DB2 Performance in Practice

Steve McDougall
Independent Database Consultant

BMC DB2 Performance Seminars
Europe, September 16 - 25 2003

“Tuning DB2 is like flying a Jumbo Jet with a thousand dials and switches”
DB2 Performance in Practice

Agenda

- The tuning challenge
- Memory resource tuning
- Gone fishing
- Tuning SQL
- Index design issues

All screen shots courtesy of T-Systems, Bielefeld, Germany

Increased Complexity
Consequences for DB2

- Added functionality
  - stored procedures, UDF, UDT, LOBs, BLOBs, CLOBs, Extenders
- Connectivity
  - ICLI, DB2 Connect, TCP/IP
- High availability
  - Web Sphere, e-business
- More data
  - Terabyte DBs not unusual
- Related systems
  - CICS, MQ-Series
  - Oracle, SQL-Server
- New applications coming
  - Federated databases
  - XML, XQUERY

Increased demands on DB2 and the DBA!

Why not just buy more hardware ???

NOT ALL PROBLEMS CAN BE SOLVED BY BUYING MORE HARDWARE!
The Tuning Challenge

- More hardware may mask the problem without removing the cause
- Pilots can fly Jumbo Jets, DBAs can tune DB2 (and often related systems too!)
- Know how to identify the problems
- Know how to fix them
- Where do we begin?

Traditional Memory Resource Tuning

- Monitor the system
- Size the various pools (not just the BPs!) and set the ZPARMs
- Tune for the observed workloads, perhaps two sets of ZPARMs
- Monitor the system
- When an exception occurs, fix it
- Monitor the system --- etc.

TIMES THE NUMBER OF DB2 SYSTEMS!

Can we automate this somehow?
Tuning Memory Resources with BMC’s Pool Advisor (*Cool Pool Tool!*)

- Statistics collected every five minutes
- Analysis performed every five minutes
  - based on a fifteen minute rolling average
- Recommends actions for the various Pools
  - BP and EDM, DSC, RID, Sort, GBP
- Text Advisor explains recommendations
- Recommendations can often be carried out automatically, if we wish
- Operates within parameters that we set
  - MIN, MAX, INC, DEC

Modes of Operation

- Increase and decrease pool sizes for best performance within overall storage usage considering MVS paging rates
  - NORMAL mode - all increase recommendations honored
  - BALANCE mode - equal trade-offs between increase and decrease recommendations
  - DECREASE mode - decrease recommendations honored (virtual storage constraints exist)
  - FORCE mode - attempts to reduce storage for non-critical components (virtual storage constraint is serious)
### DB2 Pool Status Monitor

#### DB2 Pools

<table>
<thead>
<tr>
<th>Pool ID</th>
<th>Health</th>
<th>Region</th>
<th>PageRate</th>
<th>BGOPRate</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBP1</td>
<td>GOOD</td>
<td>45M</td>
<td>0</td>
<td>14</td>
<td>100%</td>
</tr>
<tr>
<td>DBP2</td>
<td>GOOD</td>
<td>23M</td>
<td>0</td>
<td>956</td>
<td>100%</td>
</tr>
<tr>
<td>DBP3</td>
<td>GOOD</td>
<td>65M</td>
<td>0</td>
<td>0</td>
<td>100%</td>
</tr>
<tr>
<td>DBP4</td>
<td>GOOD</td>
<td>13M</td>
<td>0</td>
<td>312</td>
<td>100%</td>
</tr>
<tr>
<td>DBP5</td>
<td>GOOD</td>
<td>17M</td>
<td>0</td>
<td>214</td>
<td>100%</td>
</tr>
<tr>
<td>DBP6</td>
<td>GOOD</td>
<td>74M</td>
<td>0</td>
<td>1039</td>
<td>100%</td>
</tr>
<tr>
<td>DBP7</td>
<td>GOOD</td>
<td>49M</td>
<td>0</td>
<td>55</td>
<td>100%</td>
</tr>
<tr>
<td>DBP8</td>
<td>GOOD</td>
<td>39M</td>
<td>0</td>
<td>948</td>
<td>100%</td>
</tr>
<tr>
<td>DBP9</td>
<td>GOOD</td>
<td>71M</td>
<td>0</td>
<td>662</td>
<td>100%</td>
</tr>
<tr>
<td>DBP10</td>
<td>GOOD</td>
<td>39M</td>
<td>0</td>
<td>653</td>
<td>100%</td>
</tr>
</tbody>
</table>

### DB2 Pools

<table>
<thead>
<tr>
<th>Pool ID</th>
<th>Health</th>
<th>Region</th>
<th>PageRate</th>
<th>BGOPRate</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBP1</td>
<td>GOOD</td>
<td>45M</td>
<td>0</td>
<td>14</td>
<td>100%</td>
</tr>
<tr>
<td>DBP2</td>
<td>GOOD</td>
<td>23M</td>
<td>0</td>
<td>956</td>
<td>100%</td>
</tr>
<tr>
<td>DBP3</td>
<td>GOOD</td>
<td>65M</td>
<td>0</td>
<td>0</td>
<td>100%</td>
</tr>
<tr>
<td>DBP4</td>
<td>GOOD</td>
<td>13M</td>
<td>0</td>
<td>312</td>
<td>100%</td>
</tr>
<tr>
<td>DBP5</td>
<td>GOOD</td>
<td>17M</td>
<td>0</td>
<td>214</td>
<td>100%</td>
</tr>
<tr>
<td>DBP6</td>
<td>GOOD</td>
<td>74M</td>
<td>0</td>
<td>1039</td>
<td>100%</td>
</tr>
<tr>
<td>DBP7</td>
<td>GOOD</td>
<td>49M</td>
<td>0</td>
<td>55</td>
<td>100%</td>
</tr>
<tr>
<td>DBP8</td>
<td>GOOD</td>
<td>39M</td>
<td>0</td>
<td>948</td>
<td>100%</td>
</tr>
<tr>
<td>DBP9</td>
<td>GOOD</td>
<td>71M</td>
<td>0</td>
<td>662</td>
<td>100%</td>
</tr>
<tr>
<td>DBP10</td>
<td>GOOD</td>
<td>39M</td>
<td>0</td>
<td>653</td>
<td>100%</td>
</tr>
</tbody>
</table>

Copyright 2003 Steve McDougall - DB2 Performance in Practice
Dynamic Buffer Pool Tuning

DB2:
Buffer Pool: DB1L
Interval start: 2002-08-05 15:25.59
Current duration: 45 minutes

The efficiency rating for this buffer pool is 62%.
For most busy DB2 systems, this number should be quite high. The efficiency for this interval was poor.

Based on a recent analysis of all relevant performance indicators, an INCREASE in the size of this buffer pool should be considered.

We recommend NO CHANGE to the Sequential Steal Threshold.
We recommend the Deferred Write Threshold be INCREASED by 10%
No adverse performance event counts were detected.
To see additional information click here.
To see the list of Action Items, click here.

Dynamic Buffer Pool Tuning

Following is a list of all the pending action items for the following DB2 and storage pool, with the highest priority items listed first:

DB2:
Interval start: 2002-08-05 15:25.59
Current duration: 15 minutes

STATUS: Auto ACTION: RULE
Priority: 7 Advisor: BPM Recmd: INCREASE BP$ 100 PGS

STATUS: Auto ACTION: EXEC
Priority: 7 Advisor: BPM Recmd: SET BP$ DMQT/VDNQT 50%
Dynamic Buffer Pool Tuning

Rule Identifier: BR2  Rule Version: 2.0  DB2: DB41

Rule Description:
A significant IO rate, sequential access is low, virtual pool residency time is less than 30 seconds and system hit ratio is less than 90%.

Execution Mode: **Automatic**  [Click here to set manual mode.]
Times Triggered: 477  [Click here to reset this counter.]
Times Executed: 4  [Click here to reset this counter.]

Buffer Pool Configuration

Strategy: Place page-sets with similar characteristics in the same bufferpools.
Configuration Advisor

- We can examine the current configuration and change it
- Pool Advisor generates the necessary commands to move objects to pools in the new configuration

EDM Pool

- EDM pool efficiency: 100%
- EDM pool size (pages): 30000
- DSC datapace size: 24600
- EDM pool full failures: 0
- DSC datapace full failures: 0
Dynamic Statement Cache

- Pool Advisor reacts on two critical values
  - High Water Mark
  - Aborts caused by no storage

RID Pool

- Pool Advisor reacts on two critical values
  - High Water Mark
  - Aborts caused by no storage
Sort Pool

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of sort merge passes</td>
<td>283</td>
</tr>
<tr>
<td>Average number of workfiles requested per merge pass</td>
<td>2</td>
</tr>
<tr>
<td>Number of inefficient sort merge passes</td>
<td>5</td>
</tr>
<tr>
<td>Average number of workfiles denied per merge pass</td>
<td>15</td>
</tr>
</tbody>
</table>

Group Buffer Pools

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pool Advisor will make recommendations to change</td>
<td></td>
</tr>
<tr>
<td>– Directory to Data Entries Ratio</td>
<td></td>
</tr>
<tr>
<td>– GBP Size</td>
<td></td>
</tr>
<tr>
<td>– Checkpoint interval</td>
<td></td>
</tr>
</tbody>
</table>
Memory Savings

<table>
<thead>
<tr>
<th>DB2 Subsystem</th>
<th>Storage allocation with SAP recommendations</th>
<th>Possible Storage allocation with Pool Advisor</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2G</td>
<td>536 MB</td>
<td>323 MB</td>
</tr>
<tr>
<td>DB2H</td>
<td>357 MB</td>
<td>* 317 MB</td>
</tr>
<tr>
<td>DB2L</td>
<td>635 MB</td>
<td>* 533 MB</td>
</tr>
<tr>
<td>DBM1</td>
<td>261 MB</td>
<td>* 100 MB</td>
</tr>
<tr>
<td>DBM2</td>
<td>164 MB</td>
<td>* 100 MB</td>
</tr>
<tr>
<td>DB2W</td>
<td>371 MB</td>
<td>* 200 MB</td>
</tr>
<tr>
<td>DBF1</td>
<td>513 MB</td>
<td>* 250 MB</td>
</tr>
<tr>
<td>DBF2</td>
<td>162 MB</td>
<td>* 100 MB</td>
</tr>
<tr>
<td>DB2Q</td>
<td>423 MB</td>
<td>* 250 MB</td>
</tr>
<tr>
<td>DB2U</td>
<td>451 MB</td>
<td>* 250 MB</td>
</tr>
<tr>
<td>DB2Z</td>
<td>430 MB</td>
<td>* 250 MB</td>
</tr>
<tr>
<td>DBS1</td>
<td>1103 MB</td>
<td>** 920 MB</td>
</tr>
<tr>
<td>DBS2</td>
<td>1103 MB</td>
<td>** 950 MB</td>
</tr>
<tr>
<td>DBX1</td>
<td>536 MB</td>
<td>391 MB</td>
</tr>
<tr>
<td>Total Storage</td>
<td>6985 MB</td>
<td>4934 MB</td>
</tr>
</tbody>
</table>

SQL Everywhere

Not always able to check SQL with EXPLAIN
Gone Fishing

- Elapsed Time
- Getpages
- CPU

Why is this inadequate?

Fishing for SQL

- Traditional SQL fishing
  - Accounting report plus Explain can miss SQL which executes thousands of times - but poorly
  - Brings up too many SQLs that are OK
- Better SQL fishing
  - Tune the heavy hitters
  - Tune SQL which accesses the heaviest used objects
Tune the Heavy Hitters

- Find the big fish but consider
  - Many small fish = One big fish
- Saving 1/100's of a second on frequently executed SQL helps
  - cumulative effect
- Deals with SQL that gets through the traditional net
  - i.e. those with low elapsed time, low getpages and low CPU time - but high execution rate
- Identification problem

Using BMC’s APPTUNE
Tune the Heavy Hitters

- At first ignored (low elapsed / IN-SQL Time)
- Executes millions of times
- Worth tuning

<table>
<thead>
<tr>
<th></th>
<th>Before</th>
<th>After</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg IN-SQL per row fetched</td>
<td>0,00045</td>
<td>0,00023</td>
</tr>
<tr>
<td>Avg CPU per row fetched</td>
<td>0,00023</td>
<td>0,00017</td>
</tr>
<tr>
<td>Avg GETPAGES per row fetched</td>
<td>966</td>
<td>189</td>
</tr>
</tbody>
</table>

Heaviest Used Objects

- Find the top N busiest objects in the system
  - number of getpages in a given time period
- Look at SQLs which access these objects
  - elapsed time, getpages, CPU
- Fewer SQLs to look at
  - better chance of finding tuning candidates
Heaviest Used Objects

<table>
<thead>
<tr>
<th>Database Page Set Type</th>
<th>BPool</th>
<th>1555</th>
<th>1556</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Creator Object | Type | BPool | HRatio | Number | % Tot Time | Avg Time |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>APR3 HRP1001</td>
<td>T</td>
<td>BP2</td>
<td>99.9%</td>
<td>464017</td>
<td>97.6%</td>
<td>09.2189</td>
</tr>
<tr>
<td>APR3 HRP1000</td>
<td>T</td>
<td>BP2</td>
<td>99.6%</td>
<td>11379</td>
<td>2.4%</td>
<td>00.3708</td>
</tr>
<tr>
<td>APR3 HRP1001</td>
<td>T</td>
<td>BP2</td>
<td>81.4%</td>
<td>00.0%</td>
<td>00.0%</td>
<td>00.00</td>
</tr>
</tbody>
</table>

Stmt Call Sect BPool | Getpage | 1555 | 1556 |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Easy to read!
Statistics Recommendations

Sample

BMC184240S-RUNSTATS have not been run on this object. The Optimizer will use defaults to determine the access paths.
BMC184264W-The HIGH2KEY and LOW2KEY values are not available. This can adversely impact performance. Run RUNSTATS and rebind the object.
BMC184280W-This index is the Clustering index but it is no longer clustered. Consider running a REORG and RUNSTATS, followed by a REBIND.
BMC184290W-The average Index Leaf Distribution exceeds the installation standard. The Leaf Distribution/100 is ...
BMC184260W-The ratio of Full Key Cardinality to First Key Cardinality is almost equal. This index is not a good candidate for the Clustering index. Consider changing the CLUSTERING option to another index that will provide some degree of grouping.

Access Path Recommendations

Sample

BMC184594I-DB2 will be using a tablespace scan to satisfy this query which has been designated for ONLINE Only usage. The table does not currently have an index. The number of rows in the table is ...
BMC184596W-DB2 will be using multiple indexes to satisfy this query which has been designated for BATCH Only usage. The use of multiple indexes to satisfy this query may cause excessive overhead. The number of rows in the table is ...
BMC184600S-DB2 will be using a Hybrid Join to satisfy this query which has been designated for ONLINE Only usage. Hybrid Joins are complex to resolve and may cause excessive overhead. The number of rows in the table is ...
SQL Recommendations

Sample

BMC184000S-SELECT * on TABLE is not recommended since the answer set may change in the future.

BMC184326W-Tip: Consider using OPTIMIZE FOR N ROWS if the desired answer set is small and you want to avoid the cost of List Prefetch.

BMC184044I-A NOT BETWEEN Value1 and Value2 predicate is Non-indexable. Consider redesigning the predicate to use OR instead of NOT BETWEEN.

BMC184314W-The ORDER BY contains a column(s) which is not indexed. Drop the column from the ORDER BY or add it to the index to eliminate sorting.

Locking Recommendations

Sample

BMC184186W-The object was bound with the ISOLATION option of Repeatable Read. Review with DBA on this usage.

BMC184376W-An INTENT EXCLUSIVE lock will be acquired on a table accessed via a Table Space Scan. Consider using LOCK TABLE to minimize locks and DB2/IRLM Lock management overhead. This can reduce CPU usage and improve elapse time.

BMC184582W-To improve concurrency and throughput consider modifying the SQL text to use WITH UR or bind the object with a Cursor Isolation=UR (Uncommitted Read).
Index Design Issues

- Performance indexes
  - to avoid TS scans on large tables
  - to avoid sorts
  - index only access (not with VARCHAR till V8)
  - help joins

- Common Problems
  - Clustering on primary key
  - Low cardinality column first
  - Column ordering
  - unused and redundant indexes

Index Design Example

```
SELECT * FROM "TEVEN"
WHERE "MANDT" = ?
AND "PERNR" = ?
AND "LDATE" >= ?
AND "LDATE" <= ?
AND "STOKZ" = ?
ORDER BY "MANDT", "PERNR", "LDATE", "LTIME", "ERDAT", "ERTIM"
FOR FETCH ONLY WITH UR
```

- INDEX COLS
  - MANDT
  - PERNR
  - LDATE
  - LTIME

Elapsed 0,269

- ADD COLS
  - ERDAT
  - ERTIM
  - PDSNR

Elapsed 0,039

- Make IX clustering
- Original clustering was on the primary key
Index Advisor Component

- NEW
- Tables by getpage volume
- Table CRUD matrix
- Statement column usage
- What if analysis

DB2 Performance in Practice Summary

- Increasingly complex applications are driving tuning requirements
  - more hardware is not always the best answer
- Automation of DB2 memory resource tuning
  - smart memory tuning - now
- Superior strategies to find SQL to tune
  - heavy hitters
  - heaviest used objects
- Recommendations to improve SQL / access path
- Index design issues and common problems
Even Jumbo Jets have auto-pilots!

DB2 Performance in Practice

Steve McDougall
Independent Database Consultant
SteveMcDou@aol.com