21st Century Data Management Technology
Jim Dee, Corporate Architect, BMC
June 7, 2016

BMC Day Israel
Where IT goes Digital
Transforming Mainframes for Digital Business

Next Generation Technology for DB2

MainView Adaptive Monitoring

MLC Cost Management

Data

Systems

Cost
DB2 Then and Now – Trees vs. A Forest

1980s

- **Database Size**
  - A few Gigabytes
  - ~x10,000

- **Table Size**
  - A few million rows
  - ~x10,000

- **Number of Objects**
  - Low Hundreds
  - ~x1000

- **Transactions per second**
  - Under 100
  - ~x200

Now...

- **Database Size**
  - 10’s to 100’s of Terabytes

- **Table Size**
  - Tens of billions of rows

- **Number of Objects**
  - Tens of Thousands, SAP 100K

- **Transactions per second**
  - 10,000 – 20,000
Traditional Utility Architecture

DB2 Reorg in the 1980’s
- Unload
- SORT
- Load
- Rebuild Indexes

DB2 Reorg Now
- Unload
- Decompress
- SORT
- Recompress
- Load
- Iterative Log Apply
- Rebuild Indexes
Digital Tsunami Wreaks Havoc on Traditional Data Management Methods

• Disappearance of DB Maintenance Windows
  • "Online" tools disrupt Apps
• DBs too large to handle
  • Traditional tools take too long or can’t run at all
  • Require 3X+ storage to run
• More DBs to manage
  • Impossible to juggle the increasing number of exception conditions
• DBAs managing more DBMSs
  • Traditional tools require expert knowledge and care
How IT Copes Now...

Skipping regular DB maintenance
Maintain time consuming scripts that only schedule partial maintenance
Fire-fighting maintenance when App SLAs breached
Apps timeout during “Online” maintenance
Degrade App performance while they drop indexes to complete maintenance
Increase processing capacity to maintain service levels
Schedule disruptive outages
So What....We Got it Covered

**Costs...**

- Application outages
- Slower applications
- Stretched or Missed SLAs
- Customer satisfaction decreases
- Lost customers
- Reduced DBA productivity
- DBA firefighting mode
- Increased MLC costs
- Increased hardware costs
- Data integrity issues
- Lost Data
Managing the trees.....

Traditional DB maintenance methods increasingly require each object to be handled individually
Having to manage individual trees means you don’t have time to look after the whole forest
Customer Case

2 Billion Row Tablespace – 3NPIs

FAILED – Insufficient SORT Space

Response – DROP 2 NPIs (IMPACT APPLICATIONS!)
Customer Case

2 Billion Row Tablespace – 3NPIs

Failed!

Failed – Insufficient SORT Space

Response – Add 24 SORT Volumes (EXCESS RESOURCES!)
Customer Case

2 Billion Row Tablespace – 3NPIs

Failed!

Minutes

0 50 100 150 200 250 300 350 400

DB2 Reorg

FAILED – Insufficient SORT Space

• Indexes were dropped impacting application performance
• Added SORT Space using excess resources
• Wasted hours of CPU and people time
• All managing one tablespace!
Customer Case

2 Billion Row Tablespace – 3NPIs

**Traditional Methods - Focus on the individual Object**

- Manual Process – 3 DBAs!
- Drop Indexes – *Slowed Applications*
- Add SORT Work Space – *Excess Resources*
- **Object left disorganized**
- 12 hours wasted

**BMC - Manage the Database**

- Submitted a simple job - *success*
- Celebration!
Managing the forest.....

Traditional methods require each object to be handled individually leaving many unmanaged

NGT analyzes and automatically optimizes DBs in real time with Zero-Outage

Customer quote- “We were only effectively managing 20% of our data before NGT.”
## BMC’s NGT Suite

### Next Generation Technology

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>NGT REORG</td>
</tr>
<tr>
<td>NGT LOAD</td>
</tr>
<tr>
<td>NGT UNLOAD</td>
</tr>
<tr>
<td>NGT CHECK</td>
</tr>
<tr>
<td>NGT COPY</td>
</tr>
<tr>
<td>NGT RECOVER</td>
</tr>
<tr>
<td>NGT STATS</td>
</tr>
<tr>
<td>NGT UTILITY MANAGER</td>
</tr>
<tr>
<td>NGT LOBMaster</td>
</tr>
</tbody>
</table>
BMC Next Generation Technology

Architected for digital business

**Simple**
- Easy to implement with wildcarded minimal JCL (where appropriate) that is never generated
- Structured and unstructured data management
- Decisions made at time of utility execution

**Intelligent**
- Centralized business policies
- Analysis and automation in real time

**Powerful**
- Zero perceived outage for applications
- Fast and efficient processing
- Parallel Processing leveraging the sysplex
The New Paradigm – Next Generation Technology Utilities

Keeps Your Digital Apps Up and Running at their Peak Efficiency

NGT analyzes DBs in real time

Automatically optimizes all objects in need with Zero-Outage

No human intervention

DBs stay optimized keeping Apps Up and Costs Down
Zero Perceived Outage

• ZERO DOWNTIME! Drain is gained quickly and held for a very short time
  - No application timeouts (-911’s)
• Patented design
  - Never read the LOG
  - Not bothered by traffic around DB2
• Customer Example
  - Previous solution
    • 1000 timeouts every week
  - NGT solutions
    • Reorganizing 6 times as often
    • Eliminated timeouts
Typical Reorg Method

1. Unload/Decompress (double its size)  2
2. Sort  \( \sim n \log(n) \)  (6 – 9 passes)  \( \times 7 \)
3. Reload/Recompress  (half its size)  \( +2 \)  
   \[ \text{16 passes} \]

Plus the Index Build which is significant
And RISKY!!;
NGT Reorg Method

1. Unload/Decompress (double its size) 2
2. Sort $\sim \log(n)$ (6 – 9 passes) x7
3. Reload/Recompress (half its size) x2

Read in – Write out clustered

Plus the Index Build which is significant And RISKY!!;
Fast and Efficient Processing

Disorganized vs Random data
  – Use the structure of DB2 to our advantage

Tablespace Data is never sorted
  – No external sort called ever
  – Data is clustered

New algorithm is dynamically created for each Reorg
  – Patented process reducing/eliminating degradation

Means no Decompression/No Compression
  – Read and write data once!

Discard in online reorg
  – Full subselect capabilities
Fast and Efficient
Performance that changes the game

Times in Minutes
Incredible performance enables more reorgs more often to ensure optimal application performance!

UTS – PBG – 4 Parts 201M Rows, 45G TS 3NPIs (11.1G, 5G, 4G)
Parallel Processing Leveraging the Sysplex

Parallel Processing
Automatic
Aligned with z System growth
Future Proof Technology
Limit parallelism today
Expand for tomorrow

1 statement can process thousands of objects across a sysplex – leveraging mainframe assets
The only answer to DB2’s growing complexity is simplification

```
//PAYRRG01   JOB ...
//JOBLIB      DD DISP=SHR, DSN=BMC.NGT.LOAD
//NGTUSTEP   EXEC PGM=CDBUTIL, PARM='TST8,,RESTART'
//SYSPRINT   DD SYSOUT=*  
//UTPRINT     DD SYSOUT=*  
//CDBEXEC     DD DISP=SHR, DSN=NGT.AUTO.BMCEXEC
//SYSIN       DD *

REORG TABLESPACE PAYROL%.
```

Restart Parameter

- RESTART
- NORESTART
- QUICKEXIT
Simple Index Management

Indexes are ALWAYS Reorganized During Tablespace Reorg – Never rebuilt or updated!

Reorg of 2 Non partitioning indexes
125G each

Impact of Reorging Indexes

Focus your reorgs on improving application performance!
Traditional Unstructured Data Management

- **CHECK IX**
  - Verify if the LOB Index is valid

- **CHECK LOB**
  - Verify the internal structure of the LOB

- **CHECK DATA**
  - Verify that the ID fields in the base TS are in the LOB Index

- **REORG**
  - Separate from base tablespace Reorg
  - Requires unload of the LOB data
  - Utility JOB JCL change

Manual 4-step process
Most users find this approach impractical and decide to forgo all integrity checking except when a problem is detected (IF it is detected)

“You better get used to them!” IBM - Francis Desiron GSE 2013
Simple Unstructured Data Management

A single command performs all integrity checks and reorganizes

“Free” Integrity checking (It does not increase Reorg time)

Online 100% of the time

Simplification even with complex unstructured data
Customer Example

Unstructured Data

- Government Institution
  - LOBs since v8
  - IBM Check failed after hours of execution
  - Undiagnosed problem for > year

- Legacy LOBs
  - Diagnosed problem in 7 minutes
  - Identified Base Table structure problems
  - Identified Missing Lobs

- NGT Solution

Customer waited too long!
Object was unrecoverable! DATA LOSS
## Customer Example - Large Pharmacy

### Customer Situation
- Scheduled monthly weekend outage to reorganize databases
- Largest objects remained unreorged for more than 4 years!
- Were having to budget for increased storage capacity

### Why BMC?
- Ease of conversion
- 60% savings in existing DASD
- Eliminated IBM Automation as an expense (never could implement)
- Optimized reorganization process

### Benefits
- Eliminated monthly outage
- Reduced utility CPU usage by over 60%
- All objects reorganized as part of automatic process

© Copyright 2016 BMC Software, Inc.
The Right Tool for the Job
BMC Database Management with Next Generation Technology

• Manage and optimize the bigger, faster, always available data demands of today
• Be ready for the increasing demands of tomorrow
• Automate and enforce business policies
• Keep your apps up and running at their peak performance

Simple
Intelligent
Powerful
Customer Benchmarks
REORG TS – SEG SIZE 32, 2 Indexes

64% less CPU

53% less elapsed

© Copyright 2016 BMC Software, Inc.
REORG TS – 16 Partitions, 3 Indexes

REORG TABLESPACE

- CPU: 22% less
- Elapsed: 58% less elapsed

IBM 1
IBM 2
NGT 1
NGT 2
REORG TS – 1 Partition of 20, 1 Index

REORG TABLESPACE

- **CPU**
  - IBM 1: 700 seconds
  - IBM 2: 1000 seconds
  - NGT 1: 500 seconds (22% less)
  - NGT 2: 1000 seconds

- **Elapsed**
  - IBM 1: 2000 seconds
  - IBM 2: 2300 seconds
  - NGT 1: 900 seconds (58% less elapsed)
  - NGT 2: 2300 seconds
Thank You

Bring IT to Life.™