support
<b>PL/SQL Packages</b>	→ Native support
<b>Built-in package library</b>	→ Native support
<b>Oracle JDBC extensions</b>	→ Native support
<b>OCI</b>	→ Native support
<b>Oracle Forms</b>	→ Through partners
<b>SQL*Plus Scripts</b>	→ Native support
<b>RAC</b>	→ DB2 pureScale

**Changes are the exception. Not the rule.**

## Application Compatibility Over Time



- Data is based on DCW (Database Conversion Workbench) DB2 reports in the database
- Compatibility is improved
  - More and more complex applications
- DB2 10.5 provides > 98% compatibility

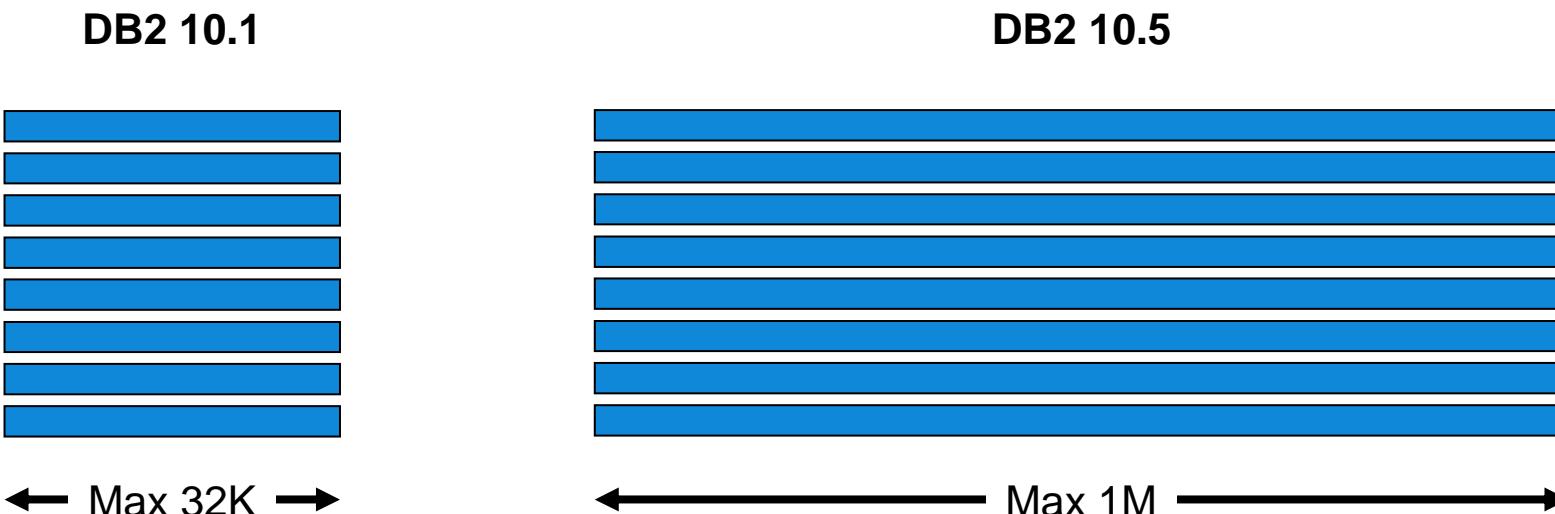
# Oracle Compatibility: Larger Row Widths

- Accommodate larger strings

- Allow tables with up to 1MB wide rows

```
CREATE TABLE emp(name      VARCHAR(4000),  
                 address   VARCHAR(4000),  
                 cv        VARCHAR(32000))
```

- Allow large row GROUP BY and ORDER BY as long as key can sort
  - SYSTABLES PCTEXTENDEDROWS column shows % of rows in a table that are extended



# Oracle Compatibility: Additional Indexing

## ▪ Function-based indexes

- Searching for computed values in a table instead of using Generated Columns
- E.g. “Find employees without worrying about the case of their names”

- ```
CREATE INDEX emp_name ON emp(UPPER(name)) ;
SELECT salary
FROM emp
WHERE UPPER(name) = 'MCKNIGHT' ;
```

## ▪ Indexes excluding NULL keys

- Enforce uniqueness only for non-NULL keys and exclude all NULL keys from Index
- Compress index for all-NULL keys
- Helps facilitate Oracle application migrations

- ```
CREATE UNIQUE INDEX emp_manages
ON emp(manages) EXCLUDE NULL KEYS
```

Name	Salary	Manages
McKnight	50000	Sales
Miller	25000	-
Van Gogh	45000	Finance
Chan	37000	-

## ▪ Random key indexes

- Avoid hot index page for incrementally issued keys
- ```
CREATE UNIQUE INDEX order_id ON order(id RANDOM);
```

# Oracle PL/SQL Compatibility

- **Create distinct type with weak type rules**

- Removes limitation of existing distinct types not having weak typing
- Optional check constraint
- Optional NOT NULL constraint
- Constraints enforced on assignment

- **Pipelined table function**

- Introduce a new PIPE statement which returns a row to caller, but continues at next statement if caller wants another row
- Incrementally produce a result set for consumption on demand

- **Ad-hoc federated table access**

- Support ad-hoc reference to remote table using server in the identifier
  - Reach out to a table in a remote database

- **Function library extensions**

- Updates to various built-in functions for improved compatibility support



Officially Supported in Fix Pack #1



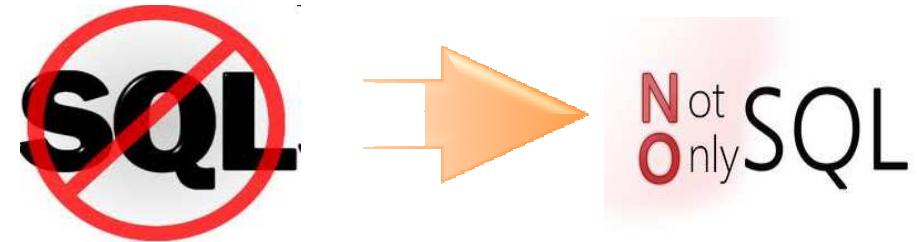
IBM

## JSON Technology



## Background – What is NoSQL

- **A class of database management systems that depart from traditional RDBMSs**
  - Does not use SQL as the primary query language
  - Is “schema-less”
    - No rigid schema enforced by the DBMS
  - Programmer-friendly for adding fields to a document
  - Might not guarantee full ACID behavior
  - Often has a distributed, fault-tolerant, elastic architecture
  - Highly optimized for retrieve and append operations over great quantities of data



**NoSQL DEFINITION:** Next Generation Databases mostly addressing some of the points: being non-relational, distributed, open-source and horizontally scalable.

The original intention has been **modern web-scale databases**. The movement began early 2009 and is growing rapidly. Often more characteristics apply such as: **schema-free, easy replication support, simple API, eventually consistent / BASE** (not ACID), a **huge amount of data** and more. So the misleading term "*nosql*" (the community now translates it mostly with "**not only sql**") should be seen as an alias to something like the definition above. [based on 7 sources, 14 constructive feedback emails (thanks!) and 1 disliking comment . Agree / Disagree? [Tell me so!](#) By the way: this is a strong definition and it is out there here since 2009!]

**LIST OF NOSQL DATABASES** [currently 150]

Emergence of a growing number of non-relational, distributed data stores for massive scale data

# Background - What is JSON?

## ▪ JavaScript Object Notation

- Serialized form of JavaScript Objects
  - Lightweight data interchange format
  - Specified in IETF RFC 4627
  - <http://www.JSON.org>

## ▪ Lightweight text interchange

- Designed to be minimal, portable, textual, and subset of JavaScript
  - Only 6 kinds of values!
  - Easy to implement and easy to use

## ▪ Replacing XML as the de facto data interchange format on the web

- Used to exchange data between programs written in all modern programming languages

## ▪ Self-describing, easy to understand

- Text format, so readable by humans and machines
- Language independent, most languages have features that map easily to JSON

```
{  
  "firstName": "John",  
  "lastName" : "Smith",  
  "age"      : 25,  
  "address"  :  
  {  
    "streetAddress": "21 2nd Street",  
    "city"        : "New York",  
    "state"       : "NY",  
    "postalCode"  : "10021"  
  },  
  "phoneNumber":  
  [  
    {  
      "type"  : "home",  
      "number": "212 555-1234"  
    },  
    {  
      "type"  : "fax",  
      "number": "646 555-4567"  
    }  
  ]  
}
```

*“Less is better: less we need to agree upon to interoperate, the more easily we interoperate”*  
JavaScript: The Good Parts, O'Reilly

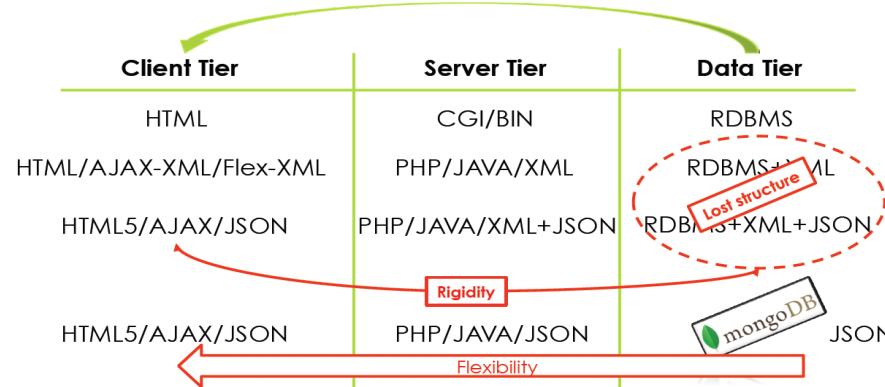
# The JSON-XML Shift

- Developers find it easier to move data back and forth without losing information in JSON vs. XML
  - XML is more powerful and more sophisticated than JSON
  - But JSON found to be 'good enough' → It makes programming tasks easier
- By the time RDBMS world got very sophisticated with XML, developers had chosen JSON
  - Application shift lead to emergence of database that store data in JSON (i.e., MongoDB)
  - JSON on the server side is appealing for developers using JSON on the client tier side



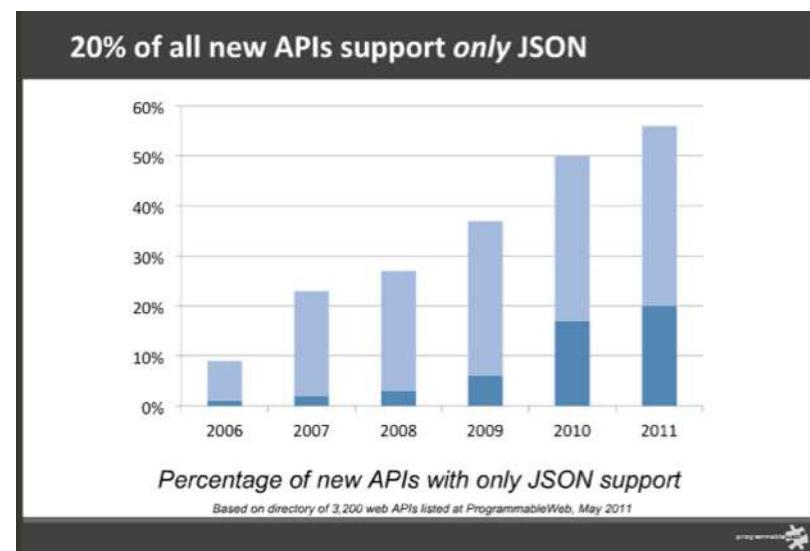
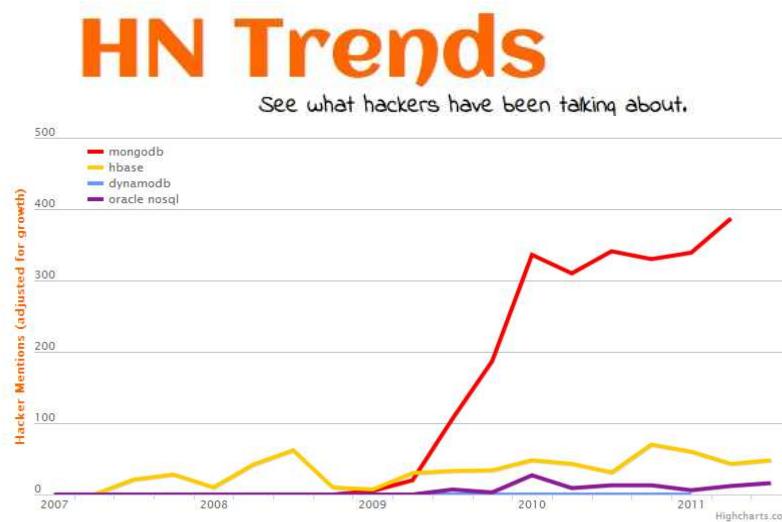
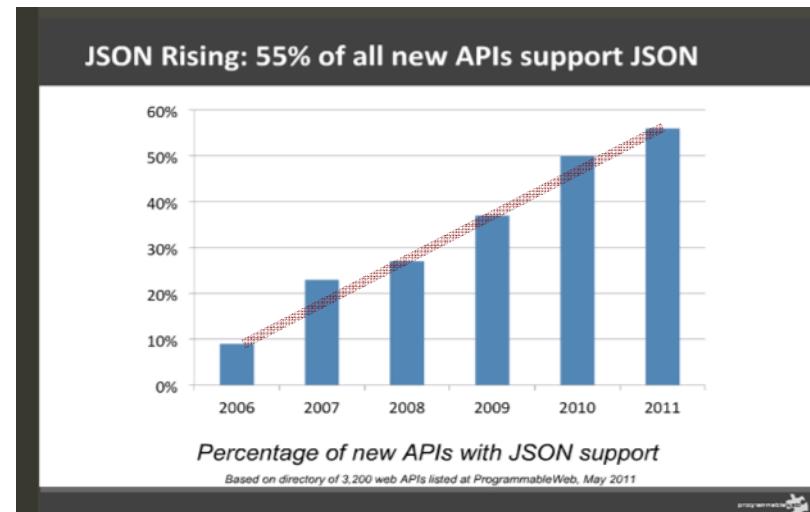
## The Technology

- Each technology is awesome & powerful in itself But...
- ...Mixing technologies and solving integration issues are not



# Open APIs State of the Market

- **JSON is the new cool**
  - XML declining: 5 years ago hardly any JSON
  
- **Why? JSON is**
  - Less verbose and smaller docs size
  - <Mytag>value</Mytag> vs. Mytag:value
  - Tightly integrated with JavaScript which has a lot of focus
  - Most new development tools support JSON and not XML





## JSON Technology in DB2 for LUW

- **Combine data from systems of engagement with traditional data in same DB2 database**

- Best of both worlds
- Simplicity and agility of JSON + enterprise strengths of DB2



- **Store data from web/mobile apps in it's native form**

- New web applications use JSON for storing and exchanging information
- It is also the preferred data format for mobile application backends



- **Move from development to production in no time!**

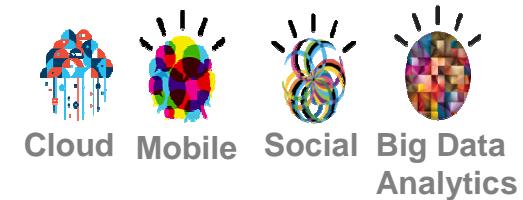
- Ability to create and deploy flexible JSON schema
- Gives power to application developers by reducing dependency on IT; no need to pre-determine schemas and create/modify tables
- Ideal for agile, rapid development and continuous integration





## JSON Technology in DB2 for LUW (cont.)

- DB2 for Linux, UNIX, and Windows now officially supports JavaScript Object Notation (JSON) DB2 NoSQL capability
  - No longer in technology preview
  - You can now store and manage JSON data in a DB2 database
    - You can create dynamic applications by using JSON's schemaless NoSQL capability. In addition to basic NoSQL operations on collections of JSON documents, this release includes support for transactions control and bi-temporal data awareness
    - JSON documents can be interfaced in the following three ways
      - DB2 JSON Java API
      - DB2 JSON command-line interface
      - DB2 JSON wire listener
  - For further details, see *JSON application development support has been added* section in the Information Center



# DB2 10.5 Packaging Simplification



# DB2 10.5 Simplifies Product Packaging

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

|                   |                                                                                                                                                                                                                                                                                                   | Departmental Market                                                                                                                                                                                                                                                       | Enterprise Market |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|
| Advanced function | DB2 Advanced Workgroup Server Edition                                                                                                                                                                                                                                                             | DB2 Advanced Enterprise Server Edition                                                                                                                                                                                                                                    |                   |
|                   | <ul style="list-style-type: none"> <li>For small OLTP and analytic deployments</li> <li>Primarily used in department environments within large enterprises or SMB/MM deployments</li> <li>Limited by TB, memory, sockets and cores</li> <li>Supports BLU, pS and DPF deployment models</li> </ul> | <ul style="list-style-type: none"> <li>For Enterprise Class OLTP and/or analytic deployments</li> <li>Targeting full enterprise/full data centre requirements</li> <li>No TB, memory, socket or core limit</li> <li>Supports BLU, pS and DPF deployment models</li> </ul> |                   |
| Core function     | DB2 Workgroup Server Edition                                                                                                                                                                                                                                                                      | DB2 Enterprise Server Edition                                                                                                                                                                                                                                             |                   |
|                   | <ul style="list-style-type: none"> <li>Entry level offering</li> <li>Single server for less intense workloads</li> <li>Limited by TB, memory, sockets and cores</li> <li>No support for BLU, pS or DPF deployment models</li> </ul>                                                               | <ul style="list-style-type: none"> <li>Entry level offering</li> <li>Single server for enterprise/more intense workloads</li> <li>No TB, memory, socket or core limit</li> <li>No support for BLU, pS or DPF deployment models</li> </ul>                                 |                   |
| Limited capacity  |                                                                                                                                                                                                                                                                                                   | Full capacity                                                                                                                                                                                                                                                             |                   |
|                   |                                                                                                                                                                                                                                                                                                   | DB2 Developer Edition                                                                                                                                                                                                                                                     | DB2 CEO           |
|                   |                                                                                                                                                                                                                                                                                                   | DB2 Express and DB2 Express-C                                                                                                                                                                                                                                             | DB2 Advanced CEO  |

# DB2® 10.5

## *with BLU Acceleration*

**Multi-workload database  
software for the era of big data**

- ***Always Available Transactions***

Disaster recovery of pureScale clusters over distances of 1000s km<sup>1</sup>; means minimal downtime

- ***Faster Analytics***

In-memory hybrid technology yields performance improvements ranging from 8-25x performance improvements<sup>2</sup>, without costs or limits of in-memory only

- ***Unprecedented Compatibility***

An average of 98% Oracle database application compatibility<sup>3</sup>

- ***Future-Proofed Infrastructure***

NoSQL support allows clients to expand and modernize their apps



*"Before we made a final decision we benchmarked some of the key database management systems. That includes Oracle, SQL Server and DB2. We ended up choosing DB2 for several reasons. One was reliability, second was performance and perhaps the most important factor was ease of use"*

*– Bashir Khan, Director of Data Management and Business Intelligence*

---

# Thank You!



**Bob Harbus**  
**Global IM Technical Sales and Competitive Database**  
**IBM Toronto Laboratory**  
**rharbus@ca.ibm.com**