Agenda

• Explore various ways Rdb buffers data
• Examine pros and cons of buffering features
Test Case

• Multi-user test
  • Rdb V7.1-401
  • Modified MF_PERSONNEL database
  • Three users reading, inserting, deleting in EMPLOYEES table
  • Short Transactions
  • Results varied a bit from one run to the next

• Standalone ES40 with HSG80
  • VMS V7.3-2
  • No other users
  • Not clustered
Effects of Number of Buffers

- More buffers increases the chance that recently accessed pages will still be in memory
- More buffers require more overhead to manage
- More pages in memory increase chances of lock conflicts and deadlocks
Local Buffers With XFC vs Global Buffers Without XFC

• By sharing a common buffer pool many processes can take advantage of the I/O done by a single process
• Keeping unreferenced data in memory saves I/O if referenced again
• Rdb global buffers and XFC have similar goals
Effects of Number of Buffers
I/O for Local vs Global Buffers

![Graph showing the relationship between the number of buffers and I/Os for local and global buffers. The graph demonstrates a significant decrease in I/Os as the number of buffers increases, with local buffers showing a more pronounced decline.]
Effects of Number of Buffers

CPU Time

- LB XFC CPU Seconds
- GB CPU Seconds

Buffers vs. CPU Seconds
Effects of Number of Buffers
Elapsed Time

Bufflers

Elapsed Seconds

LB XFC Elapsed Seconds

GB Elapsed Seconds
Effects of Different Features
Local / Global Buffers, XFC

CPU Seconds

Elapsed Seconds

320 Buffers
The Quest for More Buffers

- Shared Memory is System – takes virtual address space from system (S0) address space
- Global Buffers in VLM (“large memory is enabled”) – puts global buffers in Rdb’s implementation of extended memory
- Get buffers out of P0 address space – reduce VASFULL errors
- Makes buffers memory resident – less page faulting
Effects of Different Features
SMS, GB VLM

320 Buffers
Effects of Different Features

Fast Commit

• Don’t write modified pages to disk at the end of every transaction
• If a failure occurs before modified pages from a committed transaction are written to disk then those pages are rebuilt from the journal
• Can increase page lock conflicts
• Typically reduces/eliminates RUJ writes
Effects of Different Features

OPT

• Ownership of “hot” pages that are often modified by many processes are passed between processes without first being written to disk
Effects of Different Features
Fast Commit and OPT

Data Writes

Elapsed Seconds

320 Buffers
Effects of Different Features
Row Cache

- Row cache copies referenced rows from disk data pages into an in-memory cache
- Once a cache is loaded little read I/O is needed
- With “snapshots in cache” enabled many writes can be avoided
- Reduces page lock contention
- Requires more expertise to setup
Effects of Different Features
Row Cache

Elapsed Seconds

320 Buffers
Effects of Contention
Lock Conflicts and Deadlocks

• If there is a lock conflict on a table or row lock then all buffers are flushed and demoted

• VMS waits DEADLOCK_WAIT seconds before checking for deadlocks
After-Image Journaling
(AIJ Writes)

• Always enable ALS
  • Does double buffering
  • Less AIJ lock contention
  • Better tuned for group writes
Before-Image Journaling (RUJ Writes)

- RUJ buffers must be flushed before data buffers are flushed – synchronous write
- Lock conflicts cause data buffers (and thus RUJ buffers) to be flushed
- If Fast Commit is enabled then RUJ flushes are often avoided
- Larger ABW batches allow larger RUJ batches and thus fewer RUJ writes
Summary

• XFC works great! May be better than global buffers
• XFC and global buffers are redundant
• Don’t skimp on buffers
• Fast commit can significantly reduce writes
• Row cache prevents I/O and lock conflicts
• Beware of contention
• Always enable ALS
• Experiment to see what works best for you
For More Information

- [www.oracle.com/rdb](http://www.oracle.com/rdb)
- [metalink.oracle.com](http://metalink.oracle.com)
- [www.hp.com/products/openvms](http://www.hp.com/products/openvms)
- [Magnus.Weiman@oracle.com](mailto:Magnus.Weiman@oracle.com)
- [Paul.Mead@oracle.com](mailto:Paul.Mead@oracle.com)
QUESTIONS & ANSWERS