Information Management

IBM

Resource Optimization

Compression
Space Reclamation
Scan Sharing

Tim Vincent
DB2 LUW Chief Architect

June 18, 2010

Lower Storage Costs with Deep Compression

"With DB2 9, we're seeing compression rates up to 83% on the Data Warehouse. The projected cost savings are more than $2 million initially with ongoing savings of $500,000 a year." - Michael Henson

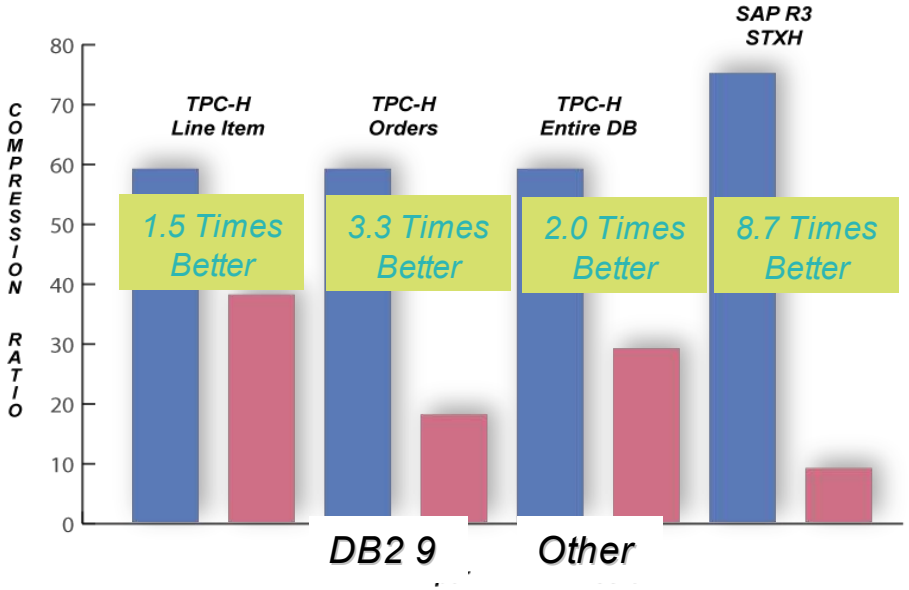
SunTrust℠

"We are saving anywhere between 60 to 65% in storage and we've actually found the performance has improved"

- Bashir Khan

DOWJONES

- **Best in industry**
- **Minimize storage costs**
- **Improve performance**
- **Easy to implement**
- **Advances with**
  - Index compression
  - Temp space compression
  - XML compression

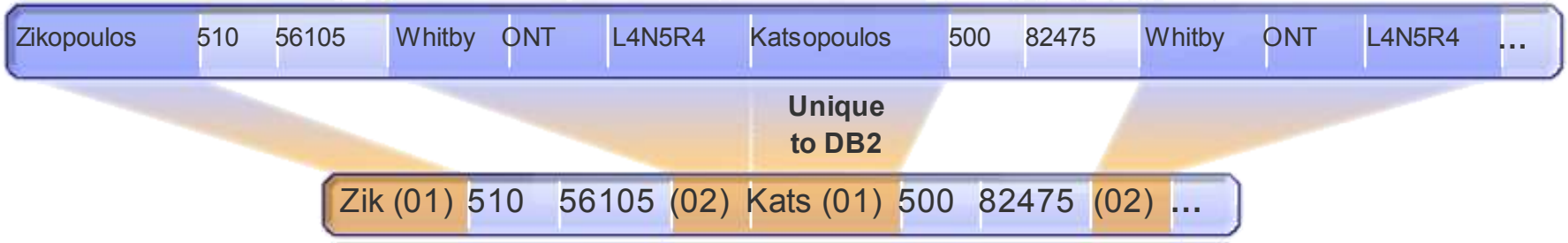


How Compression Works

- **Compression looks for repeating patterns across the entire table**
  - When a pattern is found, string is replaced with 12-bit symbol
  - Symbols are stored in a dictionary for fast lookup
- **Data resides compressed on pages (both on-disk and in memory)**
  - Significant I/O bandwidth savings – better performance
  - Significant memory savings – more efficient memory utilization

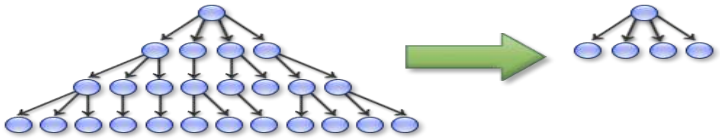
Name	Dept	Salary	City	Province	Postal_Code
Zikopoulos	510	56105	Whitby	ONT	L4N5R4
Katsopoulos	500	82475	Whitby	ONT	L4N5R4

Dictionary	
01	opoulos
02	WhitbyONTL4N5R4
…	…



Compression Improvements

- Multiple algorithms for automatic index compression



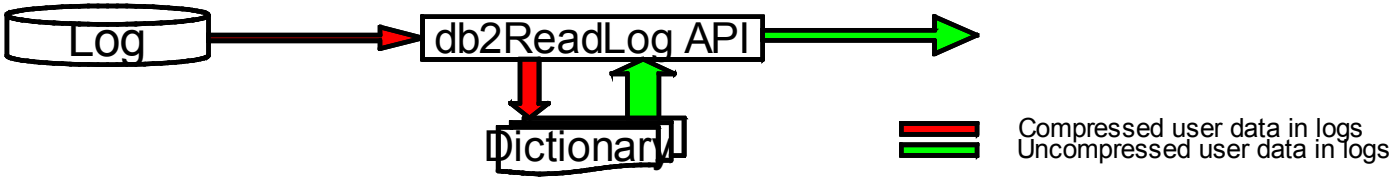
Unique in the industry

- Automatic compression for temporary tables

Unique in the industry

- Compression of large objects and XML

- Replication of Compressed Tables



Index Compression

- Algorithms implemented by the Database Engine (under-the-covers):
  - RID List Compression, Prefix Compression, and variable slot directory
  - Applies to all indexes except: Catalog indexes, MDC block indexes, XML path indexes and meta indexes, Index specifications
- Activated:
  - When row compression is activated on a table
  - CREATE INDEX with the new “COMPRESS YES” option
  - ALTER INDEX COMPRESS [YES|NO] statement, followed by an index reorg
- Savings
  - ADMIN_GET_INDEX_COMPRESS_INFO to estimate compression savings for uncompressed index
  - COMPRESS and PCTPAGESSAVED in the SYSINDEXES catalog
    - show if an index is defined as compressed and the percentage saved respectively

Index Compression Early Customer Results

Savings

DB2 9.7 Early Customer	Database Size	Data Compression Ratio	Index Compression Ratio	Total Database Saving
World leading construction machinery manufacturer, USA	725GB	72%	49%	68%
Global consumer and commercial product marketer, USA	1.4TB	58%	49%	56%
Haier Group, China	--	--	52%	--
John Deere, China	--	--	58%	--
Energy delivery company, USA	62GB	--	52%	--
Insurance company, Germany	176GB	--	50%	--
T-Systems, Germany	500GB	60%	73%	65%
Medtronic, USA	3.6TB	--	65%	--

ERP Systems

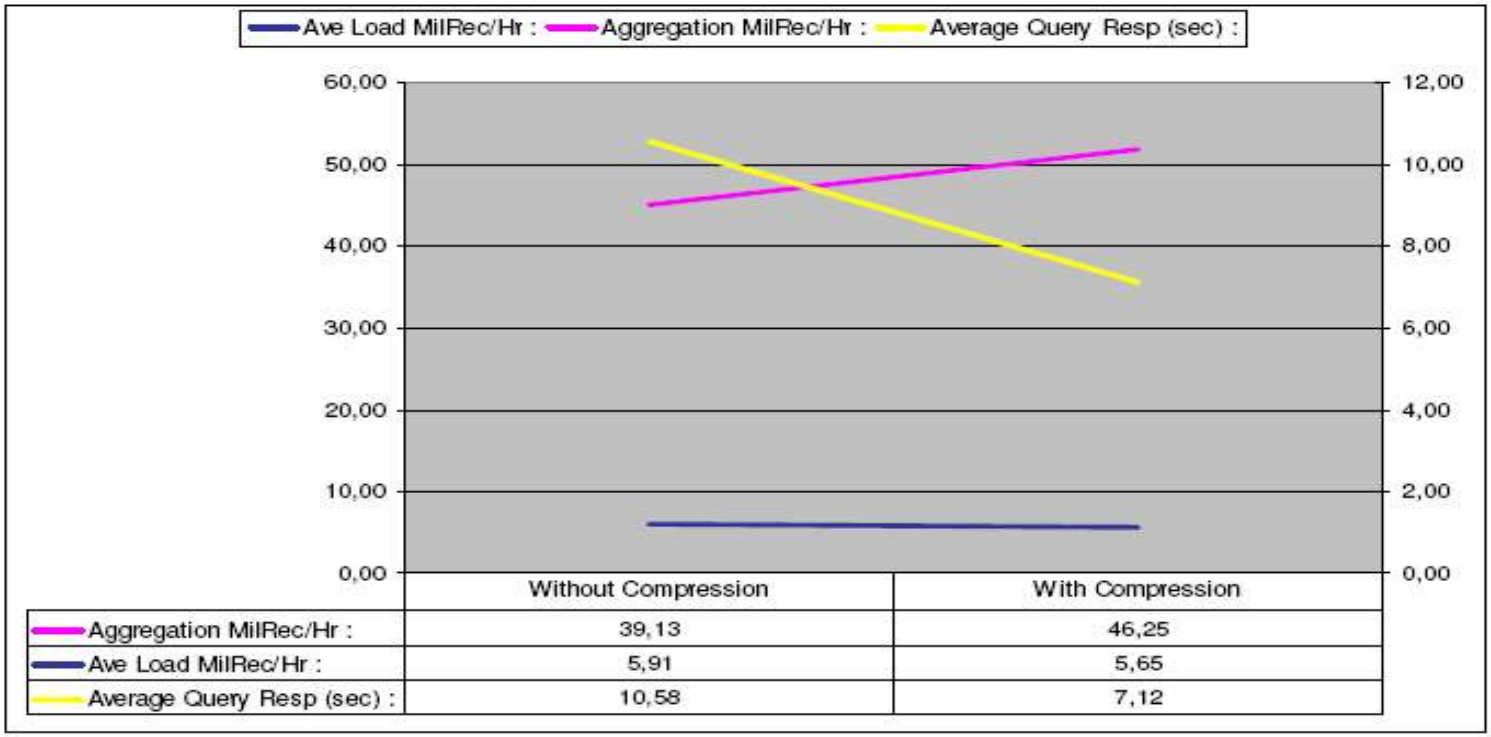
BW Systems

Temp Table Compression

- Compression of temporary tables aims to:
  - Reduce the amount of temporary disk space required
  - Have no performance penalty as a result of the extra processing required for row compression.
- Applicable to User temporary tables and System temps (DGTT/CGTT)
- Sorts, MGJN,NLJN, utilities, …
- If Deep Compression is licensed, then temporary tables will be compressed by default.
  - There is no additional action required by the user in order to use it. DB2 will evaluate the query and apply compression where appropriate.
- db2pd will report on temp tablespace usage

Deep Compression: Warehouse Results

- Customer POC
  - Compressed data from 15.3 to 7.9 TB
  - Table compression rates were between 80-85%
  - Aggregate Build throughput improved 15%
  - Query Response time decreased 23%



	Without Compression	With Compression
Aggregation MilRec/Hr :	39,13	46,25
Ave Load MilRec/Hr :	5,91	5,65
Average Query Resp (sec) :	10,58	7,12

Deep Compression: Warehouse Results

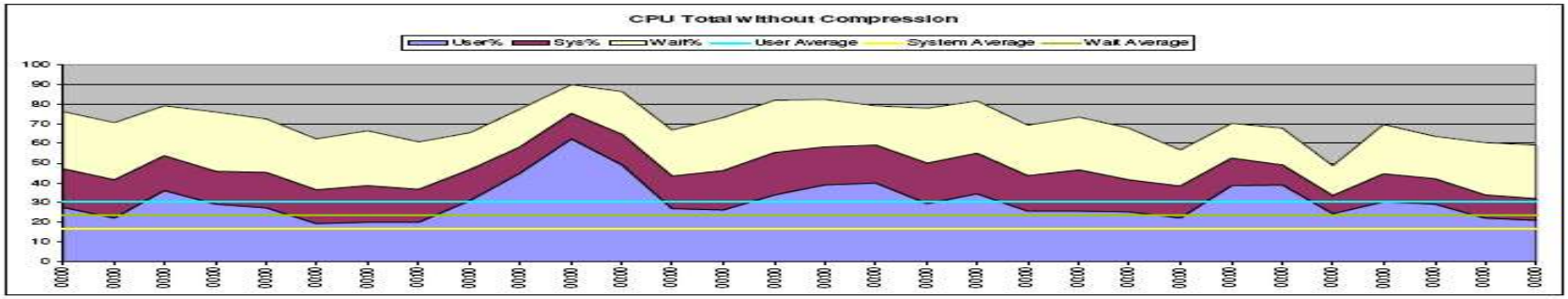


Figure 17 CPU Usage without compression

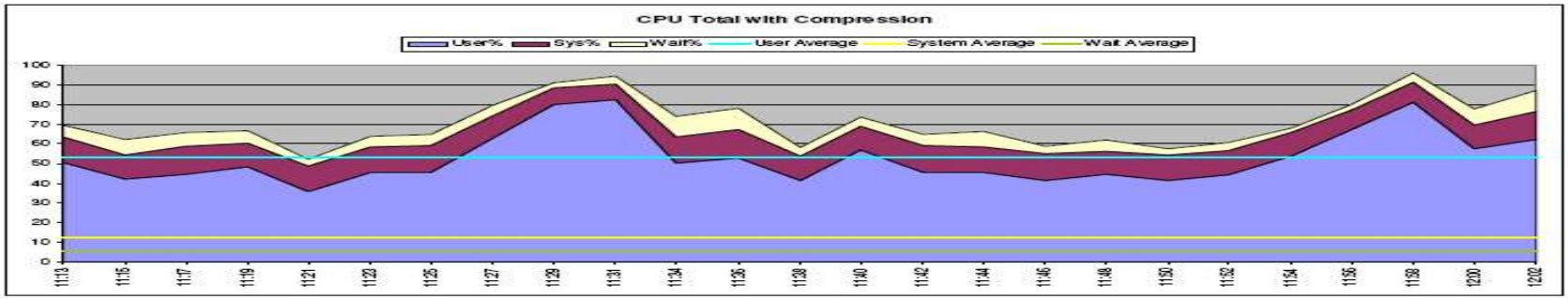
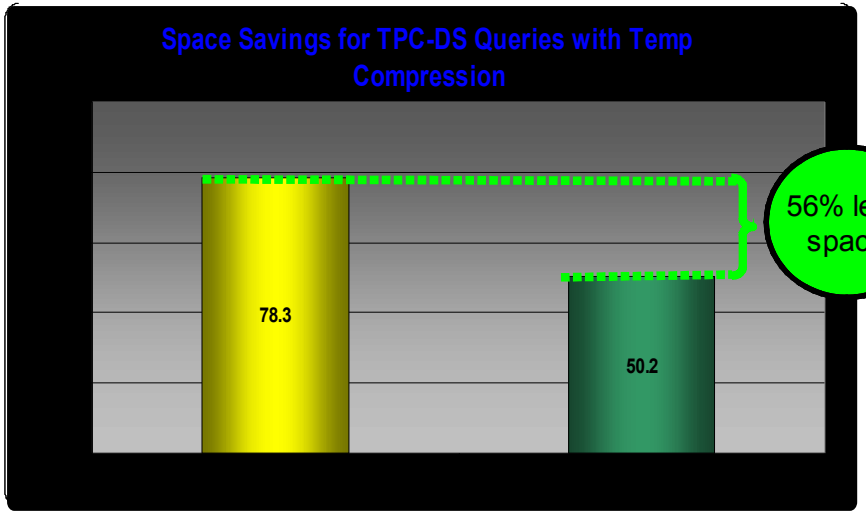
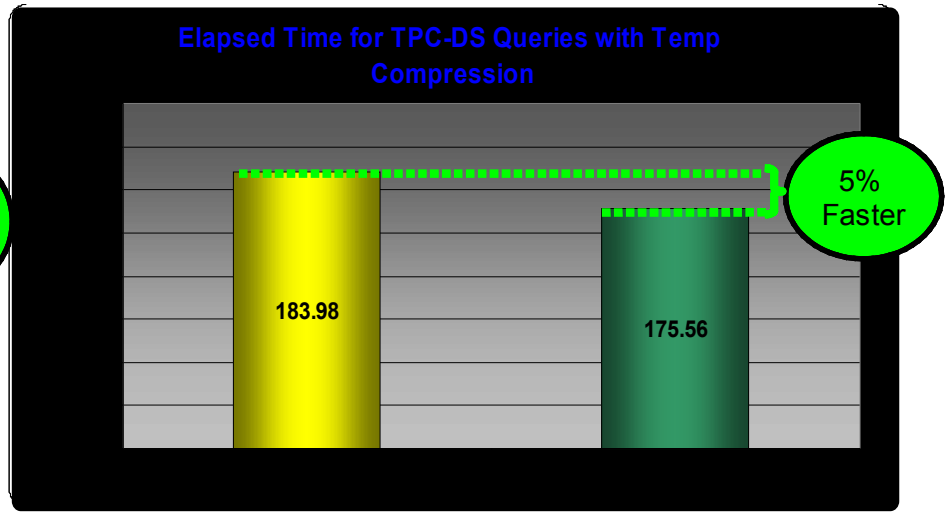


Figure 18 CPU Usage with compression

- **System CPU decreased from 16.5% to 12.3%**
- **Wait time decreased from 23.9% to 5.7%**
- **User CPU increased from 30.7% to 53% BUT combination of:**
  - Increased throughput on aggregate build (15%)
  - Decreased response time (23%)
  - Reduction in wait time
  - Compress/uncompress

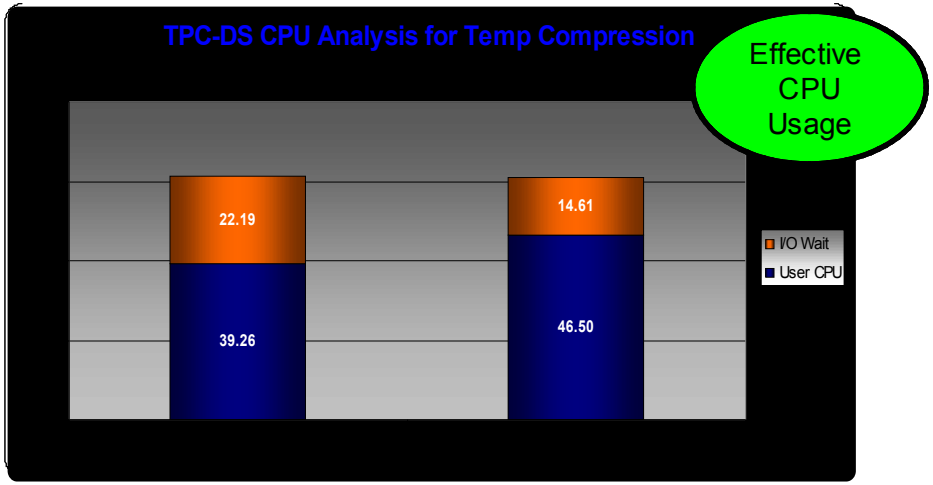
Temp Compression: Measurements





** Lower is better*

** Lower is better*



Simple Space Reclamation

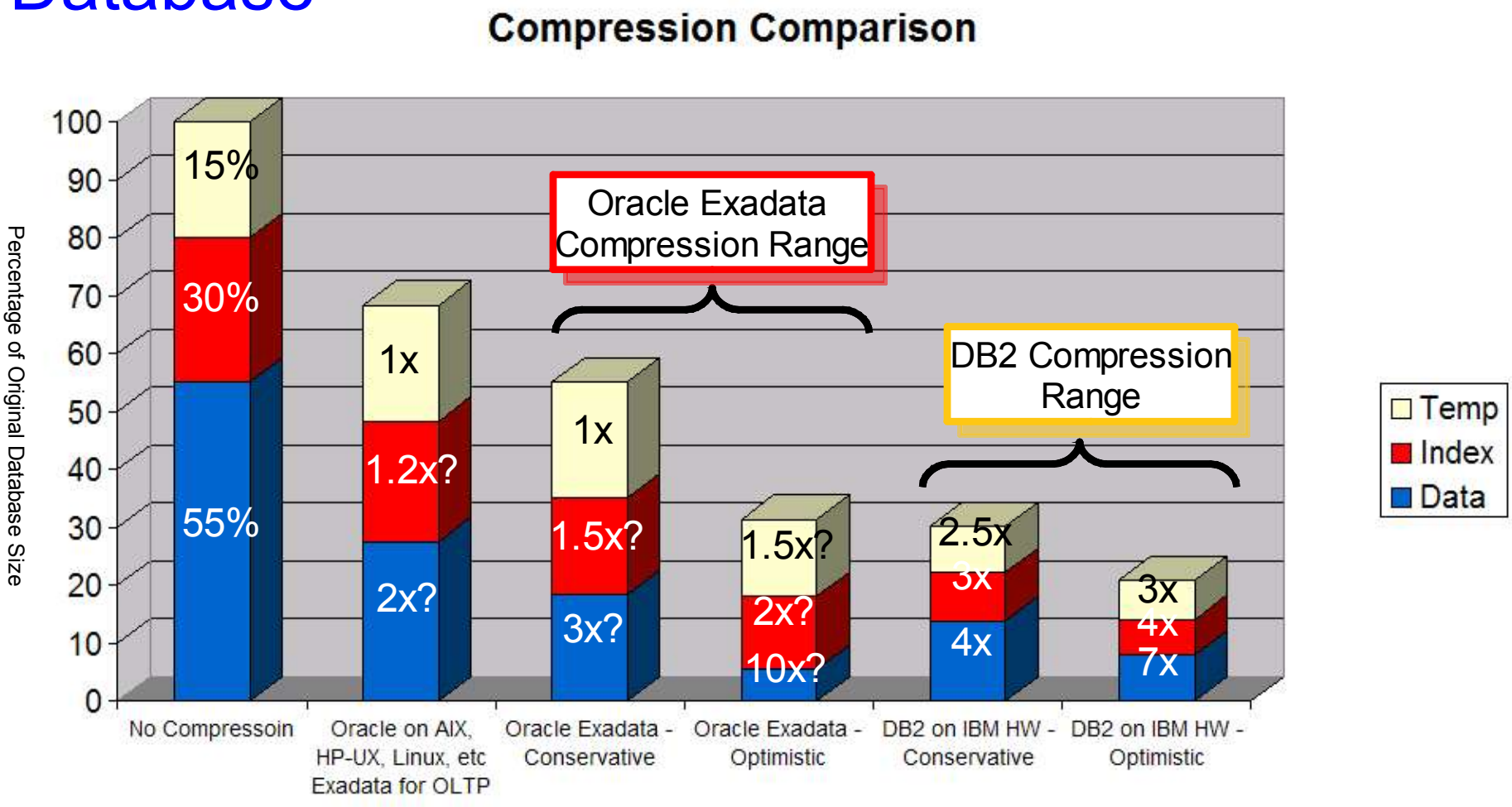
- New tablespace format to allow automated extent remapping
- Allow extents that are not assigned to any object (eg. table, index) to be used by other tablespaces

ALTER TABLESPACE REDUCE … XXX | MAX

- All new tablespaces will have this format
- Storage in an MDC table is tracked through a 'block map'
  - which extents have data and which don't
  - When a block is emptied the storage remains with the table and is available for later reuse by that table
- New option on reorg table command to not reorg the table but reclaim these empty blocks/extents

REORG TABLE <mdc table> RECLAIM EXTENTS ON [table partition clause]
ALLOW WRITE ACCESS | ALLOW READ ACCESS | ALLOW NOACCESS

DB2 Compresses all Aspects of the Database



Automatic Storage Migration

- Support ALTER DATABASE command for non-auto AS database
- Allow existing tablespaces to grow into auto storage containers

  ALTER TABLESPACE <table_space_name>
  MANAGED BY AUTOMATIC STORAGE

  Existing containers can no longer be altered.
- Support redirected tablespace restore to AS tablespace

  RESTORE DB <dbname> REDIRECT SET TABLESPACE CONTAINERS FOR <tablespaceID> USING AUTOMATIC STORAGE
- REBALANCE support after a new path is added to the database
  - Allows existing tablespaces to use new path
- Ability to DROP a path from an automatic storage database.
  - Can be used to migrate to new containers

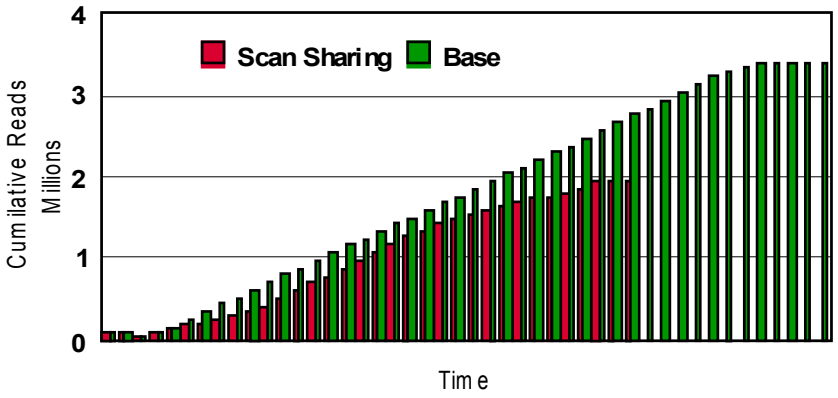
Scan Sharing Performance Test

- TPCH Q1 : CPU Intensive, Slow Query On Lineitem Table Using A Table Scan
- TPCH Q6 : IO Intensive, Fast Query On Lineitem Table Using A Table Scan

Test Scenario : Queries executed in parallel in the following sequence

- Results : 34% Improvement In End to End Timing

Reads on a disk: 42% Reduction



CPU Usage