DB2 10 Performance Topics
Session Number 2763

Cristian Molaro , MConsulting
IBM GOLD Consultant
IBM Champion

Special thanks to Jeff Berger, IBM SVL DB2 Performance

IBM Software
Information On Demand 2011

Disclaimer

Please be aware that the actual programming techniques, algorithms and all numerical parameters used in examples given in this presentation are subject to change at some future date either by a new version of DB2, a new release, a Small Programming Enhancement (SPE) or a Programming Temporary Fix (PTF).

The information contained in this presentation has not been submitted to any formal review and is distributed on an "as is" basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

DB2 is a trademark of International Business Machine corporation. This presentation uses many terms that are trademarks. Wherever we are aware of trademarks the name has been spelled in capitals.

Agenda

- Introduction
- Performance migration path
- Catalog Changes
- Statement concentration and literal replacement
- Plan Management
- Virtual Storage
- Logging enhancements
- Insert performance
- Inline Lobs
- Additional non key columns in Unique indexes
- zIIP
- Summary

Information Management

IBM

Draft Document for Review

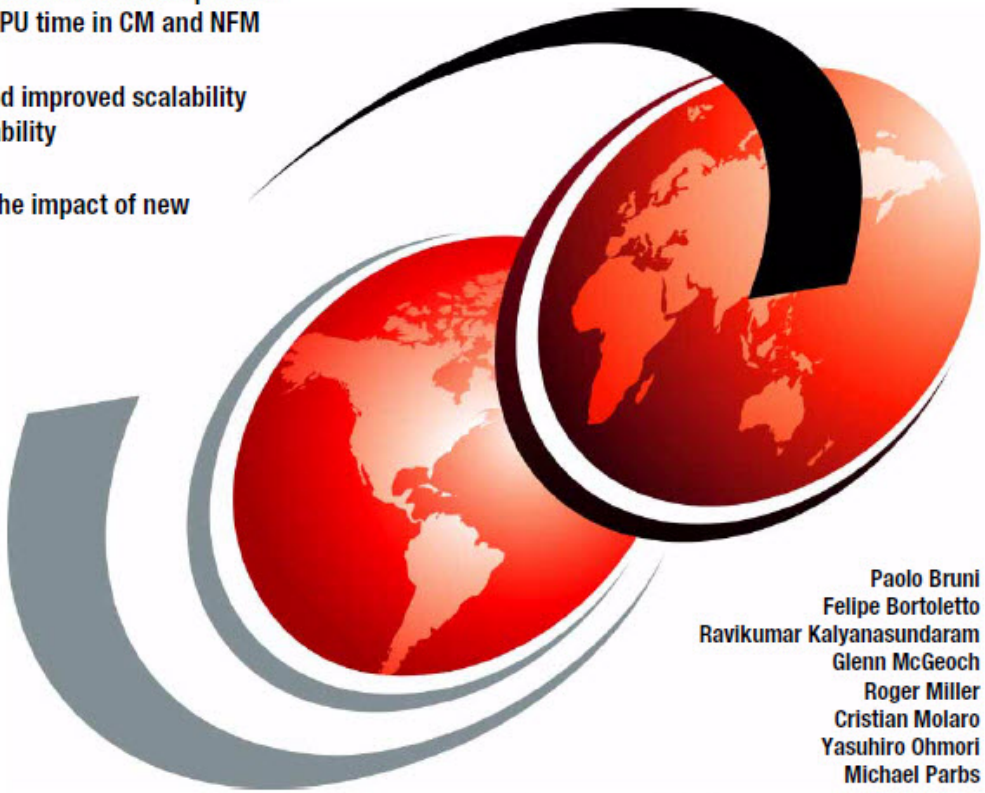
SG24-7942-00

DB2 10 for z/OS Performance Topics

Discover the functions that provide reduced CPU time in CM and NFM

Understand improved scalability and availability

Evaluate the impact of new functions



Paolo Bruni
Felipe Bortoletto
Ravikumar Kalyanasundaram
Glenn McGeoch
Roger Miller
Cristian Molaro
Yasuhiro Ohmori
Michael Parbs

Redbooks

ibm.com/redbooks

- IBM Silicon Valley Lab (image from the www)

- A lot about DB2 10 System Performance…

- Conversion Mode
  - Address space, memory changes to 64 bit, some need REBIND
  - DDF High Performance DBATs
  - DB2 9 utility enhancements in CM8
  - Index list prefetch
  - Buffer pool enhancements
- New Function Mode
  - Most utility enhancements
  - SQL Procedure Language performance improvements
  - Workfile spanned records, PBG
  - Query parallelism improvements
- Migrating soon?
  - Information APARs: II14477 (9), II14474 (8)
  - Toleration APARs: PK56922
  - DB2 9 & V8 pre-migration job APAR: PM04968

- A lot about DB2 10 Application Performance…

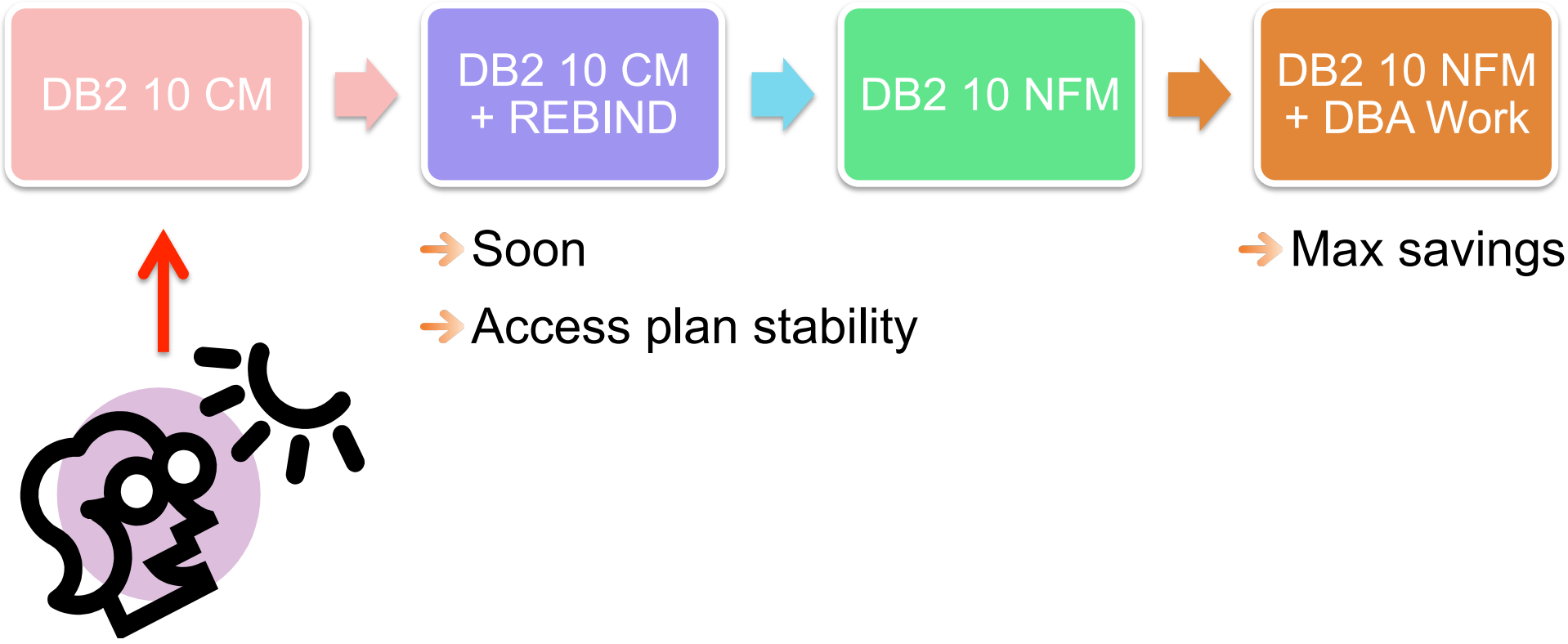
- Conversion Mode
  - SQL runtime improved efficiency
  - Faster single row retrievals via open / fetch / close chaining
  - Faster inserts
  - Parallel index update at insert
  - Workfile in-memory enhancements
  - Index list prefetch
  - Faster log I/O
  - More/better query parallelism

} Requires Rebind

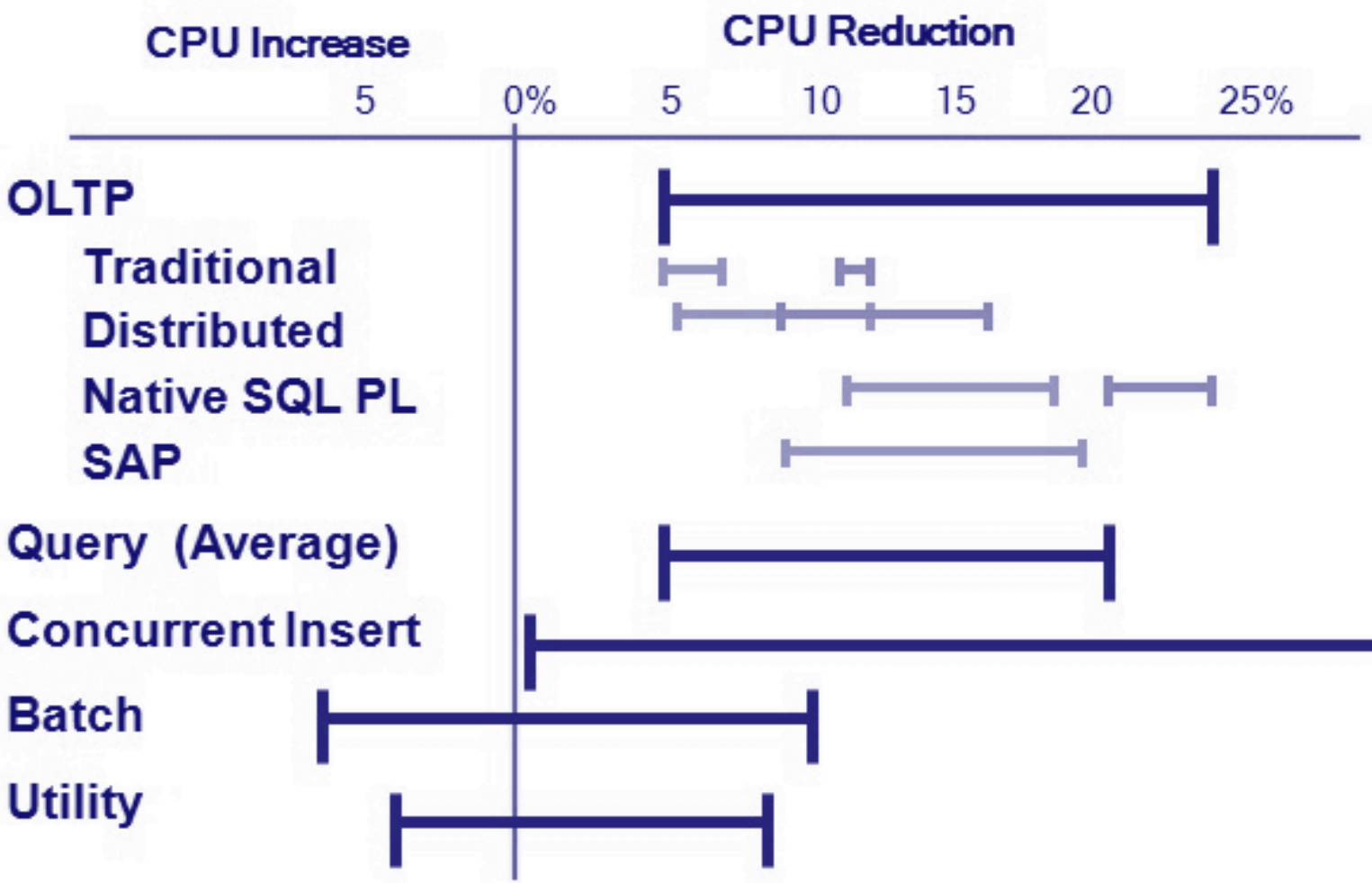
- New Function Mode
  - Efficient caching of dynamic SQL statements with literals
  - SQL Procedure Language performance improvements
  - Hash access path
  - Index include columns
  - Inline LOBs
  - MEMBER CLUSTER for UTS

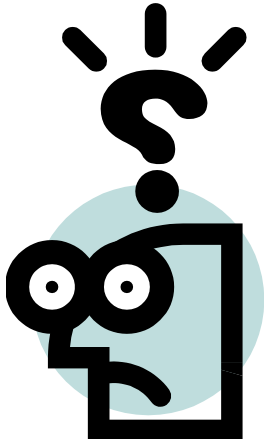
- DB2 10 *"Performance"* migration path

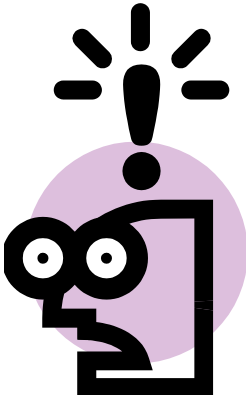
- The ideal scenario



- DB2 9 to DB2 10 Migration CPU changes







- DB2 10 Catalog changes

- Biggest catalog restructure since V1.2
- Online REORG enabled for all catalog/directory objects
- Catalog and directory is now SMS and Extended Addressability

```
 DEFINE CLUSTER                                  -
        ( NAME(DBAT.DSNDBC.DSNDB06.SYSTSFLD.I0001.A001) -
          DATACLAS(DBATDC)     -
          STORCLAS(DBATSC)     -
          LINEAR                                  -
          REUSE                                   -
          KILOBYTES(2200 2200)                      -
          SHAREOPTIONS(3 3) )                     -
      DATA                                        -
        ( NAME(DBAT.DSNDBD.DSNDB06.SYSTSFLD.I0001.A001) -
                                )
IGD17070I DATA SET DBAT.DSNDBC.DSNDB06.SYSTSFLD.I0001.A001
ALLOCATED SUCCESSFULLY WITH 1 STRIPE(S).
IGD17172I DATA SET DBAT.DSNDBC.DSNDB06.SYSTSFLD.I0001.A001
IS ELIGIBLE FOR EXTENDED ADDRESSABILITY
IDC0508I DATA ALLOCATION STATUS FOR VOLUME DBAT02 IS 0
IDC0181I STORAGECLASS USED IS DBATSC
IDC0181I DATACLASS USED IS DBATDC
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
```

- DB2 10 Catalog changes

- **Size**
  - Partition-by-growth (PBG) catalog / directory table spaces
    - V8/DB2 9 APAR PK80375 adds zparm for SPT01 compression
  - SPT01 compression in DB2 10 (no need for opaque zParm)
  - New LOB columns to the catalog (not compressed)
    - PM27811 / PM27073: change the in-line length of DSNDB01.SPT01
    - Inline LOBs may be compressed

IMPORTANT: The restructuring and the presence of LOBs imply changes to catalog recovery procedures and new disaster recovery testing

- **Improved concurrency and reduced contention**
  - Change to row-level locking
  - Indexes instead of links
  - DSN1CHKR is no longer required
  - Reduces catalog contention dramatically
    - Concurrent DDL processing
    - BIND operations

- DB2 Catalog changes and Performance

- Indexes for the catalog instead of links improves concurrency and makes catalog health checking easier
- **BUT:** operations that access the DB2 catalog are expected to show a higher CPU utilization
  - Single thread BIND/REBIND shows degraded CPU and ET performance on CM
- BIND will take longer but has less contention
  - You may change existing procedures to go parallel
  - **BUT** cannot do this until post ENFM

IMPORTANT: Catalog indexes are introduced when you migrate to CM but there is no increased concurrency in this mode

- The dynamic SQL full PREPARE increase in class 2 CPU and elapsed time ranges from **20% to 30%** when comparing DB2 9 to DB2 10
  - Extra cost is largely attributed to the utilization of indexes instead of links in the DB2 catalog

- DB2 10 Literal replacement

- Dynamic SQL with literals can be re-used in the DSC
- Literals replaced with ***&***

EXAMPLE:

```
WHERE ACCOUNT_NUMBER = 123
```

Becomes:

```
WHERE ACCOUNT_NUMBER = &
```

- How to enable:
  - `CONCENTRATE STATEMENTS WITH LITERALS` in the `PREPARE ATTRIBUTES` clause
  - Or set `LITERALREPLACEMENT` in the ODBC initialization file
  - Or set the keyword `enableLiteralReplacement='YES'` in the JCC Driver

- DB2 10 Literal replacement

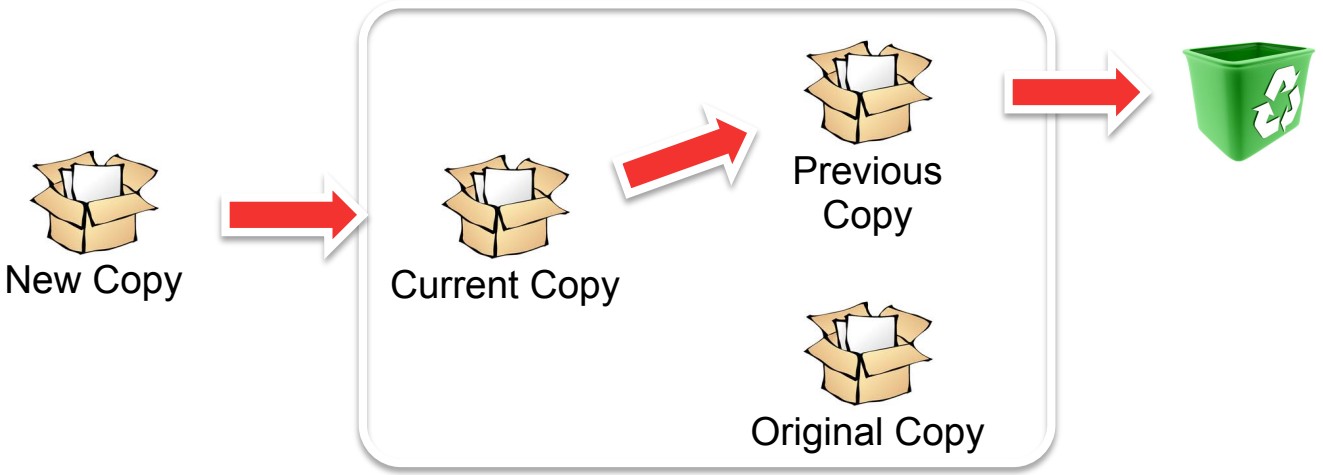
- Accounting report:

```
DYNAMIC SQL STMT                TOTAL
--------------------  --------
REOPTIMIZATION                     0
NOT FOUND IN CACHE                 4
FOUND IN CACHE                     1
IMPLICIT PREPARES                  0
PREPARES AVOIDED                   0
CACHE_LIMIT_EXCEEDED               0
PREP_STMT_PURGED                   0
CSWL - STMTS PARSED                5
CSWL - LITS REPLACED               3
CSWL - MATCHES FOUND               1
CSWL - DUPLS CRETED                0
```

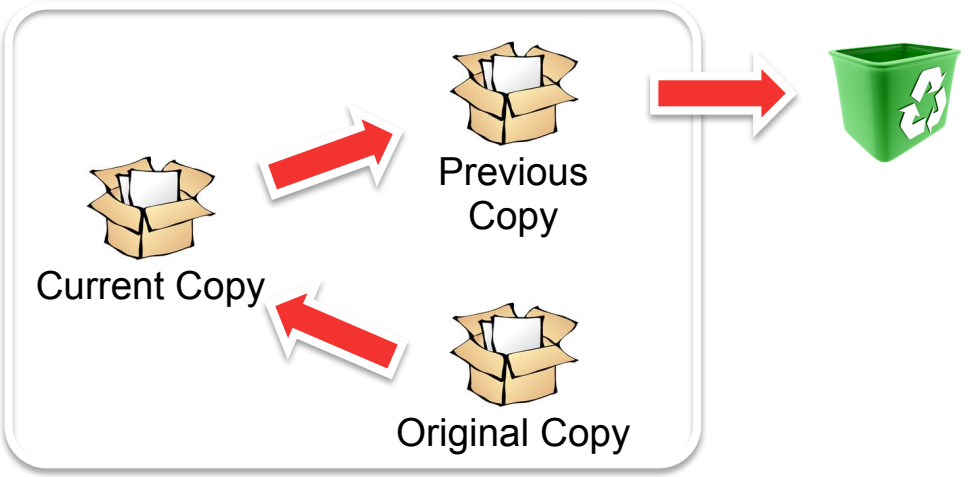
- Performance
  - Using parameter markers may still provide best performance
  - Biggest performance gain for repeated SQL with different literals

- Plan Management - EXTENDED

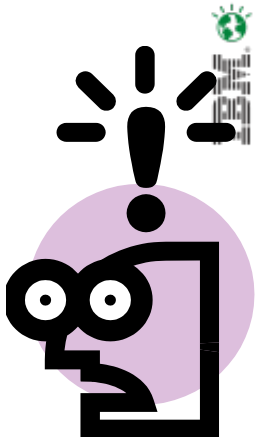
- REBIND … PLANMGMT(EXTENDED)



- REBIND … SWITCH(ORIGINAL)



- DB2 10 and Plan Management



- EXTENDED is now the DEFAULT in DB2 10
- Monitor DAS utilization by SPT01
- APAR PM09354 Support DBPROTOCOL change
  - FIX REBIND PACKAGE TO ALLOW CHANGING CERTAIN BIND OPTIONS WHEN PLANMGMT(BASIC|EXTENDED) IS SPECIFIED
- SYSIBM.SYSPACKCOPY
  - SYSPACKAGE like metadata for any previous or original package copies
  - No longer need to SWITCH to see information on inactive copies
- Watch out for PM25679: ACCESS PATH ENHANCEMENT
  - BIND option APREUSE to allow a package to reuse access paths
  - BIND option APCOMPARE to specify what actions to take if access paths change

Afraid of REBIND? Google “Escaping the REBIND blues in DB2 9 for z/OS”

- More DB2 10 Plan Management Enhancements

- EXPLAIN PACKAGE
  - Extract **in use** PLAN_TABLE information for packages
  - Useful if PLAN_TABLE information is missing
  - Useful if not BIND EXPLAIN(YES)

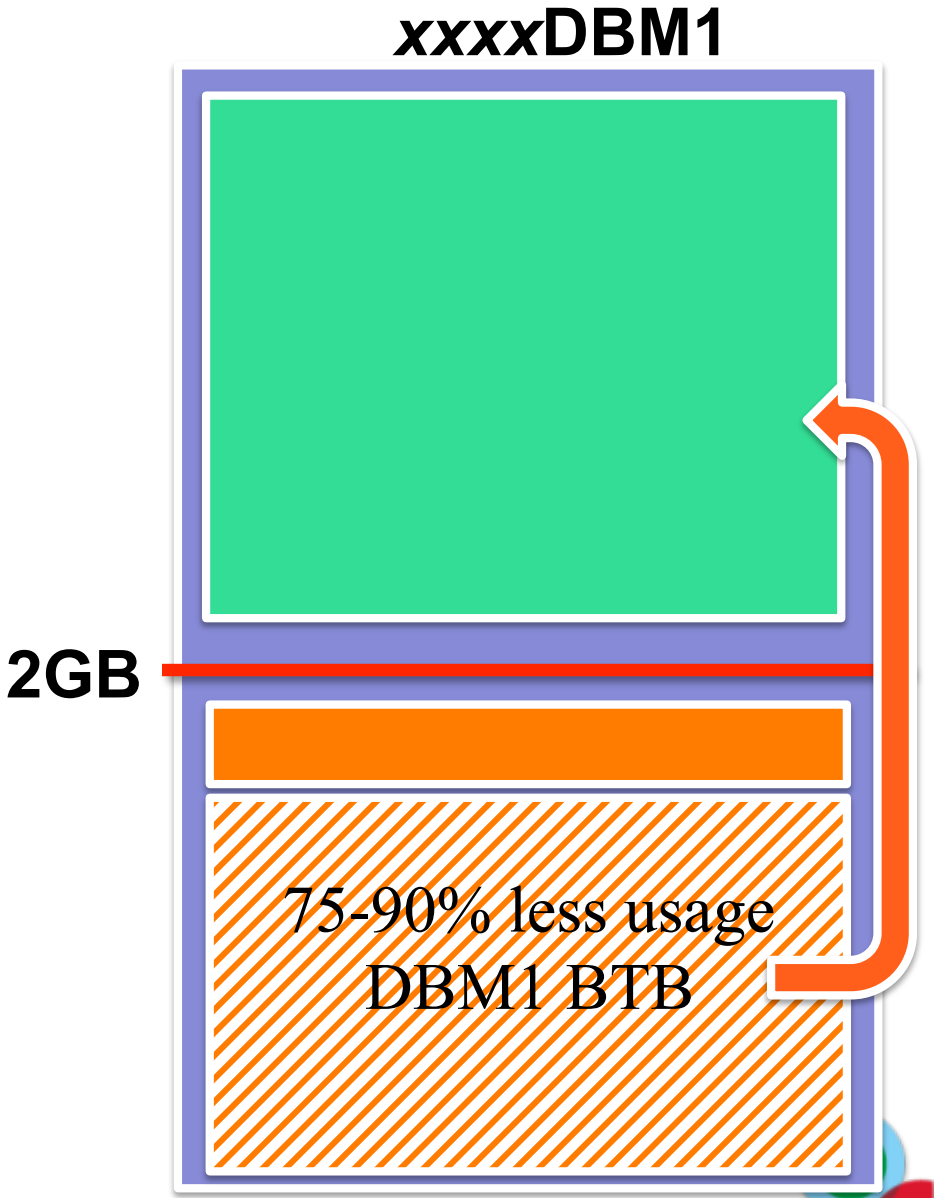
EXAMPLE:

```
>>-EXPLAIN----PACKAGE----------->
>>-----COLLECTION--collection-name--PACKAGE--package-name--------->
>----+--------------------------+----+------------------+-------->
     |                          |    |                  |
     +---VERSION-version-name---+    +---COPY--copy-id---+
```

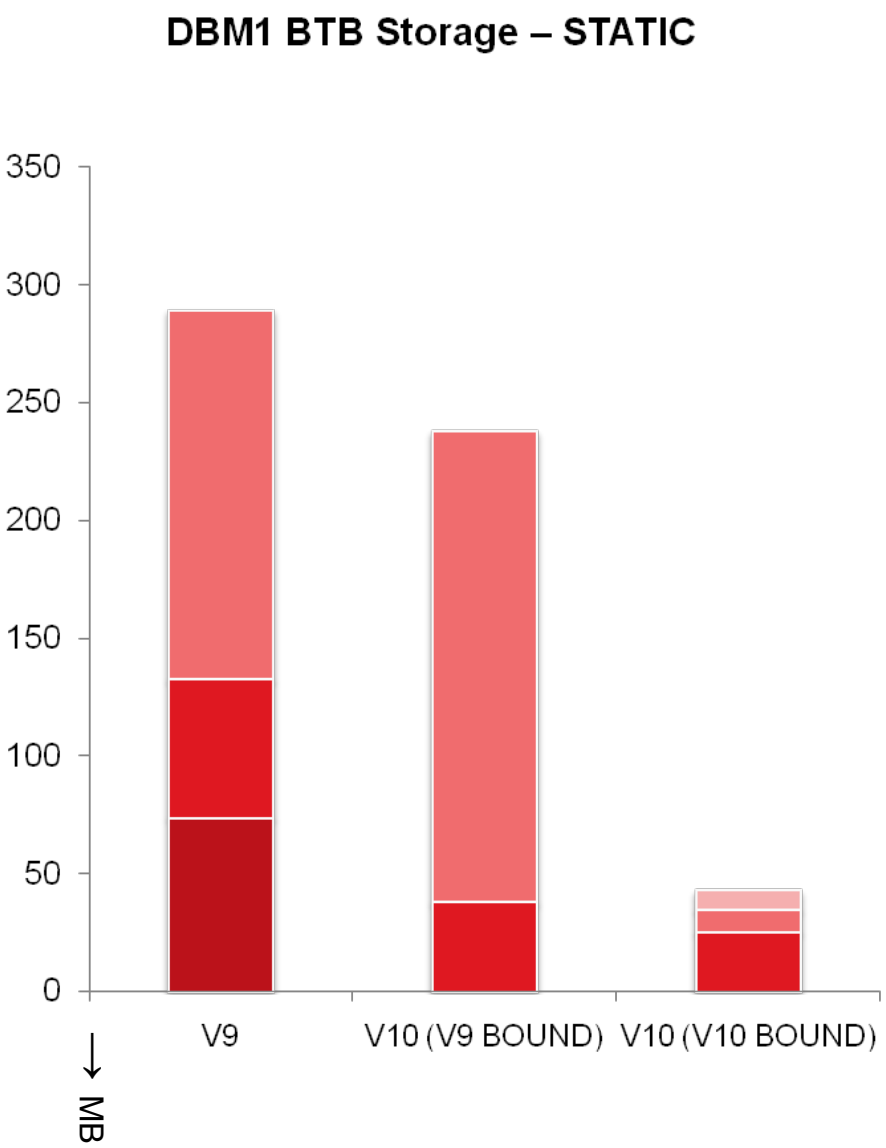
  - COPY-ID can be ‘CURRENT’, ‘PREVIOUS’, ‘ORIGINAL’
- Bind package EXPLAIN(ONLY)
  - Populates EXPLAIN tables but does not create a package
  - The existing package is not dropped or replaced, even if ACTION(REPLACE) is specified

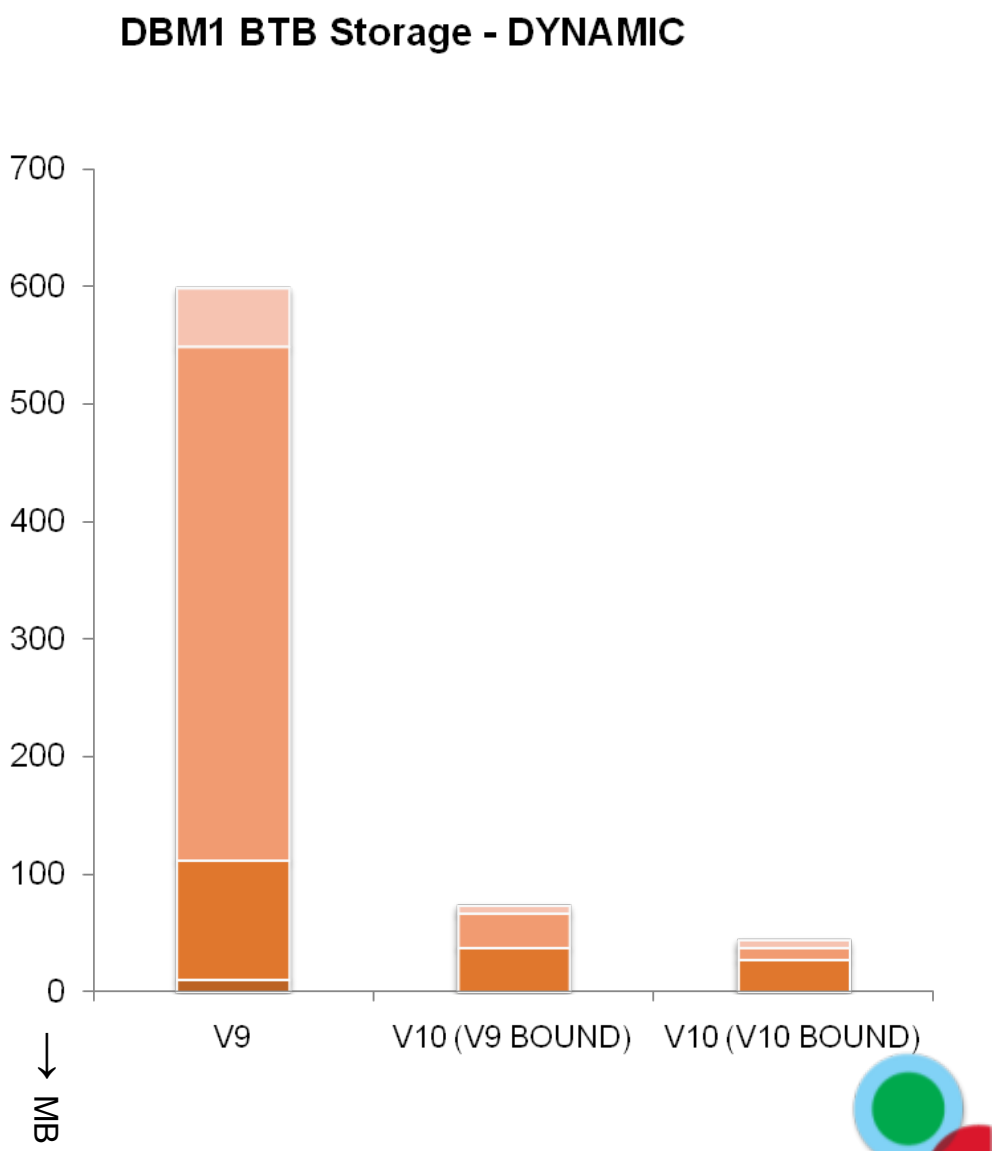
- DB2 10 and virtual storage

- DBM1 below 2GB
  - 75-90% less usage in DB2 10 compared to DB2 9
- Larger number of threads
  - Possible data sharing member consolidation
- Improve CPU with storage
  - More RELEASE(DEALLOCATE)
  - Larger MAXKEEPD values for KEEPDYNAMIC=YES
- Should I still monitor virtual storage?

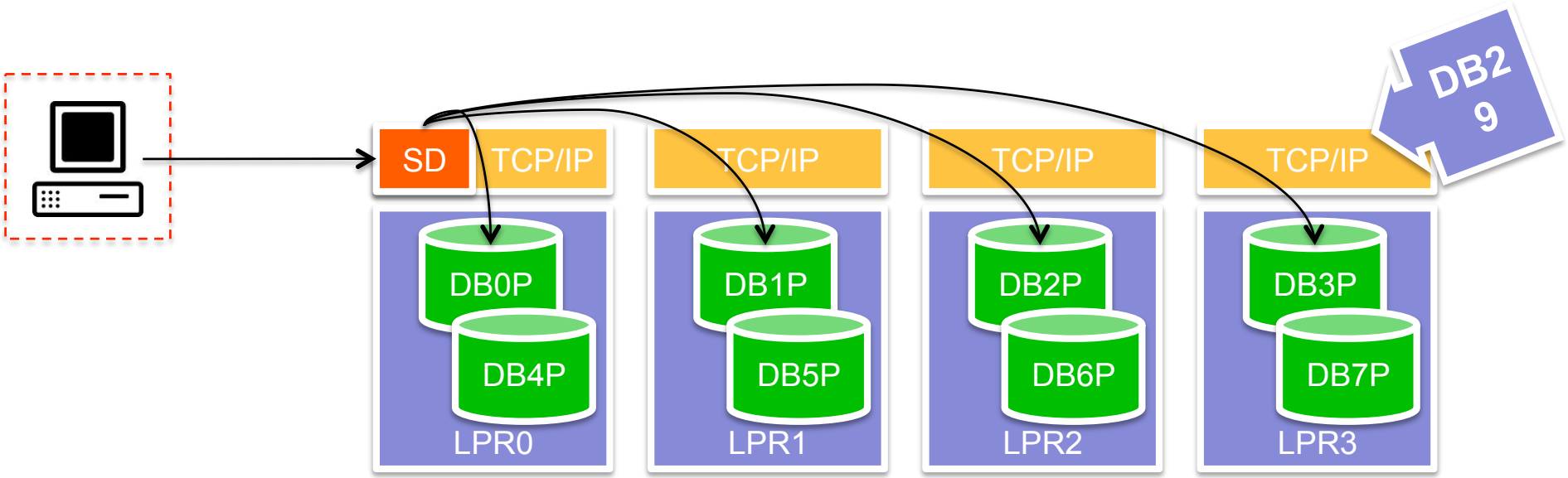


- Sample SQL workloads and DBM1 BTB

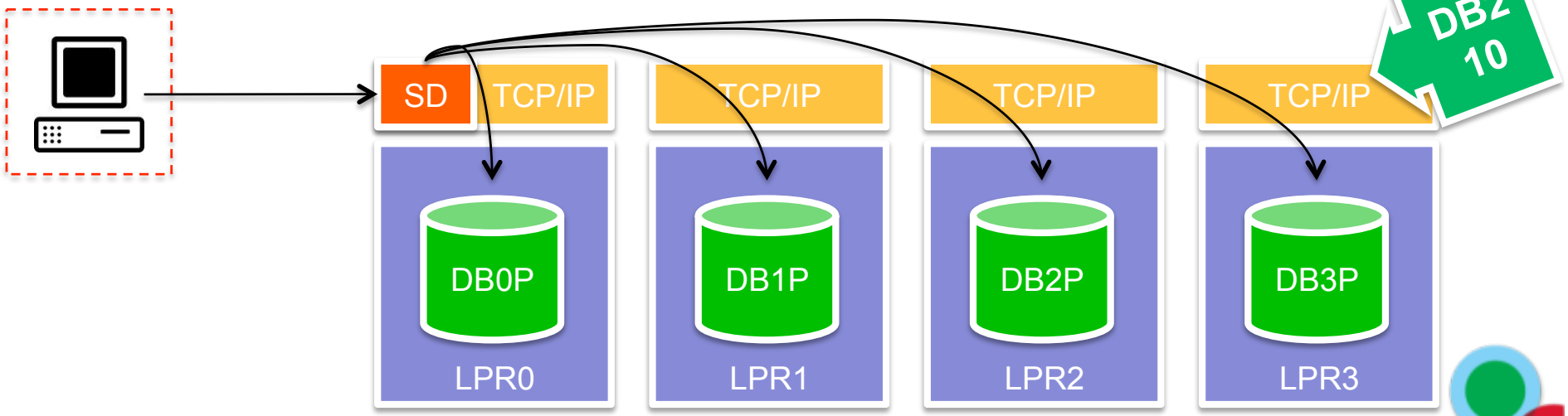




- DB2 Subsystem colocation → CONSOLIDATION

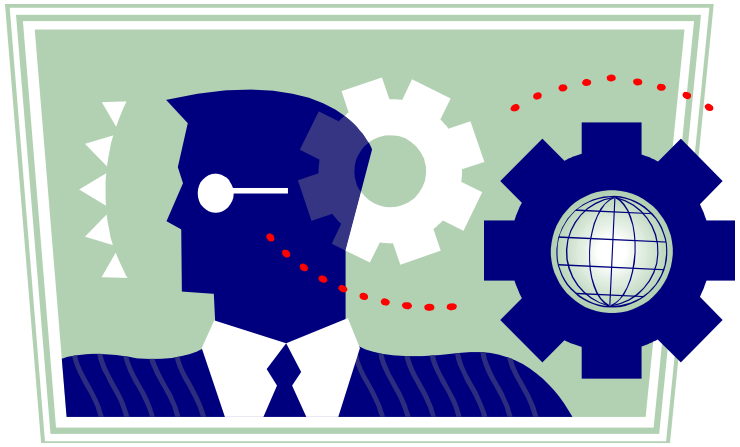


- 5 to10 times more concurrent users → up to 20,000



DB2 10 DS member consolidation

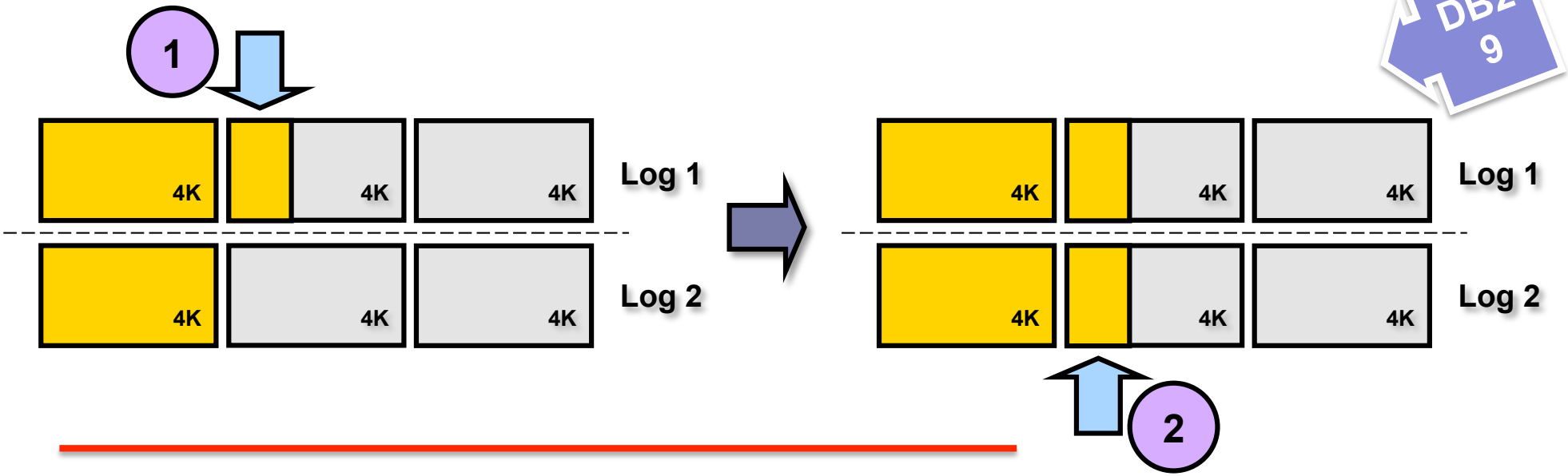
- Virtual storage can limit the number of concurrent threads for a single member or subsystem to a few hundred threads
- Virtual storage is the most common constraint



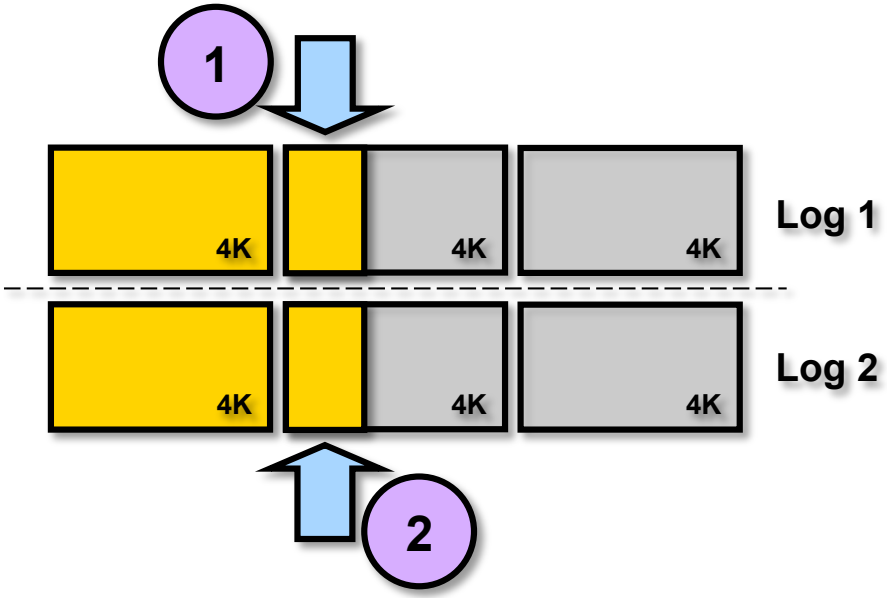
- 5 to10 times more concurrent users → up to 20,000
- Rule of thumb:
  - Save ½% CPU for each member reduced
  - More saving on real storage

DB2 10

- DB2 COMMIT & synchronous writes



DB2 9

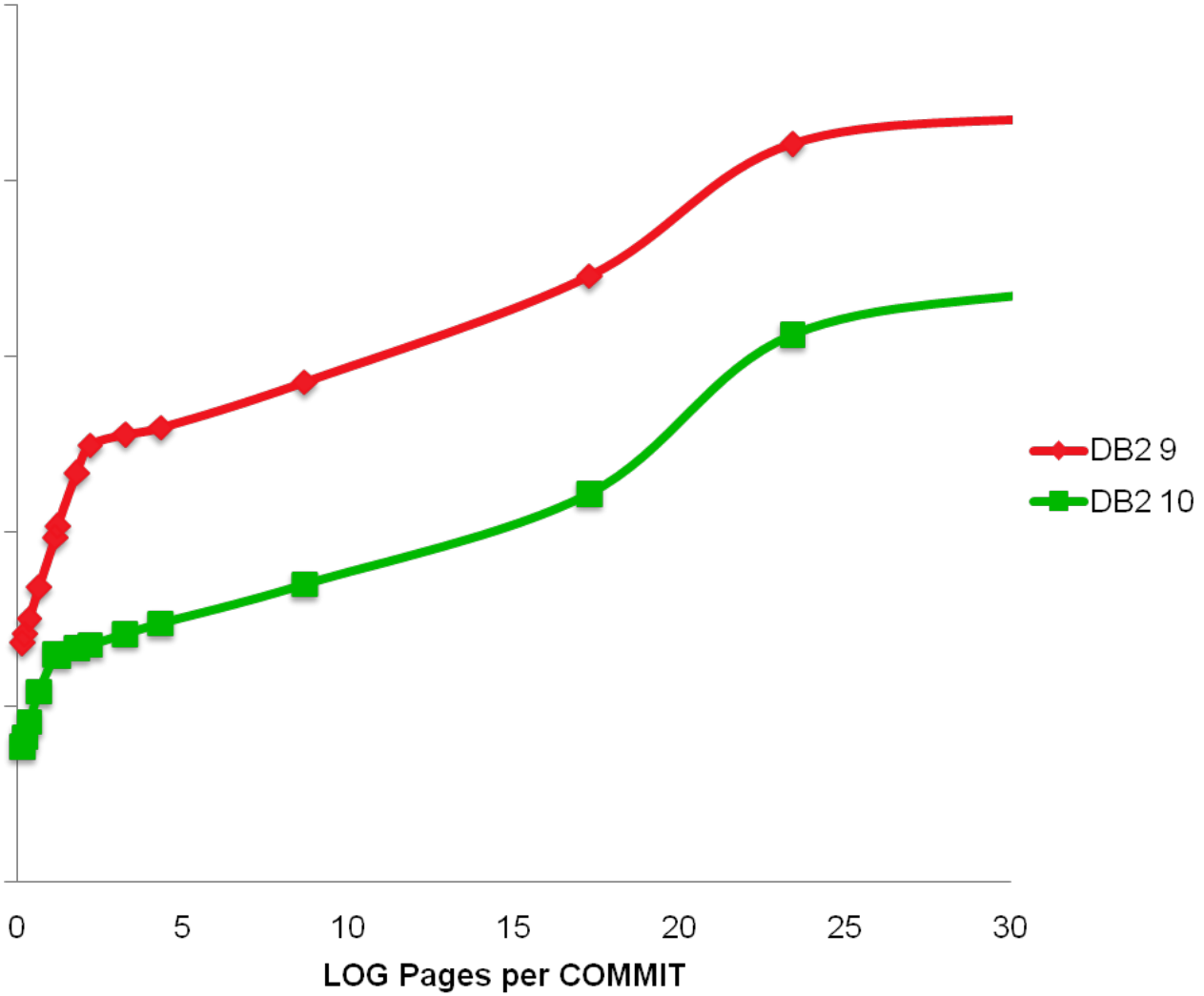


DB2 10

1 Write log page to Log 1

2 Write log page to Log 2

- DB2 10 Logging suspension time

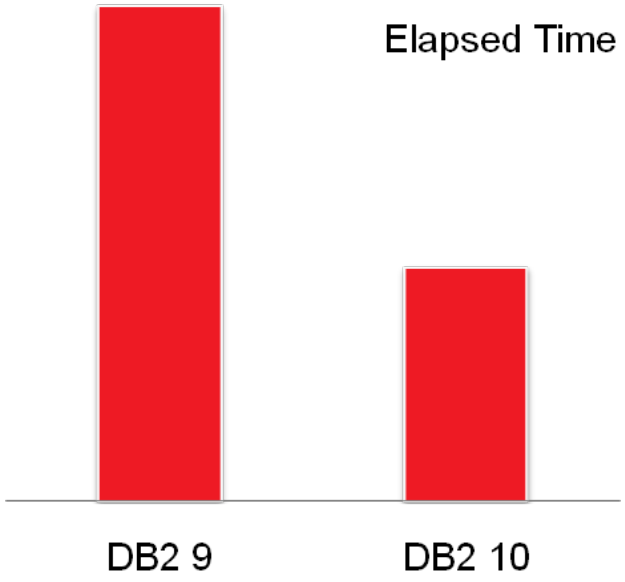


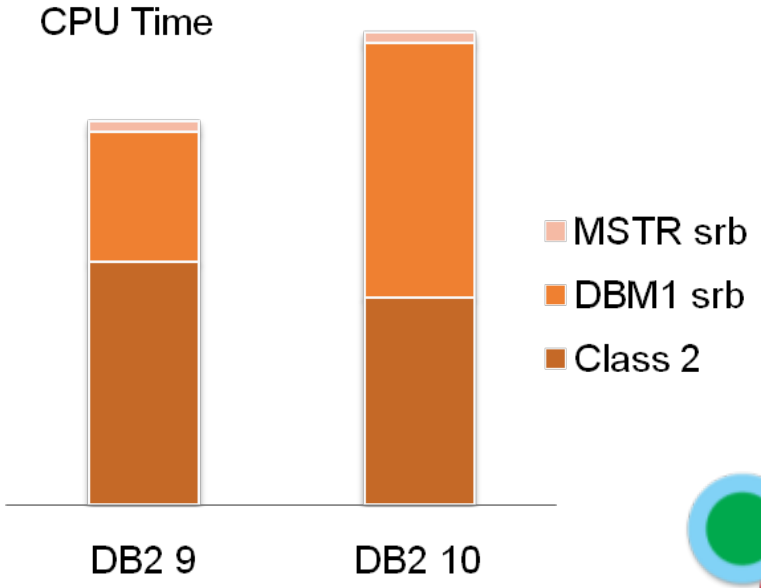
- Insert Index I/O Parallelism

- DB2 9
  - Inserts into indexes are done serially, one at the time
- DB2 10
  - Insert into multiple indexes on the same table done in parallel
  - Overlaps the synchronous I/O wait time for indexes on the same table
  - Significantly improves the performance of I/O bound insert workloads

IMPORTANT: Overall CPU increases, but the DBM1 SRB time is zIIP eligible

2000 random inserts, 6 indexes, 100% cache miss ratio with 15K rpm disk





- Additional Non-key Columns in a Unique Index

- DB2 9 definition

```
CREATE UNIQUE INDEX i1 ON t1(c1);

CREATE INDEX i2 ON t1(c1,c2,c3);
```

- Possible DB2 10 definition

```
CREATE UNIQUE INDEX i1 ON t1(c1) INCLUDE (c2,c3)
```

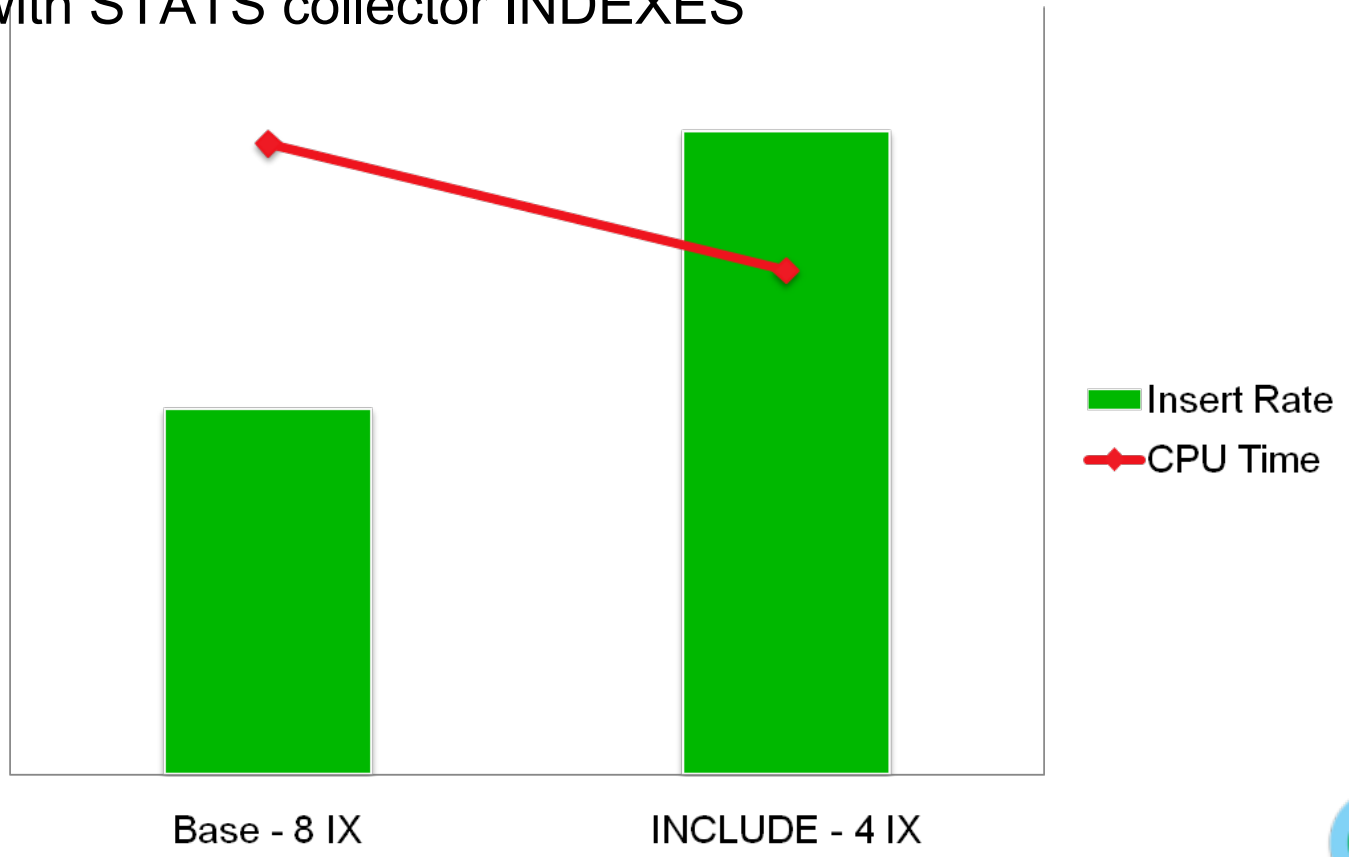
or

```
ALTER INDEX i1 ADD INCLUDE (c2,c3)
```

- Some restrictions apply:
  - INCLUDE columns are not allowed in non-unique indexes
  - Indexes on Expression will not support INCLUDE columns

- Insert Performance with INCLUDE index

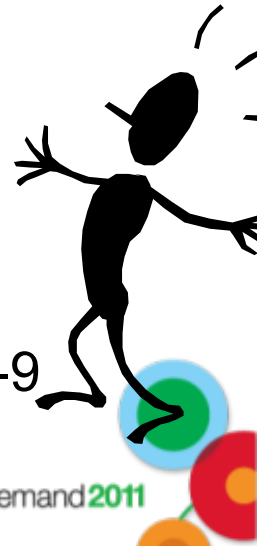
- Potential performance improvement by reduction of number of indexes
- Check RTS and LASTUSED for deciding what INDEX to drop
  - Be careful with STATS collector INDEXES



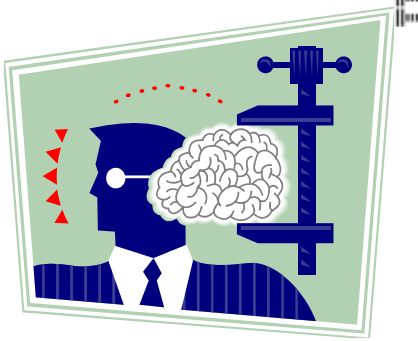
- DB2 10 and zIIP

IMPORTANT: DB2 10 EXTENDS the scope of the zIIP eligible workload

- 100% of prefetch and deferred write engines
  - Significant with index compression and Insert index I/O parallelism
- RUNSTATS
  - With no additional parameters up up to 99% of zIIP eligibility
  - With more complex statistics, like frequency statistics, less offload
- In z/OS V1R11 DFSORT allows additional zIIP redirect for DB2 utilities
  - Included with DB2 10
  - PK85889: UTILITY FUNCTION ENHANCEMENT for DB2 V8-9
- zAAP on zIIP?



- More DB2 10 and zIIP



- Parsing process of XML Schema validation
  - 100% of new validation parser is eligible
  - Offload to zIIP, zAAP, or zAAP on zIIP
  - Retro fit into DB2 9 via PK90032 and PK90040
- zIIP eligibility for DRDA workloads increased from 53% to 60%
  - Extended to DB2 V8 and DB2 9 via PM12256: DRDA PERFORMANCE IMPROVEMENT USING TCP/IP
  - Recommended: PM28626: CORRECTION OF DRDA USING TCP/IP EXECUTION VARIATION AND HANDLING OF ENCLAVE CONTROL STRUCTURE ANOMALIES

SYSTEM	CPUZIPTM	CPUTM	zIIP Ratio	SERVICE U	SU / TXN
PR01	75.21	46.64	61.72%	21761495	192579.60
PR02	65.12	54.39	54.49%	21904159	199128.72

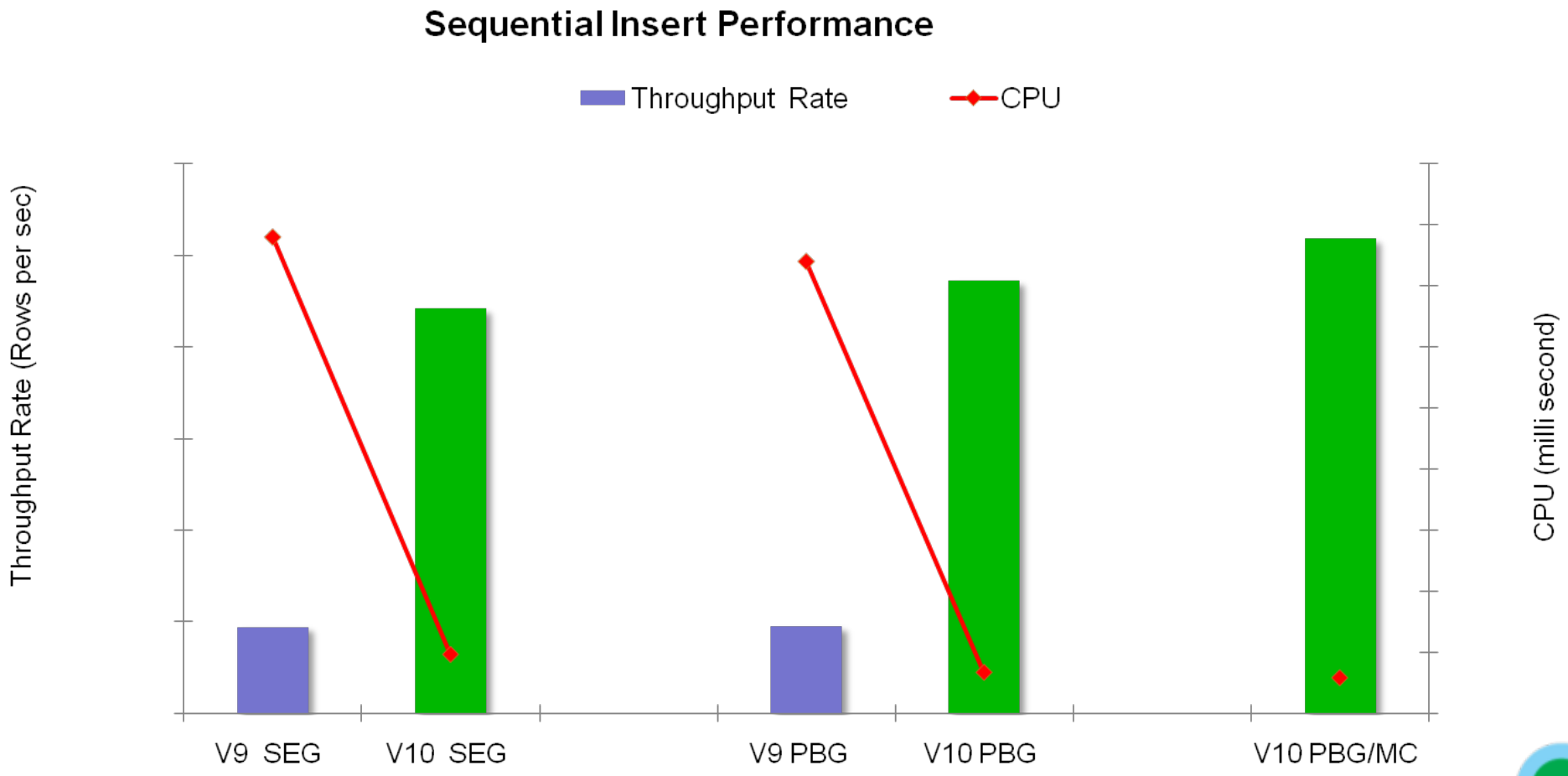
- DB2 10 INSERT Performance improvements

- DB2 10 CM
  - Space search improvement
  - Index I/O parallelism
  - Log latch contention reduction and faster commit process
  - Additional index look aside
- DB2 10 NFM
  - INCLUDE index
  - Support Member Cluster in UTS
  - Complete LRSN spin avoidance
- Universal Table Space & Member Cluster
  - Remove hot spots in concurrent sequential insert in data sharing
  - Assigns a set of pages and associated space map page to each member
  - It does not maintain data cluster during the INSERT
  - Data cluster needs to be restored via REORG

EXAMPLE: `ALTER TABLESPACE TSP1 MEMBER CLUSTER YES/NO;`

- DB2 10 INSERT Performance Improvement

- Sequential key insert into 3 tables from JDBC
- > 200 clients in a two way Data Sharing system

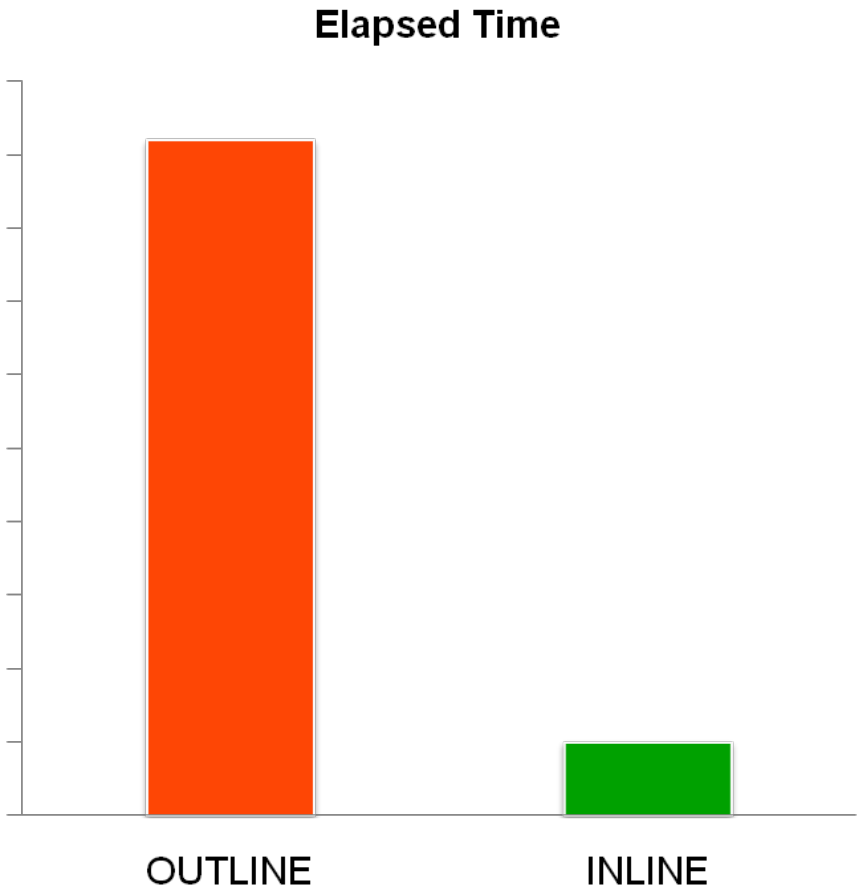


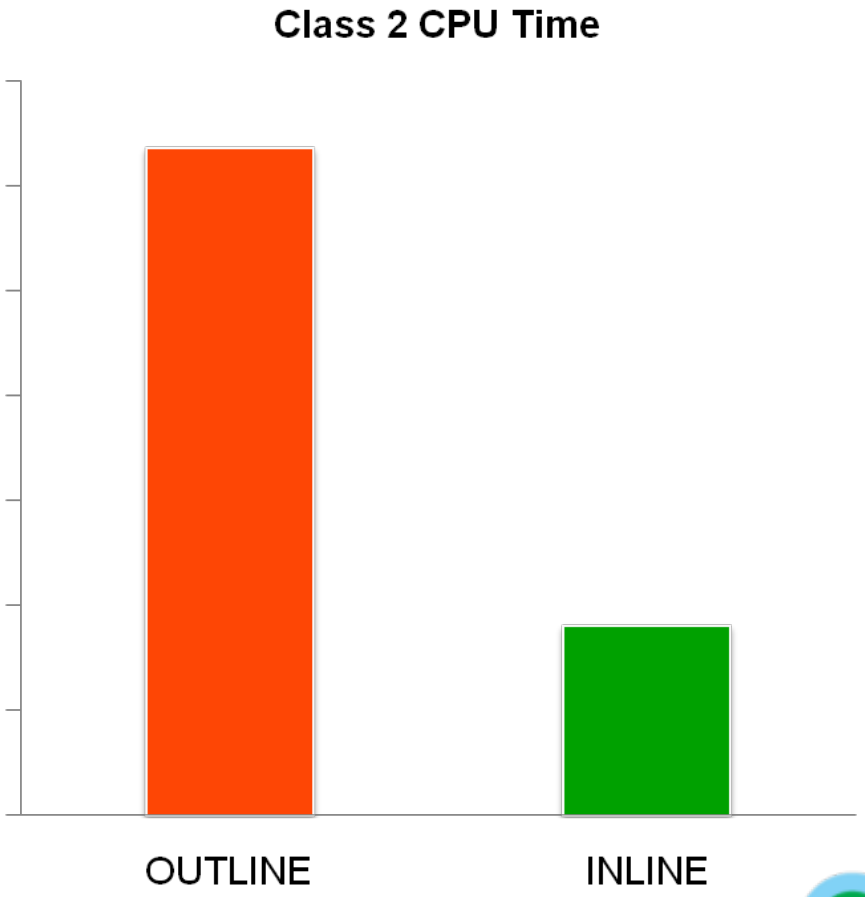
- Inline LOBs (NFM)

- Inline LOB
  - A LOB resides completely in the base table along with other non-LOB columns
  - A LOB resides partially in the base table along with other non-LOB columns and partially in the LOB table space
- Benefits
  - Small LOBs can be saved in the base table with non-LOB columns and achieve similar performance
  - In-lined piece of LOB can be indexed via index on expression
  - Can COMPRESS in-line part of the LOB
  - LOAD and UNLOAD utility are able to load and unload the complete LOB along with other non-LOB columns

- Sequential Inserts with small LOBs

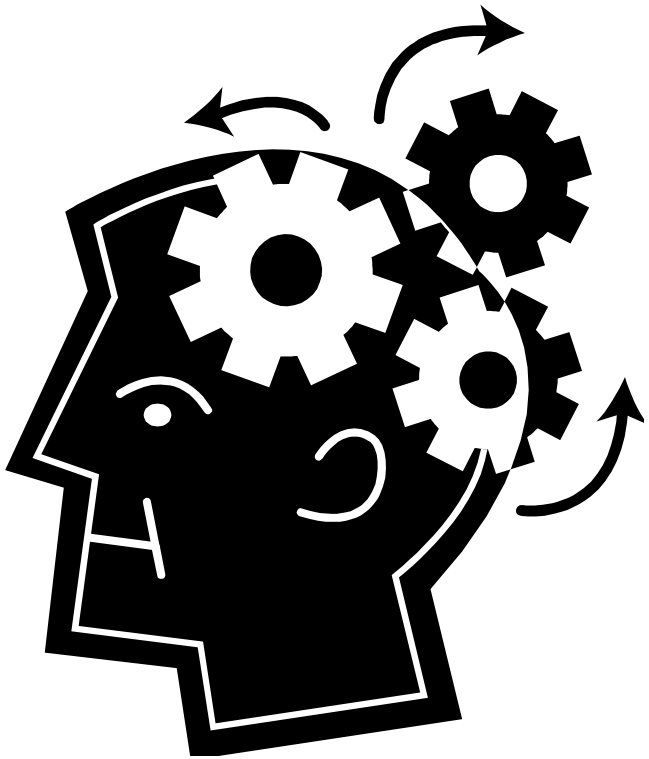
- Insert 10,000 x 200 byte LOBs
- 20000 Inserts/Commit





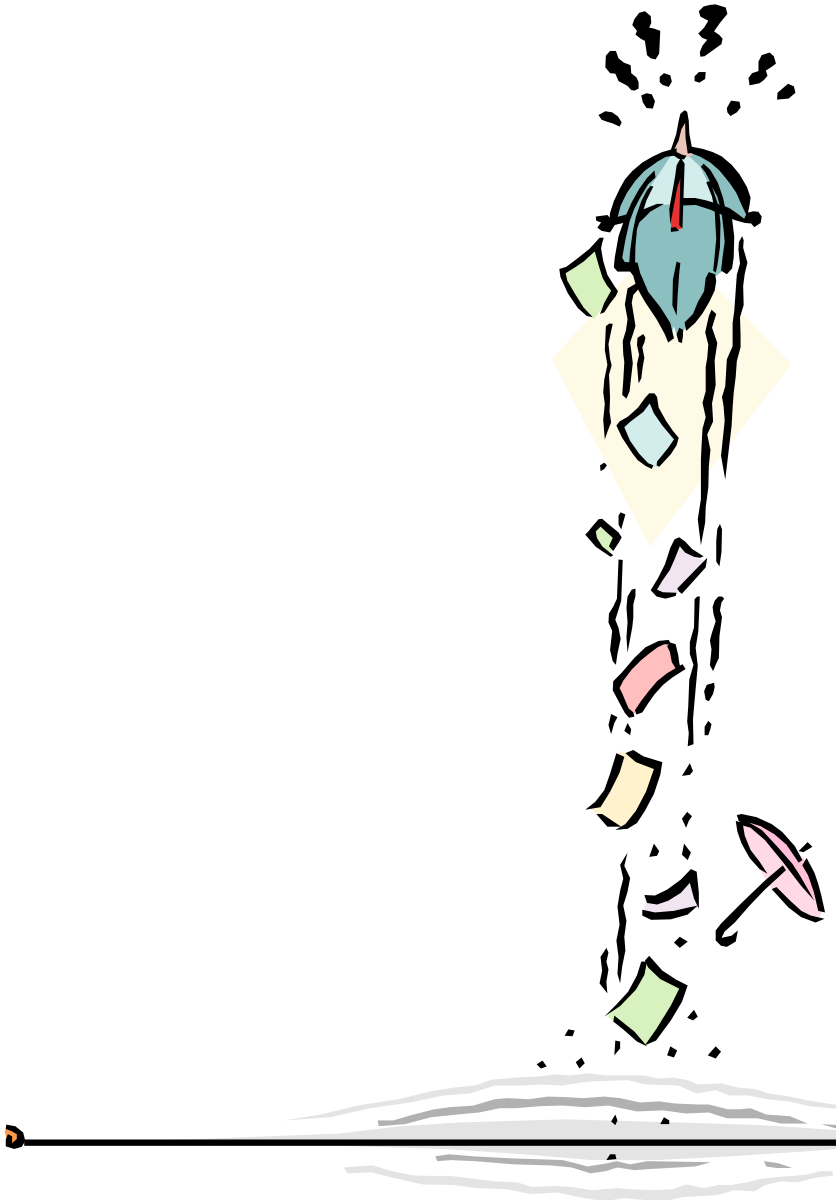
- Summary

- A lot of excellent work has been done to optimize and improve DB2 10 performance
- Key concept:
  - Many performance improvements in CM
- DB2 10 is inevitable…
  - It is good to know that a good DB2 release is waiting for you
- How much will you save with DB2 10?
  - This may be the wrong question
  - It depends…



THANKS!

- Cristian Molaro
  - cristian@molaro.be
  - @cristianmolaro



Thank You!
Your Feedback is Important to Us

- Access your personal session survey list and complete via SmartSite
  - Your smart phone or web browser at: iodsmartsite.com
  - Any SmartSite kiosk onsite
  - Each completed session survey increases your chance to win an Apple iPod Touch with daily drawing sponsored by Alliance Tech