IBM

DB2 BLU inside out

December 6th, 2013

> “Super analytics made super easy.”

Jens Seifert
Jens.Seifert@de.ibm.com
IBM Deutschland Research & Development GmbH, Böblingen

Important Disclaimer

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 standard 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.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

Trademarks

IBM, the IBM logo, ibm.com, AIX and DB2 are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Windows is a trademark of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries

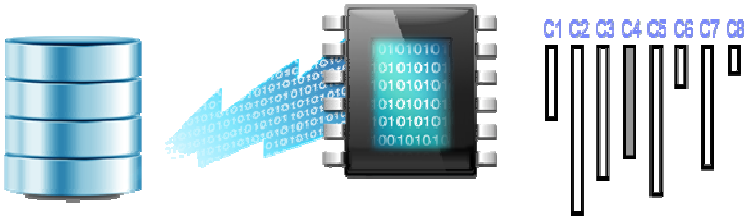
Other company, product, or service names may be trademarks or service marks of others.

SAP, SAP NetWeaver, SAP Business Information Warehouse, SAP BW, SAP NetWeaver BW, SAP ERP and other SAP products and services mentioned herein are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

Overview

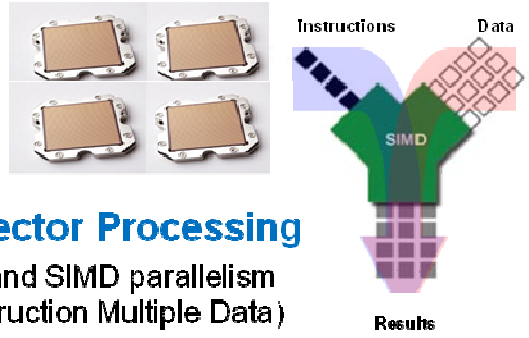
Key ideas

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

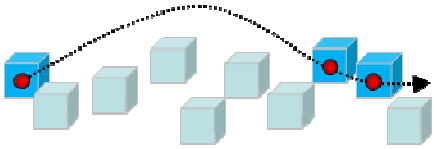


- **Actionable Compression**
  Patented compression technique that preserves order so that the data can be used without decompressing

Encoded



- **Parallel Vector Processing**
  Multi-core and SIMD parallelism (Single Instruction Multiple Data)



- **Data Skipping**
  Skips unnecessary processing of irrelevant data

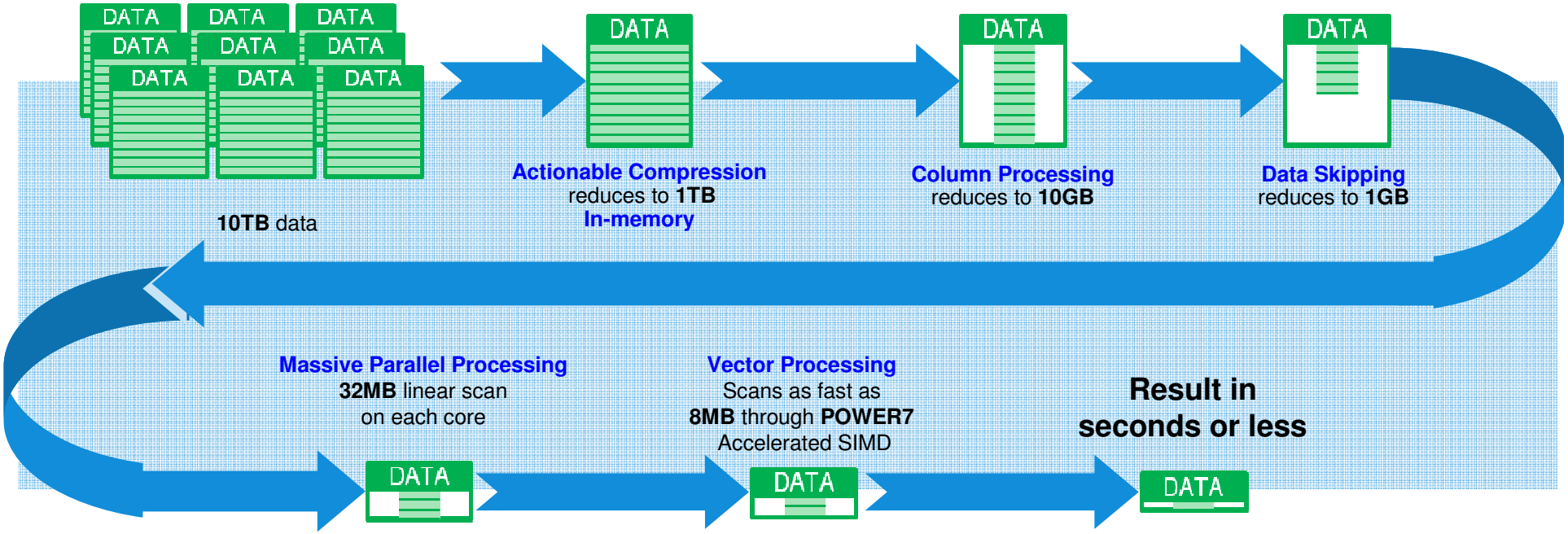
Super Fast, Super Easy — *Create, Load and Go!*

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

BLU Acceleration: 10TB Query, Seconds or Less

32 cores
1TB memory
10TB table
100 columns
10 years data

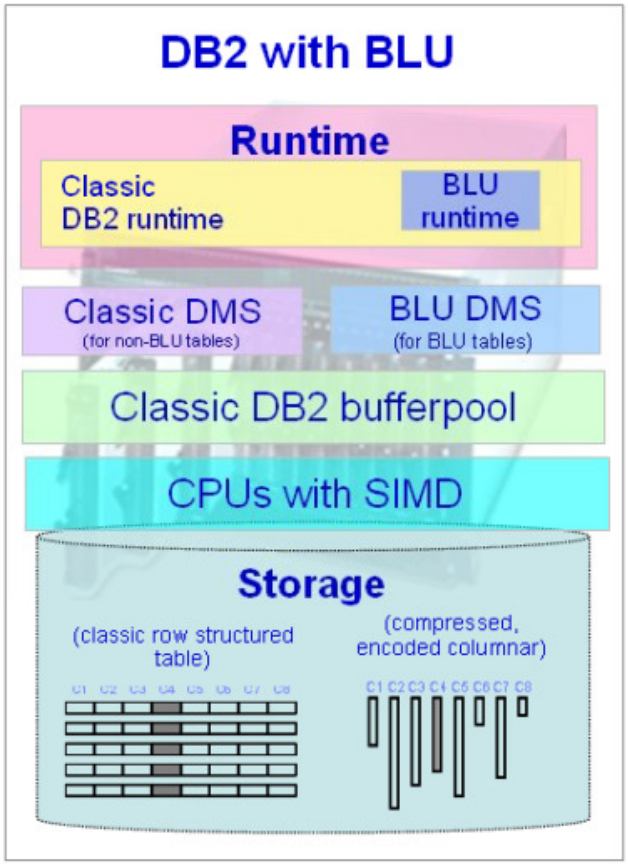
```
SELECT COUNT(*) from MYTABLE
where YEAR = '2010'
```



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

- **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



Creating a column-organized table

- Example:

```
CREATE TABLE sales_col (
  c1 INTEGER NOT NULL,
  c2 INTEGER,
   ...
  PRIMARY KEY (c1) ) ORGANIZE BY COLUMN;
```

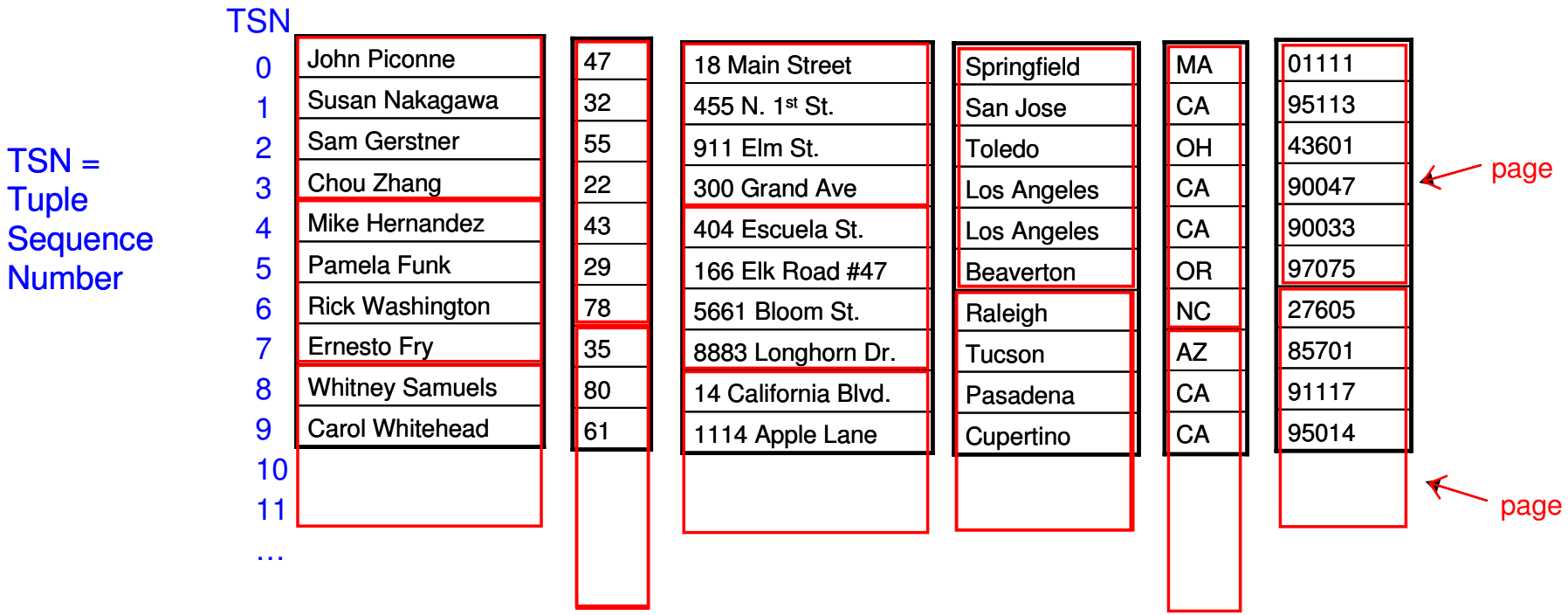
Columnar tables are always compressed by default.

- If dft_table_org = COLUMN (e.g. DB2_WORKLOAD= ANALYTICS):
  - `ORGANIZE BY COLUMN` is the default and can be omitted
  - Use `ORGANIZE BY ROW` to create row-organized tables

Data Layout

Columnar storage in DB2 (conceptual)

- Separate set of extents and pages for each column



Reclaiming Space in the Table

- **Objective:** Find empty storage extents and return pages to table space for re-use
- **Option 1:** If DB2_WORKLOAD=ANALYTICS, automatic space reclamation is active for all column-organized tables
-

```
update db cfg using auto_maint ON auto_tbl_maint ON auto_reorg ON;
```

- **Option 3**: Use REORG TABLE explicitly
  - Can use RECLAIMABLE_SPACE from ADMINTABINFO/ADMIN_GET_TAB_INFO to

```
                                                .-ALLOW WRITE ACCESS--.
>>-REORG-TABLE--table-name--RECLAIM EXTENTS--+---------------------+------><
                                             +--ALLOW READ ACCESS--+
                                             '--ALLOW NO ACCESS----'
```

What you see in the DB2 catalog: TABLEORG

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

```
SELECT tabname, tableorg, compression
FROM   syscat.tables
WHERE  tabname 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.

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, tabname, 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

SYN13033016521627515 2_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	...
0	2005-03-01	176	...
	2005-03-02	85	...
	2005-03-02	267	
	2005-03-04	231	
	...		
1023	...		
1024	...		
2047	...		

- Enables DB2 to skip portions of columns when scanning data during query
- Predicate WHERE S_DATE = 2007-01-01 would skip first range
- Predicate WHERE S_DATE = 2006-09-12 would scan both ranges

What you see in the DB2 catalog: Page Map Index

- **Automatically created and maintained**
- **Used internally to locate column data in the storage object**
- **Maps columns and TSNs to pages**

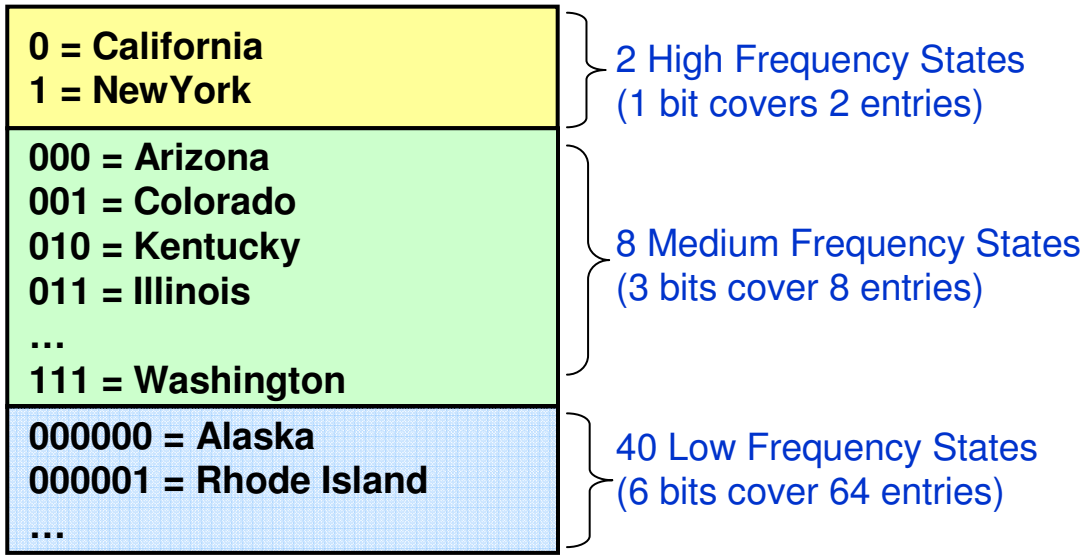
```
SELECT indschema, indname, colnames, indextype
FROM syscat.indexes
WHERE tabname = 'SALES_COL';

INDSCHEMA  INDNAME              COLNAMES             INDEXTYPE
---------- -------------------- -------------------- ---------
SYSIBM     SQL130330165215840   +ID                  REG
SYSIBM     SQL130330165216790   +COLGID+STARTTSN     CPMA

  2 record(s) selected.
```

Compression

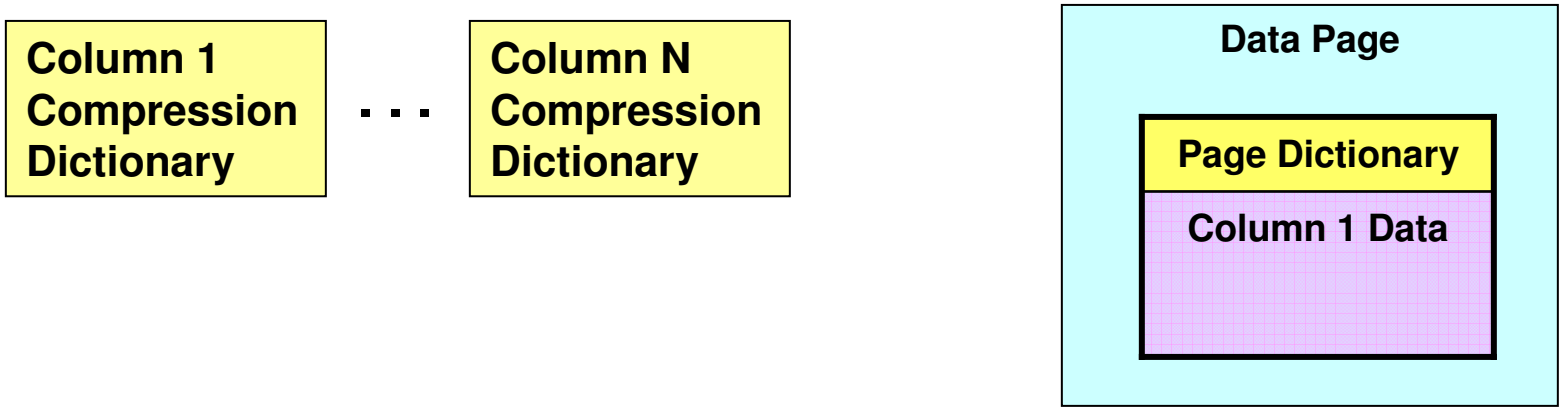
BLU uses Multiple Compression Techniques

- Approximate Huffman-Encoding ("frequency-based compression"), prefix compression, and offset compression
- Frequency-based compression: Most common values use fewest bits

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



Compression Dictionaries for Column-Organized Tables



- Column-level dictionaries: Always one per column
  - Dictionary populated during load replace, load insert into empty table
  - Automatic Dictionary Creation during Insert
- Page-level dictionaries: May also be created
  - If space savings outweighs cost of storing page-level dictionaries
  - Exploit local data clustering at page level to further compress data

Actionable Compression

- **Evaluating SQL predicates directly on compressed data**
  - No decompression required for comparisons like BETWEEN, < , >, <>, =
  - Many values can be compared with few instructions (SIMD processing)

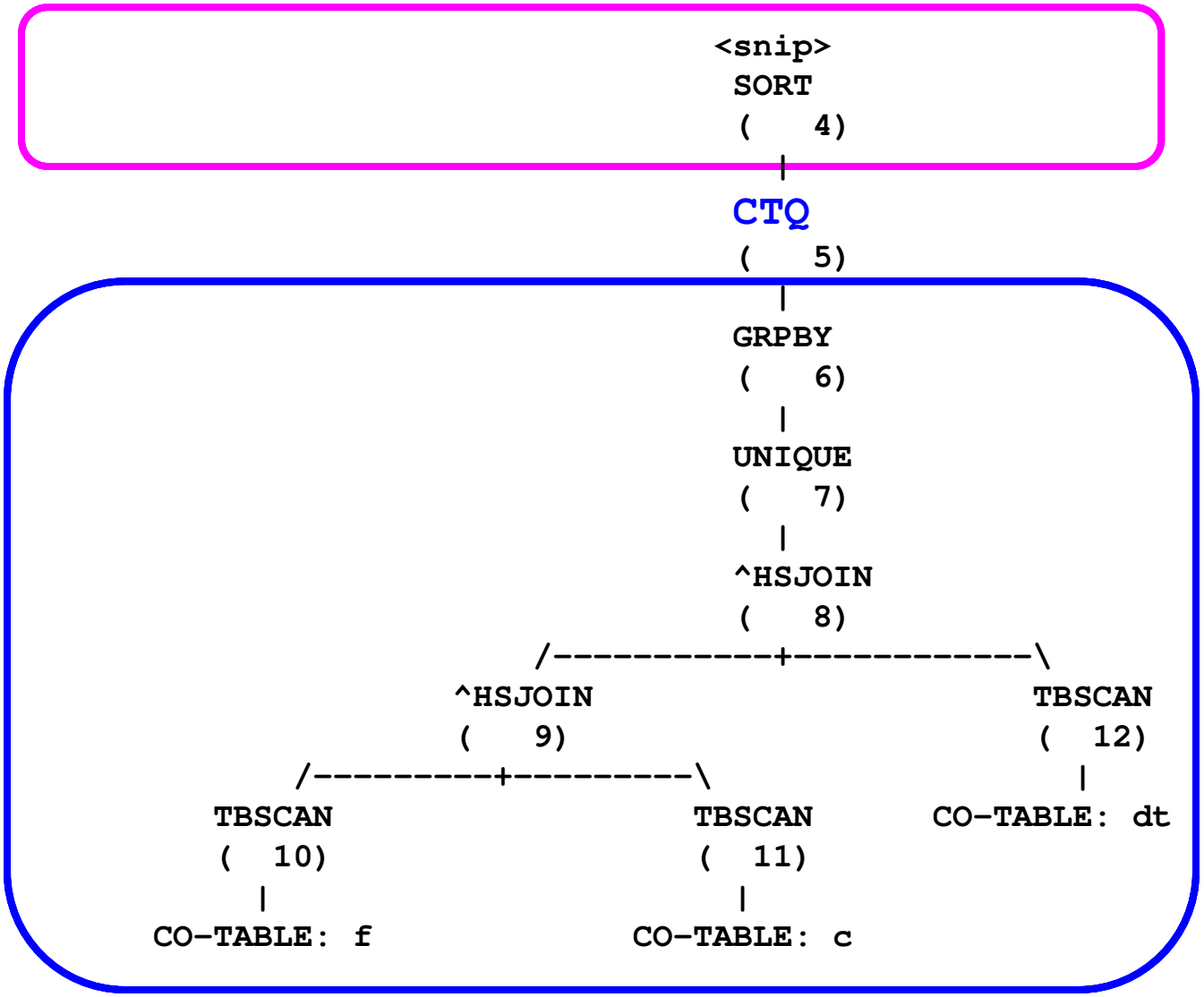
Query Processing

Sample Query

```
SELECT c.trading_name
FROM f, c, dt
WHERE f.client_dim_key = c.client_dim_key
  AND f.trade_dt = dt.dt_dim_key
  AND f.is_cancelled = 0
GROUP BY c.trading_name, dt.year
ORDER BY c.trading_name
```

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.)

Summary

- **What does BLU provide ?**
  - Columnar engine integrated into a traditional database providing excellent performance for analytics workload
- **What are key differentiators ?**
  - Actionable compression
  - Not bound to memory limits, but memory optimized
  - Well integrated into traditional database, which still can be used for high performant OLTP processing.
- **What‘s new ?**
  - SAP has announced support for DB2 BLU Acceleration