

# Do Extent Numbers Really Matter Any More?

par Steve Thomas *BMC Software*



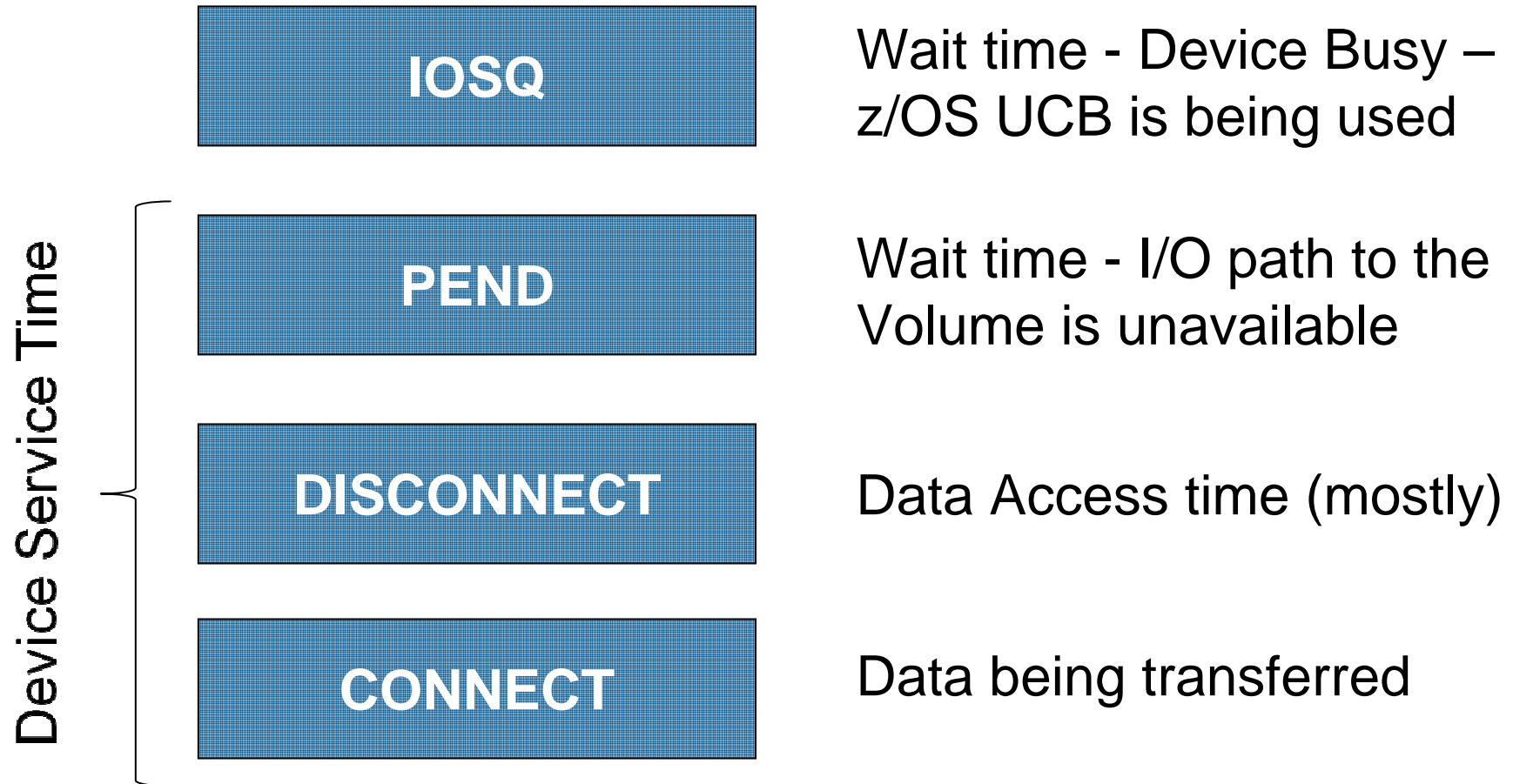
**GUIDE Share France**  
Une Association Indépendante d'Utilisateurs IBM

Réunion du Guide DB2 pour z/OS France  
Vendredi 21 novembre 2008  
Tour Manhattan BMC, Paris-La Défense

# Topics

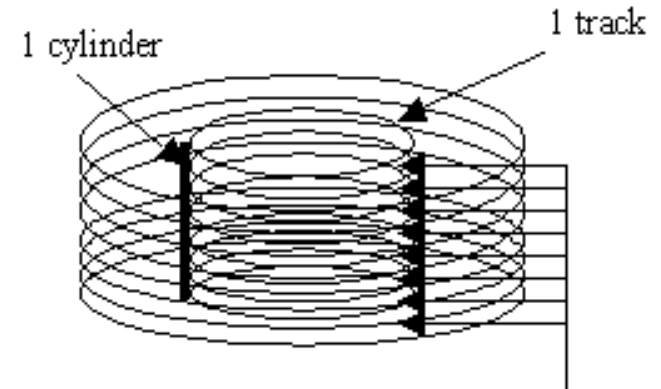
- Introduction and background information
- Advances in disk technology which impact DB2
- Some Useful Operating System changes
- Mirrors, Flashcopy and GDPS
- “*Do Extent Numbers matter?*” and other conundrums

# Elements of I/O time



# Traditional IBM DASD

	3380-A04	3390-A14
First Introduced	1980	1989
Bytes per Track	47,476	56,664
Tracks per Cylinder	15	15
Volume Capacity	630Mb	946Mb
Rotation Time	16.6ms	14.2ms
Seek Time	2-30ms	1.5-18ms
Data Transfer Rate	3.0Mb/sec	4.2Mb/sec



# Why is this important?

- We still emulate 3390 track Architectures today

3390 Model Type	Cylinders	Capacity
Model 1	1,113	946Mb
Model 3	3,339	2.83Gb
Model 9	10,107	8.51Gb
Model 27	32,760	27.84Gb
Model 54	65,520	55.68Gb

- Current restrictions:
  - 64K volumes per Sysplex
  - 65,520 cylinders/volume (z/OS 1.10 increases this to 262,668)
  - 127 extents per dataset per volume
  - 59 Volumes per dataset

# Modern Disk Storage Arrays

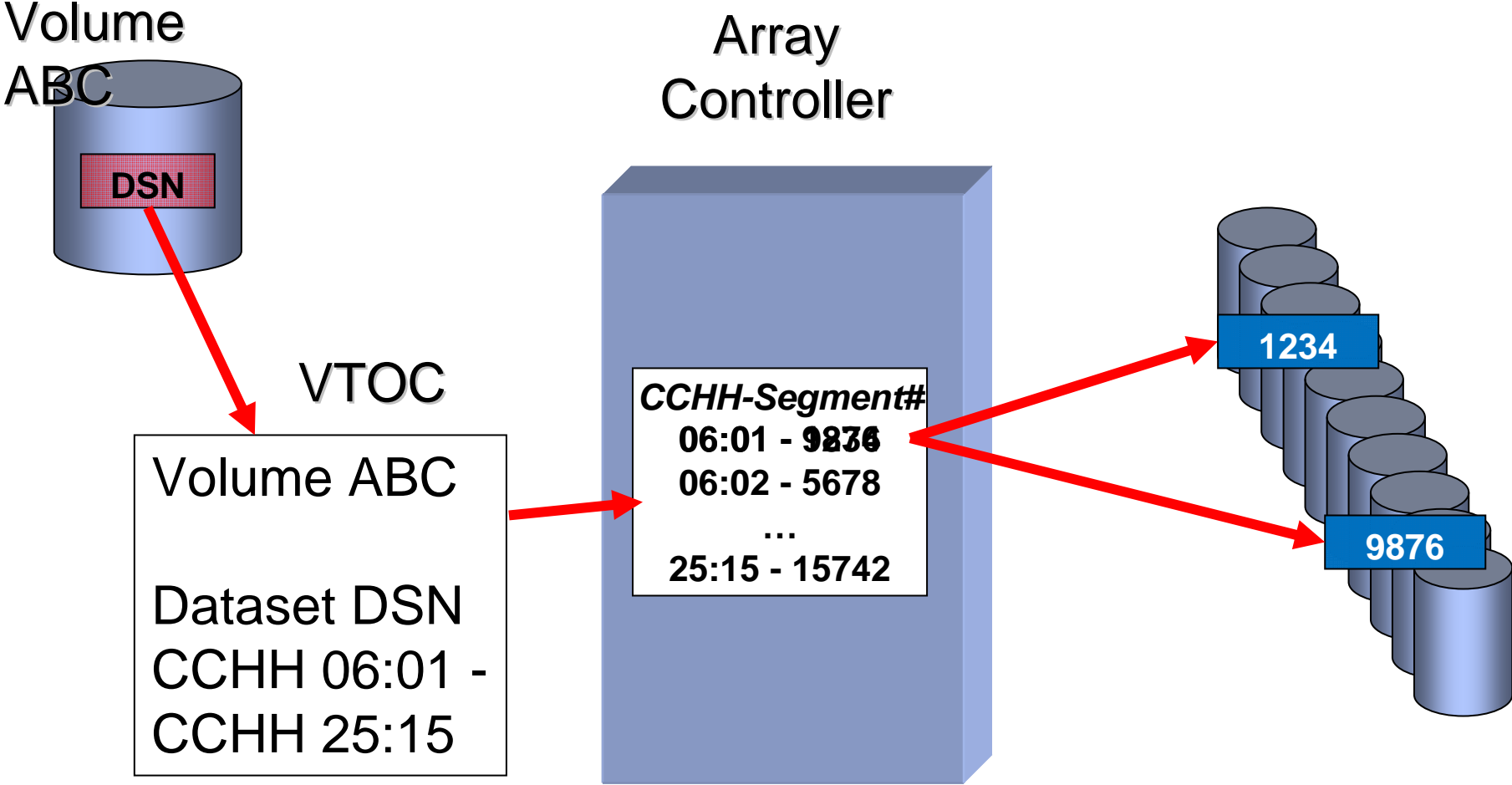
- Precise details differ but overall concepts are similar
  - All based on RAID Technology
  - Emulate 3390 Tracks with arrays of 'small' disks
  - Potentially huge capacity – over 500Tb
  - Prefetch data using a large Cache
  - Do not update data in place but write it to a new location
- Examples include:
  - IBM RAMAC Virtual Array or RVA
  - IBM Enterprise Storage Server or ESS
  - IBM TotalStorage DS8300 (pictured)
  - EMC Symmetrix Family
  - HDS 7700 and 9900



# Mapping Logical to Physical

- z/OS refers to a track by an address, **CCHH**
  - **CC** is the Cylinder Number
  - **HH** is the Track or Head Number
- On a 3380 & 3390 disks this refers to a physical track
- In an Array, Tables in the Disk Controller map the address to a physical location
  - A set of sectors on one or more of the underlying disks
- Updated tracks are written to a new location
  - Disk Controller simply updates the Control Table Data
  - Update will be initially saved in the Non-Volatile Cache
  - The old data still exists until the space is released or reused
  - This is easier to explain using a diagram...

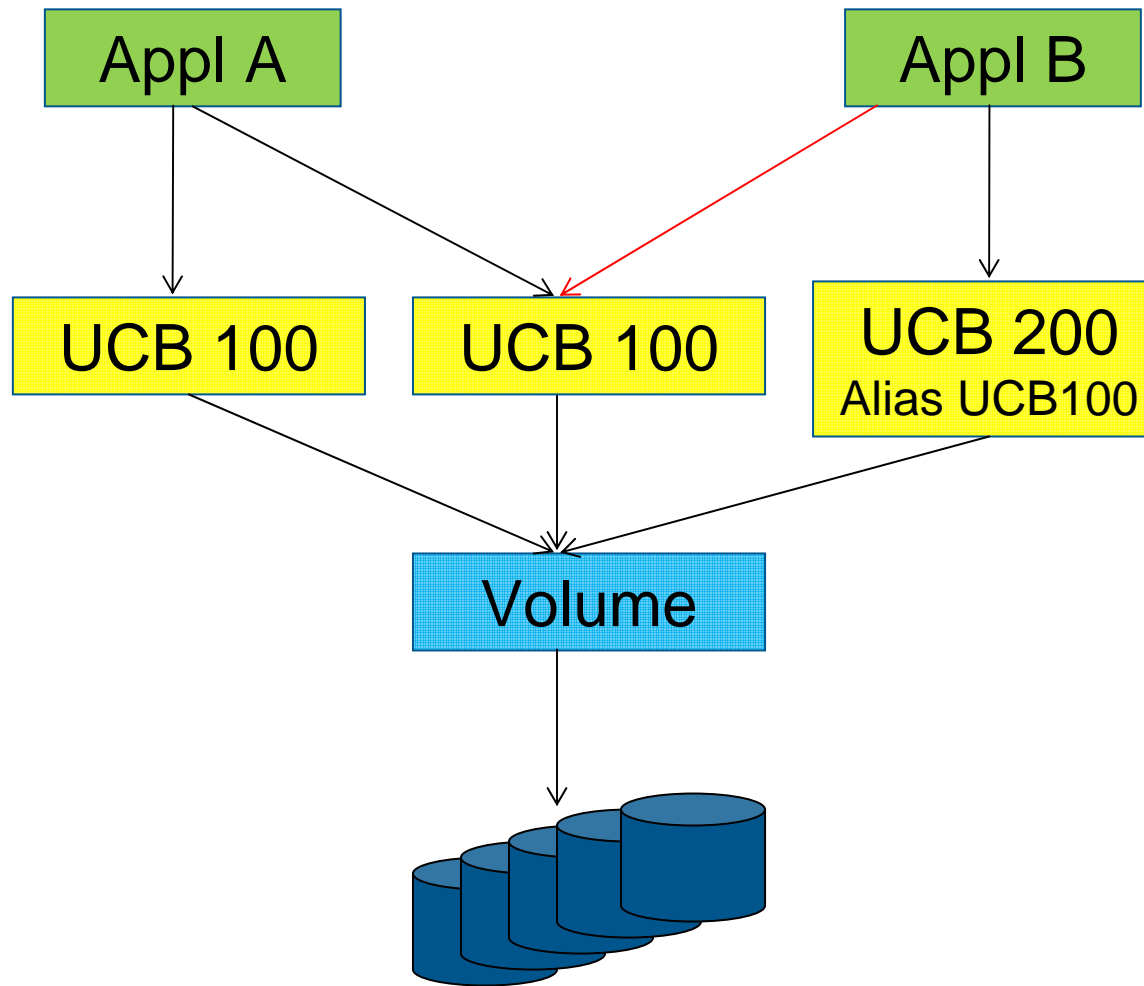
# How a track update occurs



# Disk Controller Data Cache

- Controller Caches act as a buffer to the physical disks
  - Data prefetched from Disk into the Cache before it is needed
  - Up to 256Gb on a current DS8300
- Cache provides significant performance benefit
  - Typically <1ms for Cache hit but 5-10ms for a random read
- DB2 can mark I/Os to disable pre-fetch
  - This request was ignored by RVA and ESS (Shark) devices
  - Managed by DSNZPARM SEQCACH=BYPASS/SEQ
    - Note SEQCACH is treated differently by DS8300 – see later slide
- IBM DS8300 SARC technology improves intelligence
  - Sequential Prefetching in Adaptive Replacement Cache
  - Anticipates what data you will require and pre-reads it
  - Learns from historical trends hence the term adaptive

# Parallel Access Volumes



# Dynamic vs Static PAV

- Static PAV defined by Storage Administrator
  - Each PAV is defined manually
  - How do you decide where aliases are required?
  - Changing Workloads can cause serious complications
  - Need to remember to stay within the limit of 64K UCBs
- Dynamic PAV defined by WLM based on Workload
  - Pooled by Logical Control Unit
  - Competition for PAVs if they share LCUs
  - Sysplex co-ordination can mean relatively high WLM overhead
- Whichever you use you are unlikely to eliminate IOSQ
  - Requests for the same dataset extent will still Queue
  - Larger sites may well have restrictions on UCB numbers

# HyperPAV

- Extension of PAV concept
  - Introduced by z/OS 1.8 with suitable Hardware microcode levels
  - Can also be retrofitted to z/OS 1.6. and 1.7 via PTF
- UCB aliases allocated for the duration of one I/O
  - Effectively eliminates UCB address limitations
- Each alias can be used concurrently by different systems
- No requirement for Sysplex co-ordination
  - Provides benefits of Dynamic PAV without WLM overhead
- HyperPAV (together with Multiple Allegiance) make emulating larger 3390 models much more practical as most volume contention is eliminated

# Multiple Allegiance

- The Control Unit places a reserve or lock on the volume when an I/O starts
  - Avoids inter-system I/O conflicts but introduces delays
  - These show up as PEND time in I/O reports
- PAV doesn't help as conflict is between LPARS
  - They only operate within a single z/OS Image or Sysplex
- Multiple Allegiance helps resolve this problem
  - Provided by the Control Unit
  - Allows Concurrent I/Os from different LPARS and guarantees that there will be no extent conflict
- Determination is made by the Control Unit
  - Compatible I/Os are allowed to complete without delay
  - Incompatible I/Os are queued within the Control Unit

# Extended Format datasets

- 32 bytes added to each physical CI
  - 4K DB2 page becomes 4,128 bytes
  - Applications such as DB2 do not see this extra space
  - You still get 12 4K DB2 pages to a Track
- Benefits include:
  - Improved Reliability of Channel operations
  - Enables DFSMS Striping
  - Enables Extended Addressability (DSSIZE>4Gb)
  - Used by Extended PDS libraries (PDSE)
  - Permit up to 123 extents per dataset per volume
- Used to be performance penalty for FICON Channels
  - Reported as over 50% in 2005 using DS8000 disks

# MIDAW

- Technical improvement to the Channel Instruction
  - Highly technical and outside scope of this presentation
  - See excellent IBM Red Paper in references for details
- Essentially Allows Media Manager to fully exploit the Track Level command operations of z/Architecture
  - Reduces number of Control Words required for an I/O
  - For an EF dataset reduces from 24 Control Words to 1
- Using MIDAW reduces EF dataset performance penalty
  - Requires z9 processor or above
  - Requires z/OS 1.7 or (retrofitted to 1.6 with APAR)
  - Other than that there's nothing you need to do

# DFSMS Striping

- Spreads datasets across a number of Control Units
  - Each stripe contains the same amount of space
- Striped VSAM must be Extended Format Datasets
  - CIs are distributed across the stripes
- Much faster for Sequential I/O
- Recommend that Active Logs are striped
- Also consider for other primarily Sequential Data
  - For example Work Files, DPSIs, LOBS
- Potential to stripe Disk based Sequential Files
  - Worth considering for Utility Work files, Imagecopies etc.
  - DB2 9 allows disk based archive datasets to be striped

# Increased Extent Numbers

- Introduced in z/OS 1.7
- Limit is now 123 extents per volume x 59 volumes
  - Both these numbers are architectural limitations
  - Theoretical maximum is now 7,257 extents
- Only works with DFSMS managed STOGROUPs
  - But these do NOT need to be Extended Format
- Requires a DFSMS Data Class definition change
  - Extent Constraint Removal=YES

# Sliding Extents

- DB2 V8 can allocate secondary extents automatically
  - Set SECQTY to -1 and all implicit tablespaces use this
- Enable by setting MGEEXTSZ=YES
  - Default is NO for V8
  - Changed to YES automatically when you upgrade to DB2 9
- Extent sizes allocated gradually increase
  - First 127 extents are allocated using increasing sizes
  - First extend request uses 2 cylinders
  - Each subsequent Extent is 1 cylinder larger than the previous
- Maximum Extent size is based on DSSIZE
  - Up to 16Gb the largest secondary extent size is 127Cylinders
  - 32Gb & 64Gb the largest secondary size is 559 Cylinders

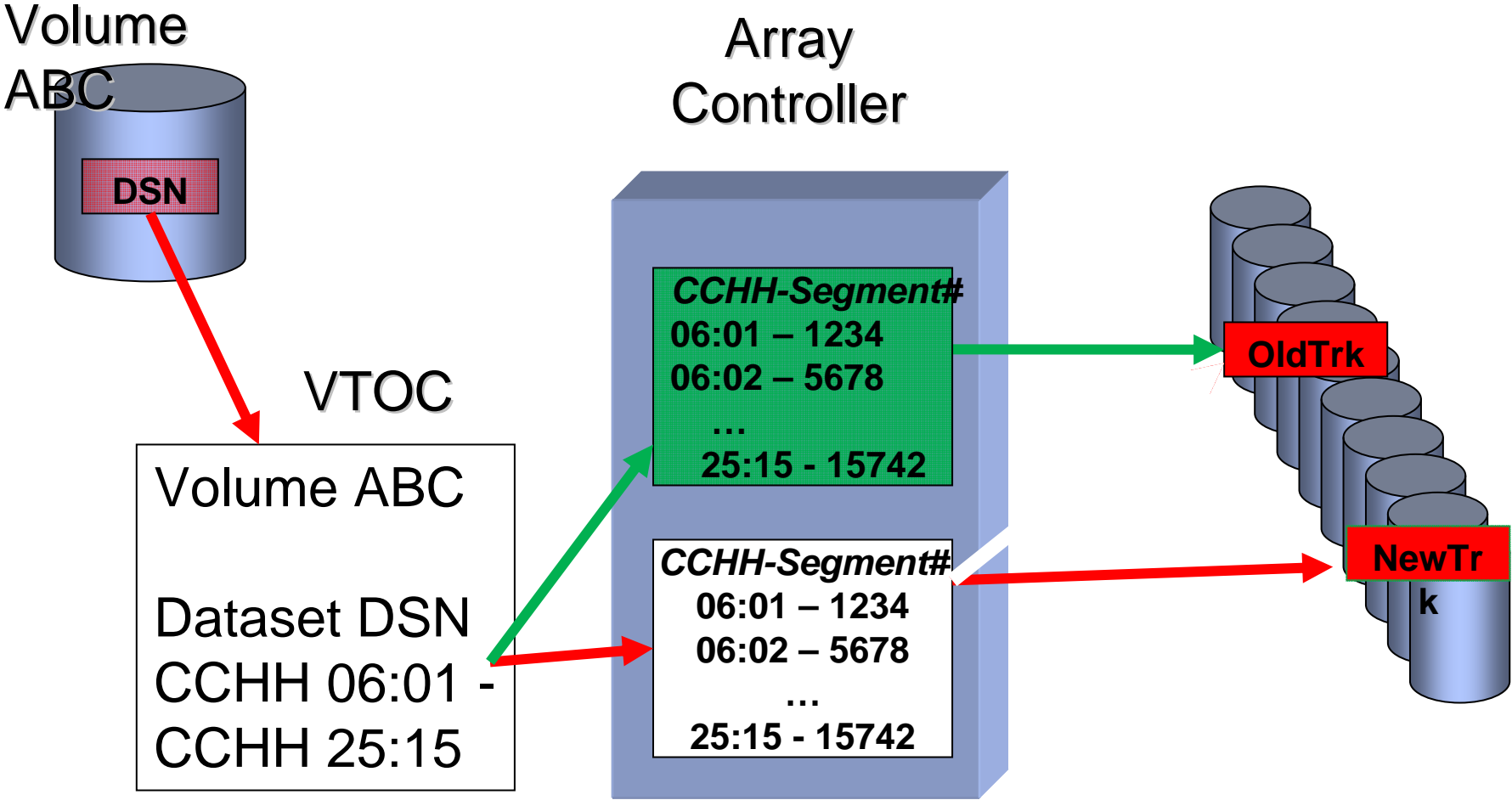
# More extent changes

- Extent Consolidation
  - Introduced in z/OS 1.5
  - If new extent is adjacent to old, they will be merged
  - Extents may end up bigger than PRIQTY or SECQTY
  - Requires DFSMS managed STOGROUPs
  - Beware REORG with manual allocations and multiple extents!
  - Better results if the volume is not fragmented
- How DB2 allocates 32Kb pages
  - Behaviour changes with DB2 V8 when you use DSVCI=YES
  - 32Kb pages allocated as 2x16Kb rather than 1x32Kb
  - 3 blocks per track = 45 per Cylinder
  - 22 Pages per Cylinder with 1 block left empty
  - This is done to stop a Page spanning across 2 Cylinders

# Flashcopy and equivalents

- Exploit opportunity provided by Control Unit technology
  - Tables now map z/OS Datasets to real underlying Disks
  - Snapping methods simply copy these pointers
- Name and exact method varies between Disk vendors
  - IBM – Flashcopy
  - HDS – Nanocopy but now also support Flashcopy
  - EMC – Timefinder
- Requirement for Consistency generally requires outage
  - Although even this is now beginning to go away

# Logical view of Flashcopy



# What can you do with it?

- Primary value is to provide very fast copy
  - Perhaps 1-2 seconds per dataset or volume
  - Largely independent of how large the object is
- This can still take some time for large systems
  - 40,000+ datasets in SAP anyone?
  - Volume level operations can be better for these
- Many utilities can now drive the process for you
  - IBM and ISV Imagecopy utilities
  - IBM BACKUP SYSTEM
- Remember back-end space will eventually be needed
  - When depends on Disk Vendor's methodology
- Can you copy the copy to release the space?

# Mirrors

- Two primary methods are available
- Synchronous Mirrors (PPRC or Metro Mirror)
  - Managed by Disk Controllers
  - I/O is not flagged as complete until remote NVS updated
  - Genuine Mirror which is completely up to date
  - Distance limitation of 300Km
- Asynchronous Mirrors (XRC or Global Mirror)
  - I/O's sent to remote site asynchronously
  - Process Managed by SDM – System Data Mover
    - Ensures all I/O's are applied in sequence at remote site
  - Remote site will be at best few I/O's behind
  - Distance limitation is effectively removed

# GDPS or Geoplex

- Service offering from IBM
  - Implemented using Automated Operations routines
- Exploits Disk technology
  - Complete copy of system at remote site
  - Includes data and system datasets such as logs
  - Kept in step using Mirrors
- Remote site usually offline
- Can have unexpected implications
  - If local mirror breaks GDPS can suspend local processing
  - Prevents bad updates being propagated to DR site

# Some related DSNZPARMS

- **SEQCACH BYPASS/SEQ**
  - Originally whether DB2 I/O should bypass Cache
    - The meaning has changed for all disks since 3390
  - BYPASS now means the disk will perform Sequential Detection
  - SEQ now creates an explicit Prefetch request
    - This will react faster and is now recommended by IBM experts
- **SEQPRES NO/YES**
  - Similar to SEQCACH but for Utilities
  - If set to YES the Cache is more likely to retain pages for subsequent update, particularly when processing NPIs
  - Used for partial LOAD and REORG utilities
  - Again suggest this should be changed to YES

# More DSNZPARMS

- **MGEXTSZ YES /NO**
  - Should DB2 be able to manage Sliding Extents?
  - Note Default changes to Yes in DB2 9 – it was NO in V8
- **TSQTY 0 and IXQTY 0**
  - Sets default object sizes if USING clause omitted
  - Default values use 1 cylinder except for LOBs which use 10 cyls
- **DSVCI YES/NO**
  - Whether to use variable CI sizes
  - Note this is turned on by default when you install V8
- **SVOLARC NO/YES**
  - Yes means allocate a single volume for disk based archives
    - Saves space if you use SMS guaranteed space option
  - Added to DB2 V8 with PQ49630

# So is it still worth monitoring Extent Numbers?

- If you're still using User Managed datasets then YES!
  - Increased extent rules only operate with DFSMS
- For System Managed datasets it depends...
  - Tests show that CPU time is not impacted much but that Elapsed time can increase if there is heavy insert activity
  - A program inserting 1m rows and using Sliding extents with a very low initial size doubled the elapsed compared to 1 extent
    - May be caused by the allocation of each new dataset
- What is certain is that the old rules of thumb need to be rewritten in light of modern technology
- So what should we be monitoring?
  - Index Cluster ratios
  - Use RTS to check numbers of inserts & updates

# Do you need to Spread Datasets across Volumes?

- Although volumes are now a logical concept there are still disks backing the data
  - Also VTOCs are still a potential point of contention
- Each channel can process one I/O each way at a time
- If multiple critical datasets conflict at the disk or channel level it can still be a problem
  - But much harder to detect and avoid than in the past
  - *Look for long PEND & DISCONNECT as proportion of I/O*
- Best opportunities if you have multiple control units
  - Consider Striped Active Logs (and Archives if DB2 9)
  - Ensure duplex copies of the BSDS, Archive Logs and other critical resources are separated between Control Units

# Are backups still needed?

# YES!

- Mirrors can and do fail
  - Numerous war stories concerning such problems
- Having 2 copies of bad data doesn't help recovery
  - GDPS can reduce the risk but not eliminate it
  - What about a program that commits in error?
- Various war stories about failures in error-proof disks
  - Writes to NVS that didn't get written to DASD at peak time etc.

# What about Dual logging?

# YES!

- Same arguments as previous slide
- The log is the most critical recovery resource in DB2
- Compared to your data it's probably very small
- Why would you consider not running with two copies?
  - To put it another way, what would you save??

# Allocating Big Drive Types

- Often discouraged due to contention
  - Increased IOSQ and PEND times if many datasets on volume
- Situation different with Hyper-PAV & Multiple Allegiance
- Model 27's and 54's much more useful
  - Handle large objects more elegantly – remember 59 volume limit
  - Reduce overall number of volumes = management overhead
- Potential issue with 123 extent/volume limit
  - Especially if manually coded small SECQTY
  - Consider Sliding Extents and Extent Consolidation
- Also remember dataset contention still exists
  - Care is still needed with placement of key DB2 datasets

# Allocating your own datasets

- Why would you still want to do this today?
- It prevents exploitation of a lot of new disk technology
  - Extended Format datasets
  - DSSIZE>4G
  - Dataset Striping
  - Almost certainly anything new that emerges
- Many of these benefits are transparent with SMS control
- DB2 9 allows SMS constructs in STOGROUP
  - Omit VOLUMES
  - Add DATACLAS, STORCLAS & MGMTCLAS instead
- Don't use VOL(\* \* \*) in Stogroups
  - Has not been needed or recommended for years

# A few good references

- **IBM Announcements – for z/OS, DS8300, DB2, z10 and more**
- **IBM Storage – Billions and Billions of Bytes**
  - [http://www-03.ibm.com/ibm/history/exhibits/storage/storage\\_intro.html](http://www-03.ibm.com/ibm/history/exhibits/storage/storage_intro.html)
- **How does the MIDAW facility improve the performance of FICON channels using DB2 and other workloads?**
  - *IBM RedPaper 2006 - J. Berger & P. Bruni*
- **Understanding the Performance Implications of PAVs and Multiple Allegiances for Storage Subsystems**
  - Dr. H. Pat Artis , Performance Associates, Inc.
- **Disk storage access with DB2 for z/OS**
  - *IBM Redpaper 2006 - Paolo Bruni & John Iczkovits*
- **DB2 and Storage Management, a Guide to Surviving a Perfect Marriage**
  - *John Iczkovits Paper, various conferences inc. SHARE Feb 2008*

Do Extent Numbers Really Matter Any More?

Steve Thomas

BMC Software

steve\_thomas@bmc.com