What's new from the Optimizer in DB2 10 for z/OS?

Session Number 1845

Terry Purcell, IBM

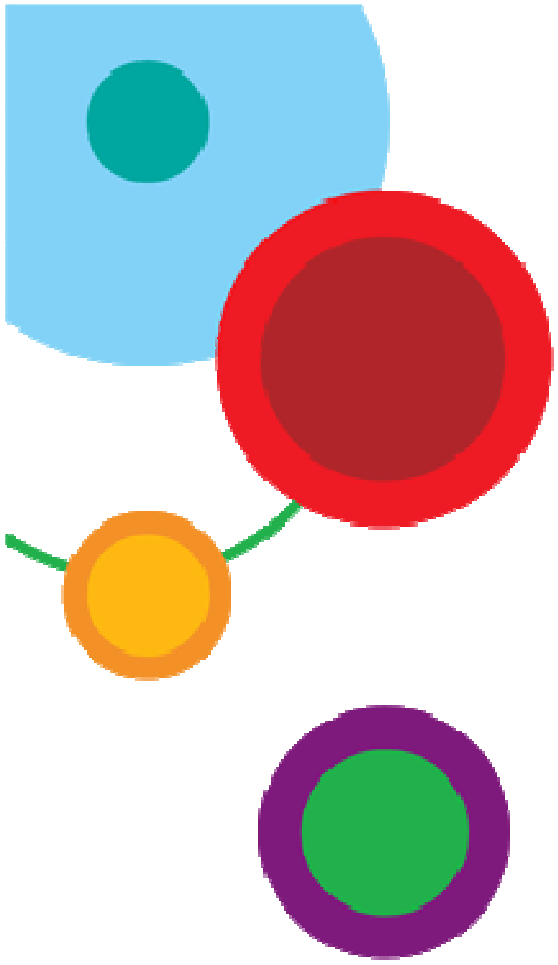
IBM Software
Information On Demand 2011

Acknowledgements and Disclaimers:

Availability. References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

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.



Agenda

- Access path management
  - Dynamic Statement Cache Enhancements
  - Access Path Stability
  - Instance Based Statement Hints/Options
- Query performance improvements
- Runstats usability and performance improvements

Literal Replacement

- Dynamic SQL with literals can now be re-used in the cache
  - Literals replaced with &
    - Similar to parameter markers but not the same
- To enable either you:-
  - Put 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
- Lookup Sequence
  - Original SQL with literals is looked up in the cache
  - If not found, literals are replaced and new SQL is looked up in the cache
    - Additional match on literal usability
    - Can only match with SQL stored with same attribute, not parameter marker
  - If not found, new SQL is prepared and stored in the cache

Literal Replacement …

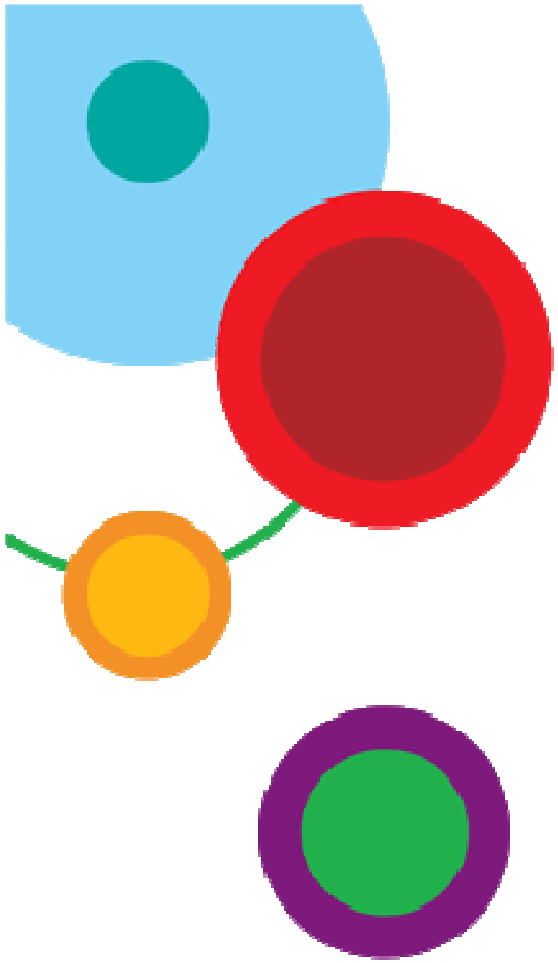
- Example:

```
WHERE ACCOUNT_NUMBER = 123456
```

  - This would be replaced by

```
WHERE ACCOUNT_NUMBER = &
```

- Performance Expectation
  - Using parameter marker still provides best performance
  - Biggest performance gain for repeated SQL with different literals
  - NOTE: Access path is not optimized for literals
    - Neither are access paths based on parameter markers/host variables
    - Need to use REOPT for that purpose
    - ***LIKE predicates not supported



Agenda

- Access path management
  - Dynamic Statement Cache Enhancements
  - Access Path Stability
  - Instance Based Statement Hints/Options
- Query performance improvements
- Runstats usability and performance improvements

DB2 10 Updates to Plan Management

- SYSIBM.SYSPACKCOPY
  - New catalog table
  - Hold SYSPACKAGE-style metadata for any previous/original package copies
  - No longer need to SWITCH to see information on inactive copies
    - Complaint from DB2 9

- APRETAINDUP option of REBIND
  - Default YES
    - Retain duplicate for BASIC or EXTENDED
  - Optional NO
    - Do not retain duplicate access path as PREVIOUS or ORIGINAL
      - PREVIOUS/ORIGINAL must be from DB2 9 or later
  - NOTE: APAR PM27811 supports inline LOB (and compression) for SPT01

Access Path Reuse / Compare For Packages

- Can we retain the same underlying access paths, and hence, minimize impact?
  - **BIND/REBIND PACKAGE … APREUSE(ERROR)**
    - For REBIND, DB2 attempts to reuse prior access paths
    - For BIND, DB2 attempts the reuse prior access paths for any queries that haven't changed
    - Error returned (and bind fails) if reuse of access path not possible
  - **BIND/REBIND PACKAGE… APCOMPARE(WARN | ERROR)**
    - DB2 issues a warning/error for each statement that has an access path change
- Prior package must have been bound on V9 or V10 (not V8 or prior)
- Not available at GA…..but
  - Function available with APAR PM25679 (closed July 2011)!!!

Retrieving Access Path with EXPLAIN(NO)

- EXPLAIN PACKAGE
  - Extract PLAN_TABLE information for packages
  - The package/copy must be created on DB2 9 or later
  - Useful if you didn't BIND with EXPLAIN(YES)
    - Or PLAN_TABLE entries are lost

```
>>-EXPLAIN----PACKAGE----------->

>>-----COLLECTION--collection-name--PACKAGE--package-name--------->

>----+--------------------------+----+-------------------+-------->
     |                          |    |                   |
     +---VERSION-version-name---+    +---COPY--copy-id---+
```

- COPY-ID can be 'CURRENT', 'PREVIOUS', 'ORIGINAL'

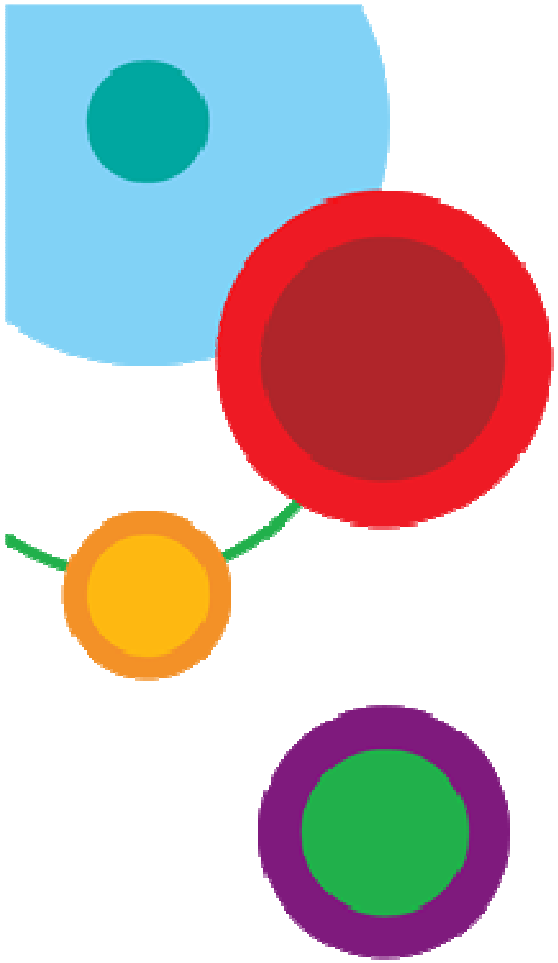
What if ? for BIND/REBIND

- Bind/Rebind package EXPLAIN(ONLY) & SQLERROR(CHECK)
  - Existing package copies are not overwritten
    - Performs explain or syntax/semantic error checks on SQL
  - Requires BIND, BINDAGENT, or EXPLAIN privilege.
  - Supported for BIND only at GA
    - Targeted to application changes
      - Eg. Development environment is DB2 LUW, production DB2 for z/OS
    - REBIND added in APAR PM25679

What is the difference of each EXPLAIN usage?

- BIND/REBIND with EXPLAIN(YES)
  - Generates a new access path, populates PLAN_TABLE and creates new package
- BIND/REBIND with EXPLAIN(ONLY)
  - Generates a new access path, populates PLAN_TABLE, but does NOT create a new package

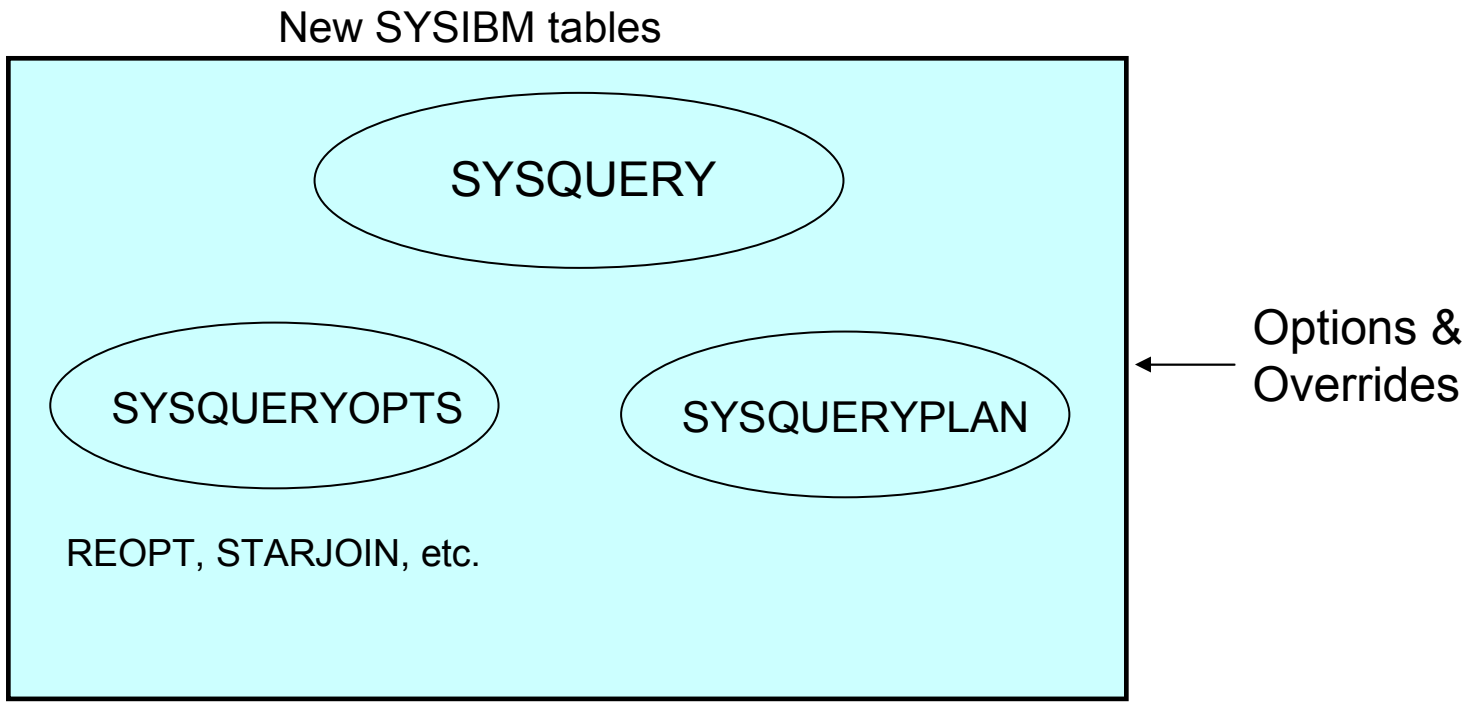
- EXPLAIN PLAN (issued in SPUFI/QMF/DSNTEP2 etc)
  - Generates a new access path and populates PLAN_TABLE
- EXPLAIN PACKAGE
  - Does not generate new access path. Extracts existing access path from package and populates PLAN_TABLE.



Agenda

- Access path management
  - Dynamic Statement Cache Enhancements
  - Access Path Stability
  - Instance Based Statement Hints/Options
- Query performance improvements
- Runstats usability and performance improvements

Access Path Repository – Hints/Statement level



Access Path Stability with statement level hints

- Current limitations in hint matching
  - QUERYNO is used to link queries to their hints – a bit fragile
  - For dynamic SQL, require a change to apps – can be impractical
- New mechanisms:
  - Associate query text with its corresponding hint … more robust
  - Hints can be enforced for the entire DB2 subsystem
    - irrespective of static vs. dynamic, etc.
  - Hints integrated into the access path repository
- PLAN_TABLE isn't going away
  - Only the "hint lookup" mechanism is being improved.

Statement level hints (cont.)

- Steps to use new hints mechanism
  - Enable OPTHINTS zparm
  - Populate a user table DSN_USERQUERY_TABLE with query text
    - Insert from SYSPACKSTMT (static) or DSN_STATEMENT_CACHE_TABLE (dynamic)
  - Populate PLAN_TABLE with the corresponding hints
    - QUERYNO must match between PLAN_TABLE & DSN_USERQUERY_TABLE
  - Run new command BIND QUERY
    - To integrate the hint into the repository.
    - Next package bind or dynamic prepare can pickup hint.
  - FREE QUERY can be used to remove the query.

Statement-level BIND options

- Statement-level granularity may be required rather than:
  - Subsystem level ZPARMs (STARJOIN, SJTABLES, MAX_PAR_DEGREE)
  - Package level BIND options (REOPT, DEF_CURR_DEGREE)
- For example
  - Only one statement in the package needs REOPT(ALWAYS) 
- New mechanism for statement-level bind options:
  - Similar to mechanism used for hints
    - Enable OPTHINTS zparm
    - Populate DSN_USERQUERY_TABLE with query text and desired option
      - Use a QUERYNO that is NOT in PLAN_TABLE
    - Issue BIND QUERY
    - Next package bind/rebind or dynamic prepare can pickup statement option
    - FREE QUERY can be used to remove the query

Agenda

- Access path management
- Query performance improvements
  - Predicate application
    - IN-list
    - Complex ORs
    - View/Table expr Merge
    - Stage 2 predicate pushdown
  - Safe Query Optimization
  - Parallelism Enhancements
  - Misc Query Perf Enhancements
- Runstats usability and performance improvements

16

Improvements to predicate application

- **Major enhancements to OR and IN predicates**
  - Improved performance for AND/OR combinations and long IN-lists
    - General performance improvement to stage 1 predicate processing
  - IN-list matching
    - Matching on multiple IN-lists
    - Transitive closure support for IN-list predicates
    - List prefetch support
    - Trim IN-lists from matching when preceding equals are highly filtering
  - SQL pagination
    - Single index matching for complex OR conditions
- **Many stage 2 expressions to be executed at stage 1**
  - Stage 2 expressions eligible for index screening

IN-list Table - Table Type 'I' and Access Type 'IN'

- The IN-list predicate will be represented as an in-memory table if:
  - List prefetch is chosen, OR
  - More than one IN-list is chosen as matching.
  - The EXPLAIN output associated with the in-memory table will have:
    - New Table Type: TBTYPE – 'I'
    - New Access Type: ACTYPE – 'IN'

```
SELECT *
    FROM T1
    WHERE T1.C1 IN (?, ?, ?);
```

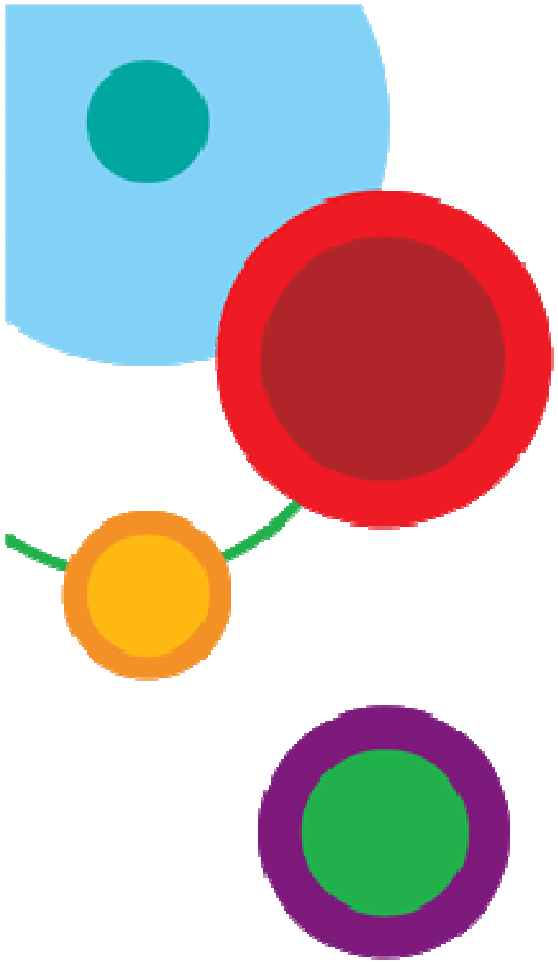
QBNO	PLANNO	METHOD	TNAME	ACTYPE	MC	ACNAME	QBTYPE	TBTYPE	PREFETCH
1	1	0	DSNIN001(01)	***IN***	0		SELECT	***I***	
1	2	1	T1	I	1	T1_IX_C1	SELECT	T	***L***

IN-list Predicate Transitive Closure (PTC)

```
SELECT *
FROM T1, T2
WHERE T1.C1 = T2.C1
  AND T1.C1 IN (?, ?, ?)

  AND T2.C1 IN (?, ?, ?) ←   Optimizer can generate
                             this predicate via PTC
```

- Without IN-list PTC (DB2 9)
  - Optimizer will be unlikely to consider T2 is the first table accessed
- With IN-list PTC (DB2 10)
  - Optimizer can choose to access T2 or T1 first.



Agenda

- Access path management
- Query performance improvements
  - Predicate application
    - IN-list
    - Complex ORs
    - View/Table expr Merge
    - Stage 2 predicate pushdown
  - Safe Query Optimization
  - Parallelism Enhancements
  - Misc Query Perf Enhancements
- Runstats usability and performance improvements

SQL Pagination targets 2 classes of OR queries:

- Cursor scrolling (pagination) SQL
  - Retrieve next n rows
  - Common in COBOL/CICS and any screen scrolling application
  - Not to be confused with “scrollable cursors”
    - Hence term pagination to avoid confusion (???)
- Complex OR predicates against the same columns
  - Common in SAP

- In both cases:
  - The OR (disjunct) predicate refers to a single table only.
  - Each OR predicate can be mapped to the same index.
  - Each disjunct has at least one matching predicate.

Simple scrolling – Index matching and ORDER BY

- Scroll forward to obtain the next 20 rows
  - Assumes index is available on (LASTNAME, FIRSTNAME)
  - WHERE clause may appear as:

```
WHERE (LASTNAME='JONES' AND FIRSTNAME>'WENDY')
   OR (LASTNAME>'JONES')
ORDER BY LASTNAME, FIRSTNAME;
```

  - DB2 10 supports
    - Single matching index access with sort avoided
  - DB2 9 requires
    - Multi-index access, list prefetch and sort
    - OR, extra predicate (AND LASTNAME >= 'JONES') for matching single index access and sort avoidance

Complex OR predicates against same index

- Given WHERE clause
  - And index on one or both columns

```
WHERE (LASTNAME='JONES' AND FIRSTNAME='WENDY')
   OR (LASTNAME='SMITH' AND FIRSTNAME='JOHN');
```

- DB2 9 requires
  - Multi-index access with list prefetch

- DB2 10 supports
  - Matching single index access – no list prefetch
  - Or, Multi-index access with list prefetch

SQL Pagination – PLAN_TABLE representation

- Order of PLAN_TABLE entries is by coding sequence
  - Determination of execution sequence deferred to runtime
    - When all host variables/parameter markers are resolved
  - For this example, coding seq does not match execution seq

```
WHERE (LASTNAME>'JONES')
   OR (LASTNAME='JONES' AND FIRSTNAME>'WENDY')
ORDER BY LASTNAME, FIRSTNAME;
```

QBlockno	Planno	Accessname	Access_Type	Matchcols	Mixopseq
1	1	IX1	NR	1	1
1	1	IX1	NR	2	2

New access type (NR = IN-List Range)

Coding seq

Agenda

- Access path management
- Query performance improvements
  - Predicate application
    - IN-list
    - Complex ORs
    - View/Table expr Merge
    - Stage 2 predicate pushdown
  - Safe Query Optimization
  - Parallelism Enhancements
  - Misc Query Perf Enhancements
- Runstats usability and performance improvements

View/Table Expression Merge

- Merge enhancements for View/Table Expressions
  - With outer joins.
- Physical materialization is an overhead.
  - Can limit the join sequence considered.
  - Can limit the ability to apply predicates early
  - Cannot create indexes on materialized work files
- Generally merge is preferred over materialization

Merge expression on preserved side of Outer Join

- DB2 can merge view/table expr on preserved side of outer join
  - CASE, VALUE, COALESCE, NULLIF, IFNULL
    - Exception if merged predicate is stage 2

```
SELECT A.C1, B.C1, A.C2, B.C2
FROM T1, (SELECT COALESCE(C1, 0) as C1 ,C2
            FROM T2 ) A              <--table expression 'A' will be Merged
          LEFT OUTER JOIN
                (SELECT COALESCE(C1, 0) as C1 ,C2
                 FROM T3 ) B      <-- B will be Materialized
          ON A.C2 = B.C2
WHERE T1.C2 = A.C2;
```

Merge single table view/table expr with subquery

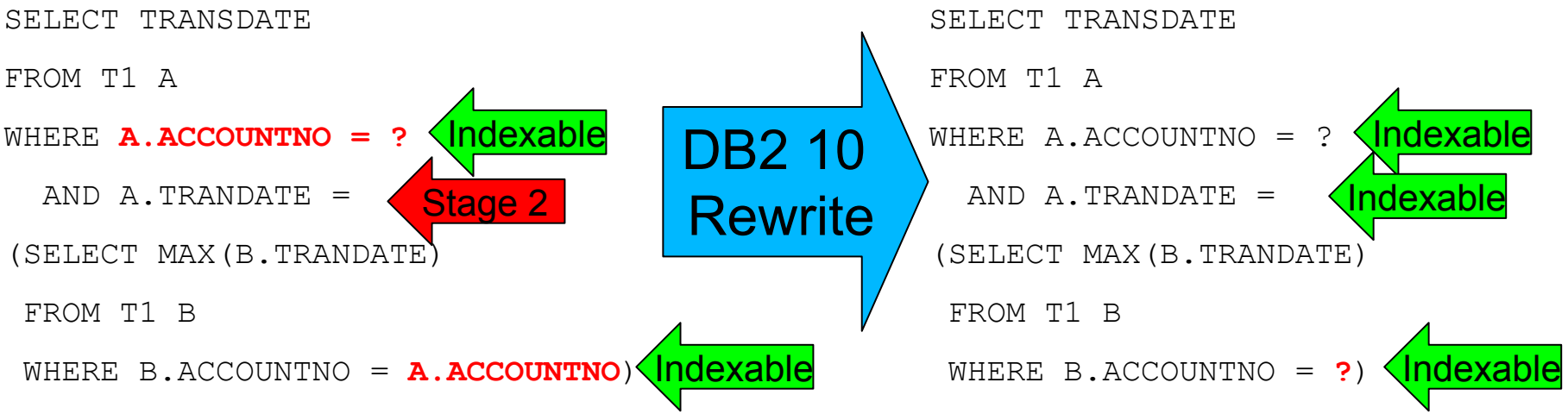
```
SELECT *
FROM T1 LEFT OUTER JOIN
      (SELECT * FROM T2
       WHERE T2.C1 = (SELECT MAX(T3.C1) FROM T3 ) ) TE <--subquery
ON T1.C1 = TE.C1;
```

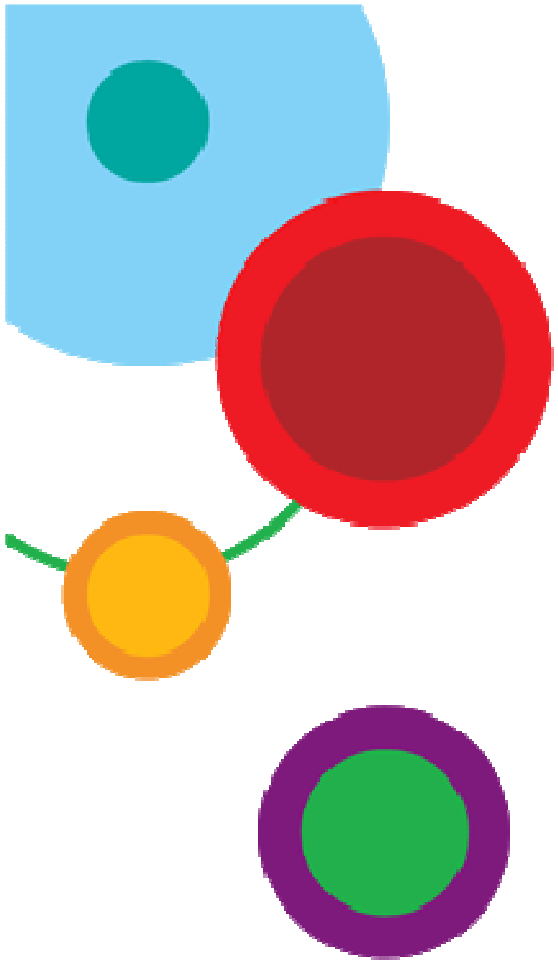
```
SELECT *
FROM T1 LEFT OUTER JOIN T2                    <-- table expression is merged
ON T2.C1 = (SELECT MAX(T3.C1) FROM T3)        <-- subquery ON-predicate
AND T1.C1 = TT.C1;
```

- View/Table expr with subquery on NULL-supplied side
  - Merge into ON clause
- On preserved side
  - Merge into WHERE clause

Correlated to Non-correlated Rewrite

- DB2 10 can rewrite correlated to non-correlated
  - If correlation predicates are covered by local predicates in outer
    - Can result in additional index matching predicate





Agenda

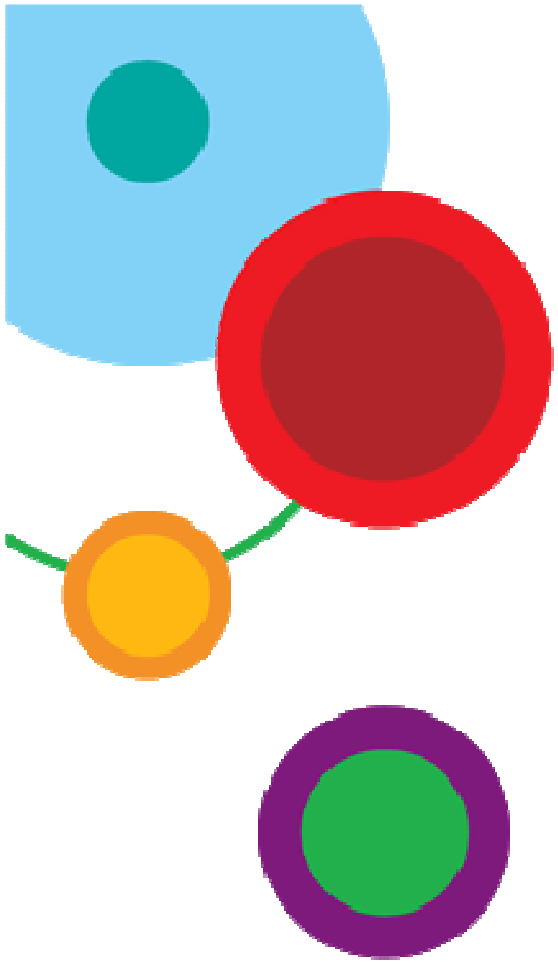
- Access path management
- Query performance improvements
  - Predicate application
    - IN-list
    - Complex ORs
    - View/Table expr Merge
    - Stage 2 predicate pushdown
  - Safe Query Optimization
  - Parallelism Enhancements
  - Misc Query Perf Enhancements
- Runstats usability and performance improvements

Stage 2 predicates “pushed down” to IM/DM

- Most Stage 2 (residual) predicates can execute as index screening (indexable) or as stage 1 (sargable)
  - CPU time improvement
  - Reduced data getpages if stage 2 predicate becomes index screening
  - Applies to
    - Arithmetic/datetime expressions, scalar built-in functions, CASE, CAST, (essentially all expressions without subqueries)
    - OR’d predicates cannot span different predicate stages
- Externalized in DSN_FILTER_TABLE column PUSHDOWN
- Not enabled for List Prefetch access type

Stage 2 predicates “pushed down” to IM/DM

- Examples
  - Suppose there exists index on (C1,C3)
  - … WHERE SUBSTR(C1,1,1) = ? ==> index screening
  - … WHERE SUBSTR(C1,1,1) = ? OR C3 = ? ==> index screening
  - … WHERE SUBSTR(C1,1,1) = ? OR C4 = ? ==> stage 1
  - … WHERE SUBSTR(C1,1,1) = ? AND C4 = ? ==> index screening and stage 1
  - … WHERE SUBSTR(C1,1,1) = ? OR C3 = (SELECT...) ==> stage 2
  - … WHERE SUBSTR(C1,1,1) = ? AND C3 = (SELECT...) ==> index scr. and stage 2



Agenda

- Access path management
- **Query performance improvements**
  - Predicate application
  - **Safe Query Optimization**
  - Parallelism Enhancements
  - Misc Query Perf Enhancements
- Runstats usability and performance improvements

Minimizing Optimizer Challenges

- Potential causes of sub-optimal plans
  - Insufficient statistics
  - Unknown literal values used for host variables or parameter markers

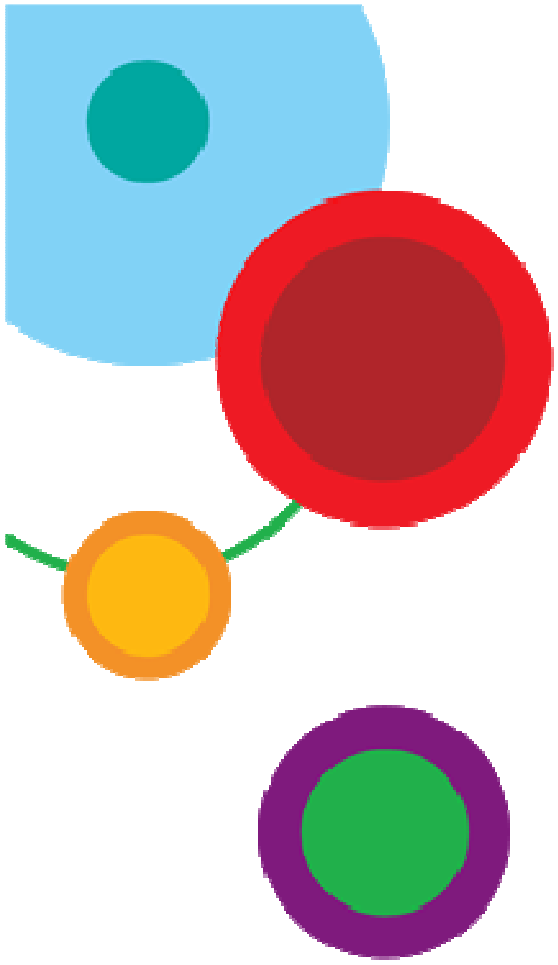
- Optimizer will evaluate the risk for each predicate

  - For example: WHERE BIRTHDATE < ?
    - Could qualify 0-100% of data depending on literal value used
  - As part of access path selection
    - Compare access paths with close cost and choose lowest risk plan

Minimizing impact of RID failure

- RID overflow can occur for
  - Concurrent queries each consuming shared RID pool
  - Single query requesting > 25% of table or hitting RID pool limit
- DB2 9 will fallback to tablespace scan*
- DB2 10 will continue by writing new RIDs to workfile
  - Work-file usage may increase
    - Mitigate by increasing RID pool size (default increased in DB2 10).
    - MAXTEMPS_RID zparm for maximum WF usage for each RID list

* Hybrid join can incrementally process. Dynamic Index ANDing will use WF for failover.



Agenda

- Access path management
- **Query performance improvements**
  - Predicate application
  - Safe Query Optimization
  - **Parallelism Enhancements**
  - Misc Query Perf Enhancements
- Runstats usability and performance improvements

Removal Of Parallelism Restrictions #1

- **Support parallelism for multi-row fetch**
  - **In previous releases**
    - parallelism is disabled for the last parallel group in the top level query block
      - if there is no more table to join after the parallel group
      - and there is no GROUP BY clause or ORDER BY clause
  - Example:- SELECT * FROM CUSTOMER
    - There is no parallel group in the query and there are no table joins
    - There is no GROUP BY clause
    - There is no ORDER BY clause
    - So NO PARALLELISM will be used

- Restriction is only removed if the CURSOR has FOR FETCH/READ ONLY
  - Ambiguous Cursors will not have the restriction removed

Removal Of Parallelism Restrictions #2

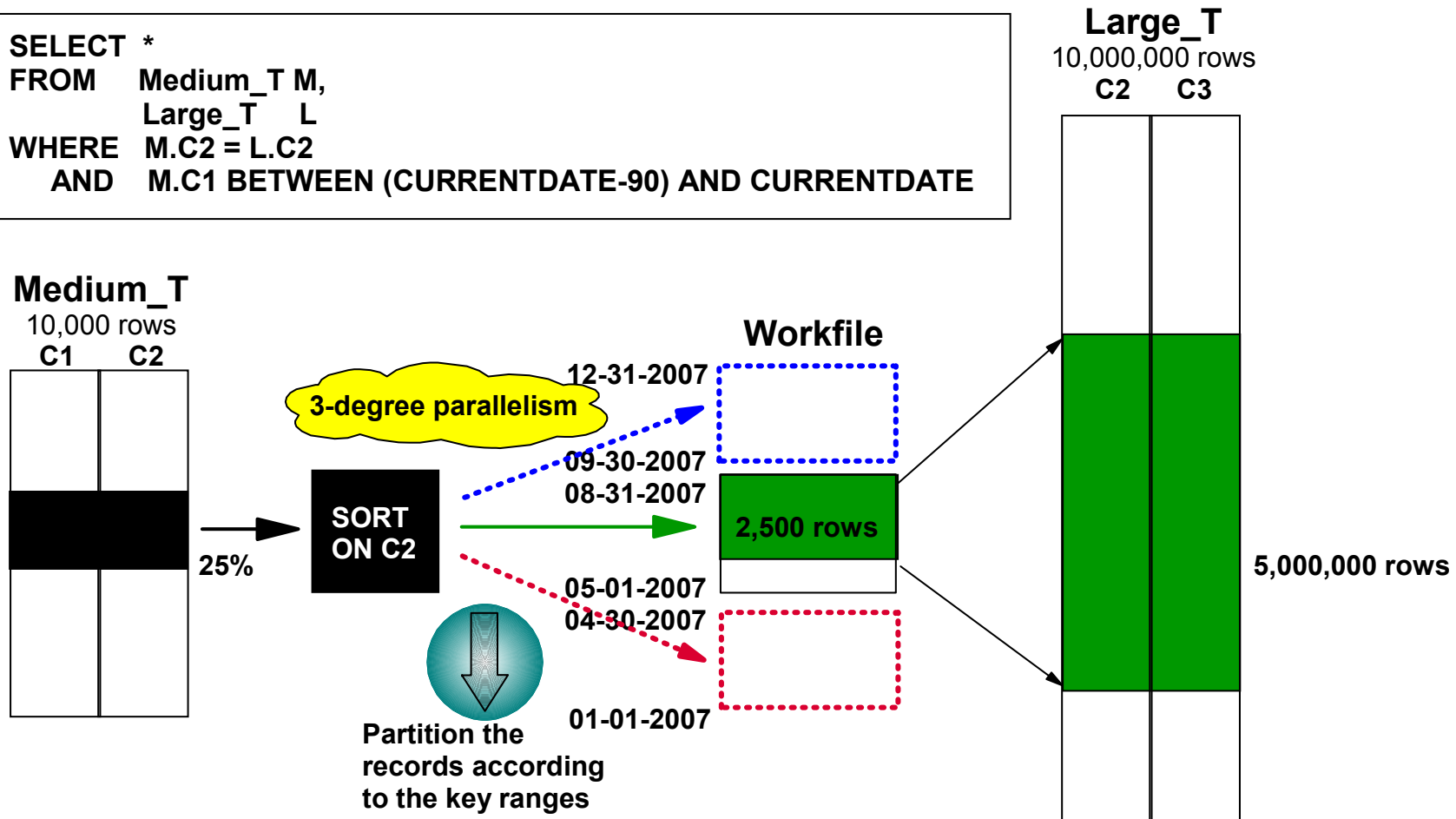
- Allow parallelism if a parallel group contains a work file
  - DB2 generates temporary a work file when view or table expression is materialized
  - This type of work file can not be shared among child task in previous releases of DB2, hence parallelism is disabled
  - DB2 10 will make the work file shareable
    - only applies to CP mode parallelism and no full outer join case

Parallelism Enhancements - Effectiveness

- Previous Releases of DB2 use Key Range Partitioning
  - Key Ranges Decided at Bind Time
  - Based on Statistics (low2key, high2key, column cardinality)
    - Assumes uniform data distribution
    - Histograms can help
      - But rarely collected
  - If Statistics are outdated or data is not uniformly distributed what happens to performance?

Key range partition - Today

```
SELECT  *
FROM    Medium_T M,
        Large_T  L
WHERE   M.C2 = L.C2
  AND   M.C1 BETWEEN (CURRENTDATE-90) AND CURRENTDATE
```



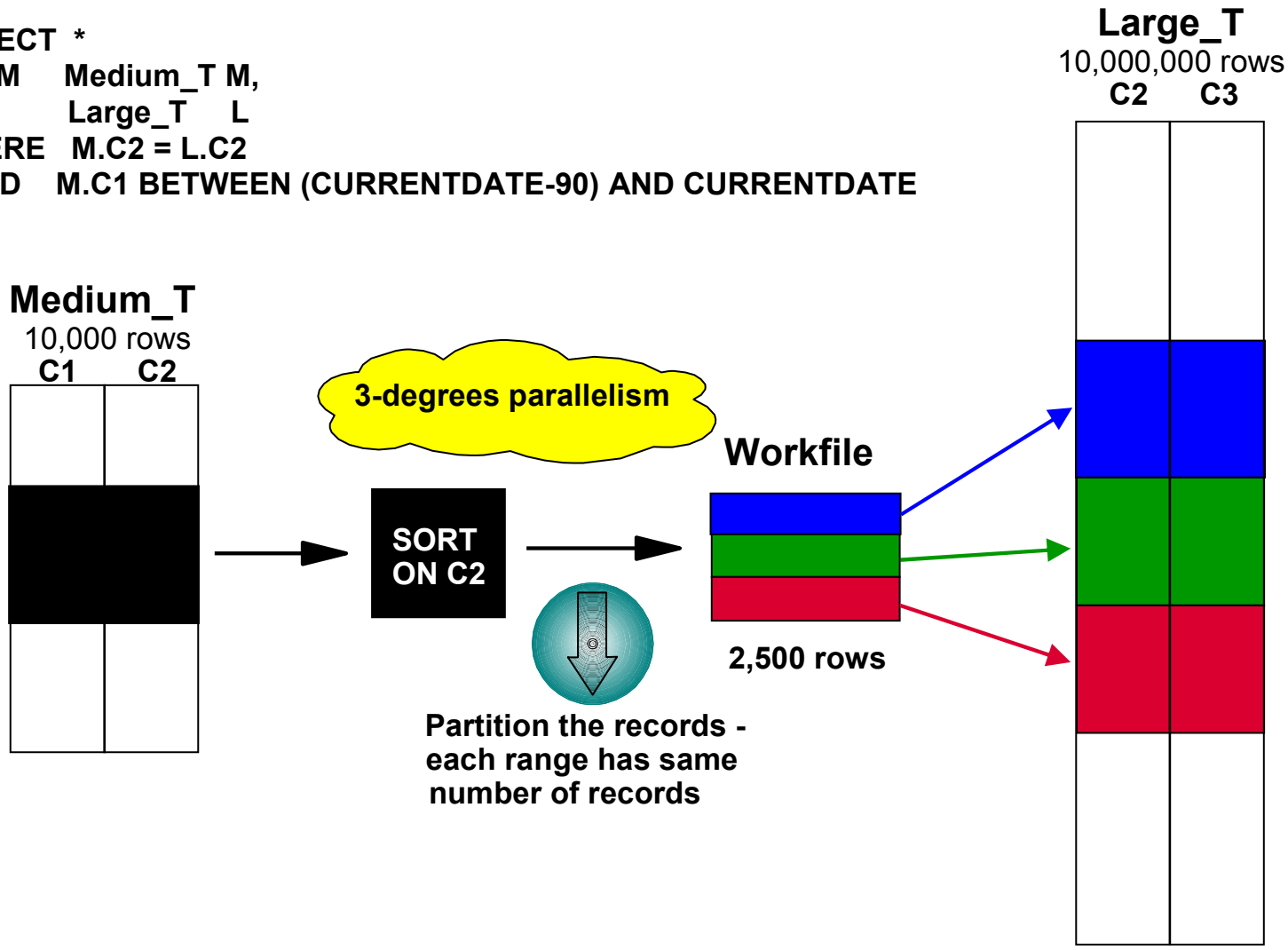
M.C1 is date column, assume currentdate is 8-31-2007, after the between predicate is applied, only rows with date between 06-03-2007 and 8-31-2007 survived, but optimizer chops up the key ranges within the whole year after the records are sorted :-(

Parallelism Effectiveness – Record range

- DB2 10 can use **Dynamic record range partitioning**
  - Materialize the intermediate result in a sequence of join processes
  - Results divided into ranges with equal number of records
  - Division doesn't have to be on the key boundary
    - Unless required for group by or distinct function
  - Record range partitioning is dynamic
    - no longer based on the key ranges decided at bind time
  - Now based on number of composite records and parallel degree
    - Data skew, out of date statistics etc. will not have any effect on performance

Dynamic record range partition

```
SELECT  *
FROM    Medium_T M,
        Large_T   L
WHERE   M.C2 = L.C2
  AND   M.C1 BETWEEN (CURRENTDATE-90) AND CURRENTDATE
```



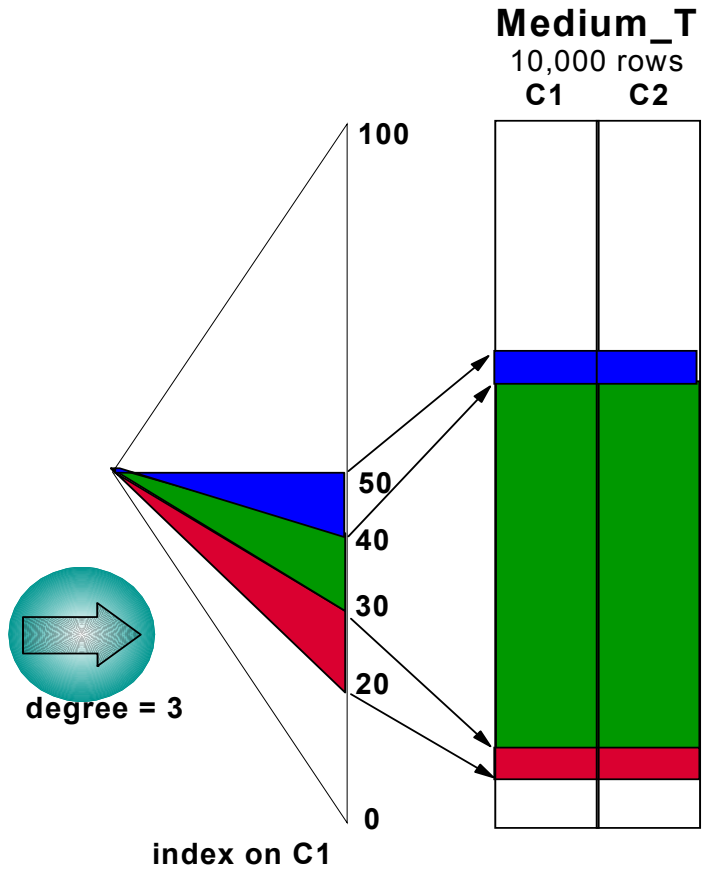
Parallelism Effectiveness - Straw Model

- Previous releases of DB2 divide the number of keys or pages by the number representing the parallel degree
  - One task is allocated per degree of parallelism
  - The range is processed and the task ends
  - Tasks may take different times to process

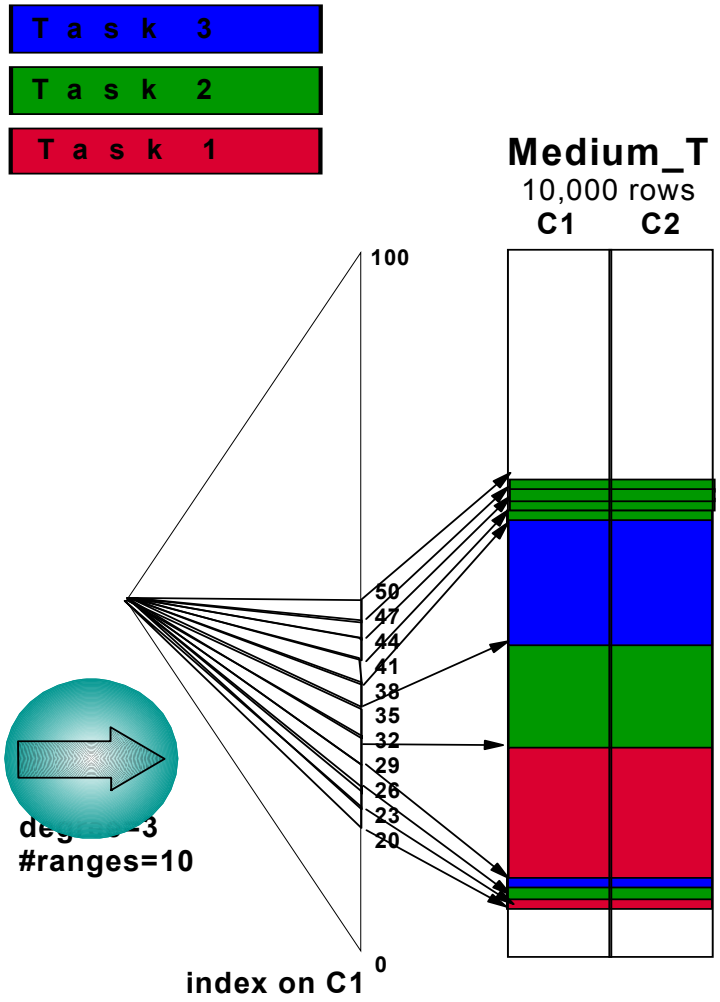
- DB2 10 can use the Straw Model workload distribution method
  - More key or page ranges will be allocated than the number of parallel degrees
  - The same number of tasks as before are allocated (same as degree)
  - Once a task finishes it's smaller range it will process another range
  - Even if data is skewed this new process should make processing faster

STRAW Model

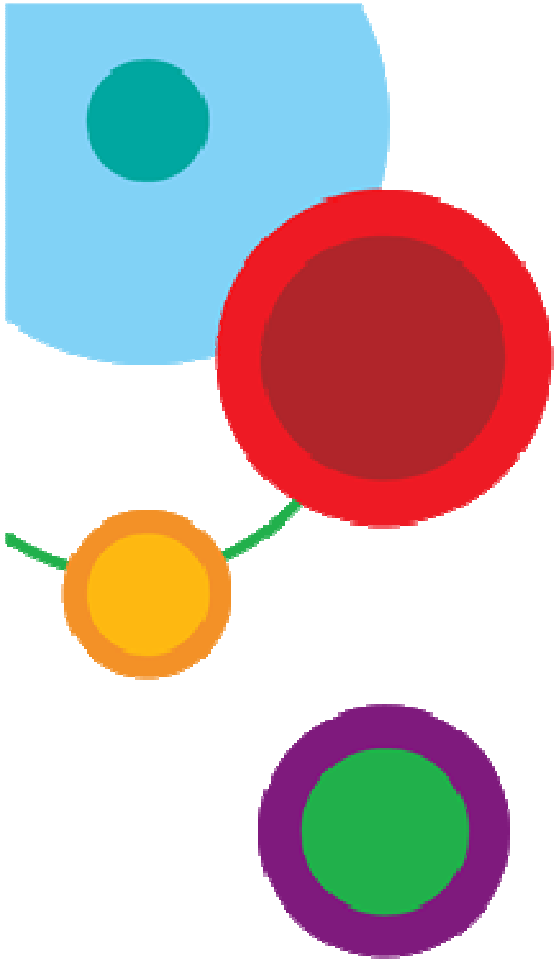
```
SELECT *
FROM   Medium_T M
WHERE  M.C1 BETWEEN 20 AND 50
```



Divided in key ranges before DB2 10



Divided in key ranges with Straw Model



Agenda

- Access path management
- **Query performance improvements**
  - Predicate application
  - Safe Query Optimization
  - Parallelism Enhancements
  - **Misc Query Perf Enhancements**
- Runstats usability and performance improvements

Sort Performance Enhancements

- FETCH FIRST n ROWS ONLY (FFnR) and Sort
  - DB2 9 added in-memory replacement for FFnR to avoid sort
    - Provided (n * (sort key + data)) < 32K
  - DB2 10 extends this to 128K

- Avoid workfile usage for small sorts
  - DB2 9 avoided allocating WF for final sort only
    - If <= 255 rows and result < 32K (sort key + data)
  - DB2 10 extends this to intermediate sorts also
    - Except for parallelism or SET function

- Hash support for large sorts
  - Potential for reduction in number of merge passes

Extending VOLATILE TABLE usage

- VOLATILE TABLE support added in DB2 V8
  - Targeted to SAP Cluster Tables
    - Use Index access whenever possible
    - **Avoids list prefetch**
      - Can be a problem for OR predicates or UPDATEs at risk of loop

- DB2 10 provides VOLATILE to general cases
  - Tables matching SAP cluster tables will maintain original limitations
    - Table with 1 unique index
  - Tables with > 1 index will follow NPGTHRSH rules
    - Use Index access whenever possible
    - **No limitation on list prefetch**
    - Less chance of getting r-scan when list-prefetch plan is only alternative

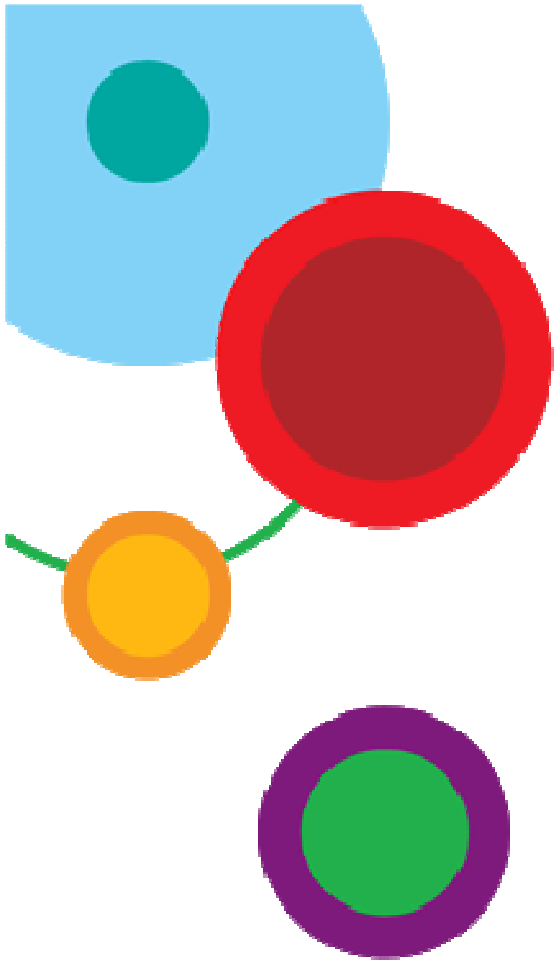
Misc Performance enhancements

- Index INCLUDE columns
  - Create an Index as UNIQUE, and add additional columns
  - Ability to consolidate redundant indexes

```
INDEX1 UNIQUE (C1)
INDEX2 (C1,C2)
```

Consolidate to

```
INDEX1 UNIQUE (C1) INCLUDE (C2)
```



Agenda

- Access path management
- Query performance improvements
- Runstats usability and performance improvements
  - Optimizer Exploitation of Real-Time Stats
  - Auto-Stats
  - RUNSTATS Simplification/Performance

Optimizer Validation with Realtime Stats

- Index Probing & RTS lookup
  - Estimate # of rids within a given start/stop index key range at bind/prepare
- Exploited when these two conditions are met.
  - Query has matching index-access local predicate
  - Predicate contain literals, or REOPT(ALWAYS|ONCE|AUTO)
- And 1 of the following is also true
  - Predicate is estimated to qualify no rows
  - Stats indicate the table contains no rows
  - Table is defined as VOLATILE or qualifies for NPGTHRSH
- New EXPLAIN table to externalize runtime estimates
  - User managed DSN_COLDIST_TABLE

RUNSTATS Problem Summary

- Collecting stats is a difficult and time consuming manual process
  - Need to look at the queries to figure out what stats are needed
  - Need to repeatedly look at the RTS tables to figure out when to recollect
- Inadequate stats collection leads to poor or inconsistent query performance
- Solution is to automate the process
  - More efficient
  - More accurate
  - More stable

Autonomic Statistics Solution Overview

- Autonomic Statistics is implemented though a set of Stored Procedures
  - *Stored procedures are provided to enable administration tools and packaged applications to automate statistics collection.*
    - ADMIN_UTL_MONITOR
    - ADMIN_UTL_EXECUTE
    - ADMIN_UTL_MODIFY
  - Working together, these SP's
    - Determine what stats to collect
    - Determine when stats need to be collected
    - Schedule and Perform the stats collection
    - Records activity for later review
  - *See Chapter 11 "Designing DB2 statistics for performance" in the DB2 10 for z/OS Performance Monitoring and Tuning Guide for details on how to configure autonomic monitoring directly within DB2.*

RUNSTATS Simplification/Performance Overview

- RUNSTATS options to SET/UPDATE/USE a stats profile
  - Integrate specialized statistics into generic RUNSTATS job
    - RUNSTATS … TABLE tbl COLUMN(C1)… **SET PROFILE**
      - Alternatively use **SET PROFILE FROM EXISTING STATS**
    - RUNSTATS … TABLE tbl COLUMN(C5)… **UPDATE PROFILE**
    - RUNSTATS … TABLE tbl **USE PROFILE**

- New option for page-level sampling
  - But what percentage of sampling to use?
    - RUNSTATS … TABLE tbl **TABLESAMPLE SYSTEM AUTO**

  - NOTE: Sampling results in fewer pages (rows) read
    - Thus, RUNSTATS will not see all possible values

Information Management Communities

- **On-line communities, User Groups, Technical Forums, Blogs, Social networks, and more**
  - Find the community that interests you…
    - **World of DB2 for z/OS** http://db2forzos.ning.com/
    - **Information Management** ibm.com/software/data/community
    - **Business Analytics** ibm.com/software/analytics/community
    - **International DB2 User Group** www.idug.org

- **IBM Champions**
  - Recognizing individuals who have made the most outstanding contributions to Information Management, Business Analytics, and Enterprise Content Management communities
    - ibm.com/champion

Useful DB2 for z/OS URLs

- **DB2 10 Launch Website** → http://bit.ly/DB210Launch
- **DB2 for z/OS Website** → http://www-01.ibm.com/software/data/db2/zos/
- **DB2 Product Library** → http://www.ibm.com/software/data/db2/zos/library.html
- **DB2 Newsletter** → http://www.ibm.com/vrm/newsletter/11065

- **Latest Whitepapers**
  - Business Value of DB2 10 – Julian Stuhler
  - A Matter of Time: Temporal Data Management
  - Why DB2 for z/OS is BETTER than Oracle RAC ?
- **DB2 for z/OS e-Kit** → http://bit.ly/DB210e-Kit
- **Upcoming Conferences/Events**
  - **IDUG DB2 Tech Conference EMEA Prague - 13th -18th Nov 2011**
  - **DB2 10 Migration Planning Workshop Prague - 13th November 2011**
  - **IDUG DB2 Tech Conference AG Denver - 14th -16th May 2012**
  - **IDUG 10 Migration Planning Workshop Denver - 13th May 2012**

Top DB2 for z/OS e-Communities

- World of DB2 for z/OS - 1700+ members → http://db2forzos.ning.com/
- DB2 10 LinkedIn - 1000+ members → http://linkd.in/IBMDB210
- DB2 for z/OS What's On LinkedIn – 2000+ members → http://linkd.in/kd05LH
- DB2 for z/OS YouTube → http://www.youtube.com/user/IBMDB2forzOS
- WW IDUG LinkedIn Group - 2000 +members → http://linkd.in/IDUGLinkedIn
- IBM DeveloperWorks → http://www.ibm.com/developerworks/data/community/

DB2 10 is Here !

Customers seeing reduced costs, simplified workloads through proven technology

Reduced Costs	Simplified Workloads	Proven Technology
"Based on the performance metrics from our controlled test environment, we see a significant amount of CPU and Elapsed time savings. This release has many features that will help bring down our operating costs." Morgan Stanley Morgan Stanley DB2 Team	"With DB2 10 able to handle 5-10 times as many threads as the previous version, the upgrade will immediately give the bank some much-needed room for future workload growth while simultaneously reducing their data sharing overhead." BANCO DO BRASIL Paulo Sahadi - Senior Production Manager, Information Management Division at Banco do Brasil	"Every single SQL statement we have tested has been better or the same as our current optimal paths – we have yet to see any significant access path regression. We had to spend a lot of time tuning SQL with DB2 9, but we expect that to disappear when we upgrade to DB2 10." Philipp Nowak, BMW DB2 Product Manager
"We are particularly interested in the performance improvements due to the potential CPU reductions that we realized during our DB2 10 Beta testing. Our early testing has shown out-of-the-box processing cost reductions of between 5% - 10% and for some workloads as high as 30%. Potential cost savings of this magnitude cannot be ignored given today's business climate."  Large Global Bank	"The new temporal functionality in DB2 10 for z/OS will allow us to drastically simplify our date-related queries. In addition, we'll be able to reduce our storage costs by using cheaper storage for inactive rows and reduce our processing cost by having DB2 handle data movement more efficiently than the custom code we've written to do the same work in the past" Large Insurance Company	The new audit capabilities in DB2 10 will allow tables to be audited as soon as they are created, which is an obvious benefit for the business and will reduce costs and simplify our processes" Guenter Schinkel -Postbank Systems AG Postbank

For more customer references visit : http://www.ibm.com/software/data/db2/zos/testimonials.html

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
- Visit Us at

  www.ibm.com/software/data/db2/db210