Preparing SMSVSAM for HSM and Catalog

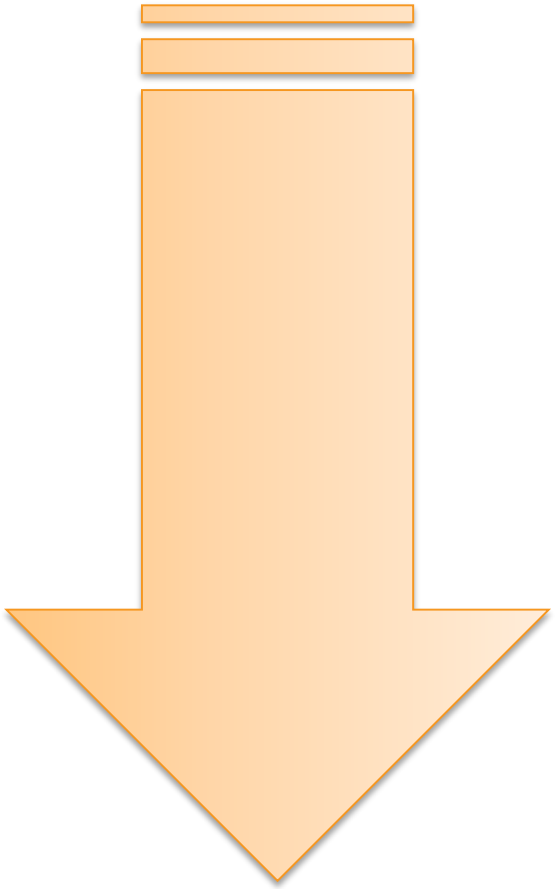
VSAM/RLS Performance and Tuning

Neal Bohling
IBM Development
bohling@us.ibm.com

Session #17832

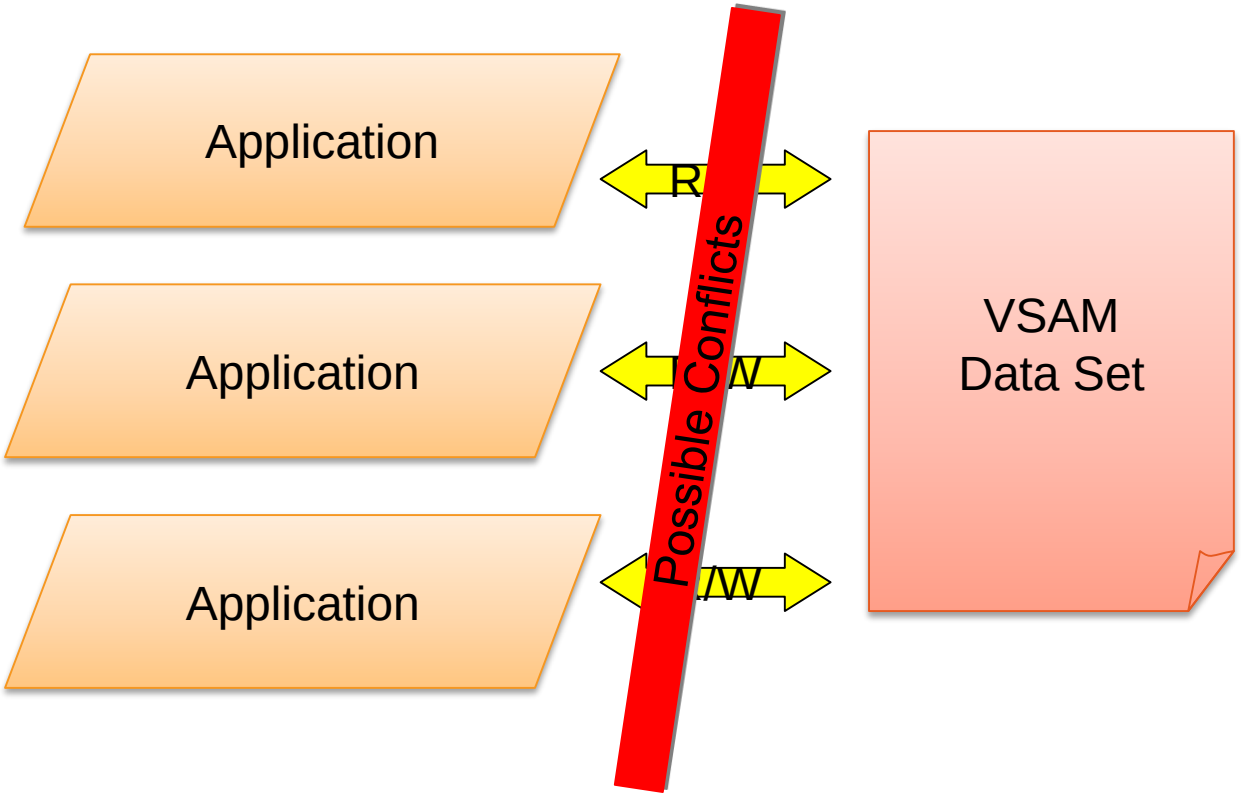
Overview

- Quick SMSVSAM Overview
- Why SMSVSAM for HSM or Catalog?
- Adding Support
  - Changes to SMSVSAM
  - Changes to HSM
  - Changes to Catalog
- Monitoring
  - SMSVSAM
  - HSM
  - Catalog



SMSVSAM Overview

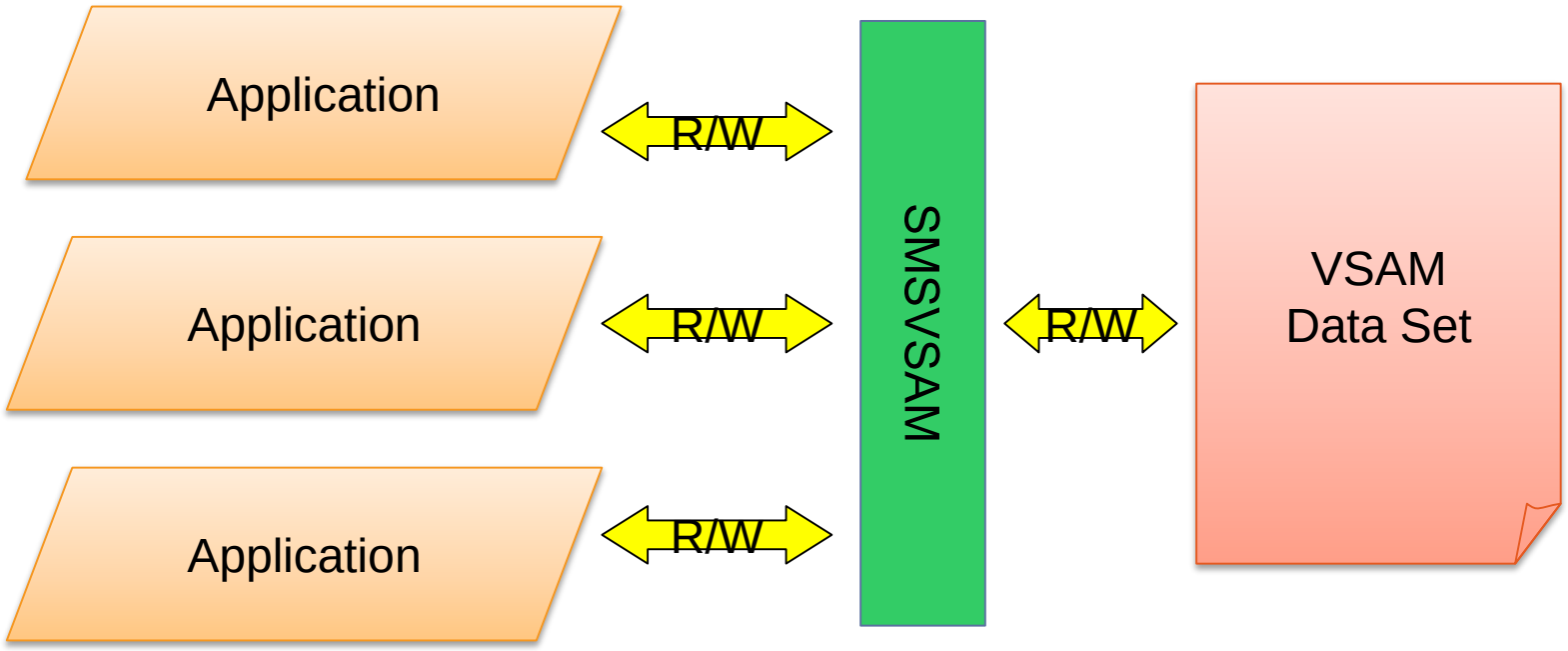
Regular VSAM cannot have multiple updates simultaneously



Serialized at Data Set Level
or application responsible for serialization.

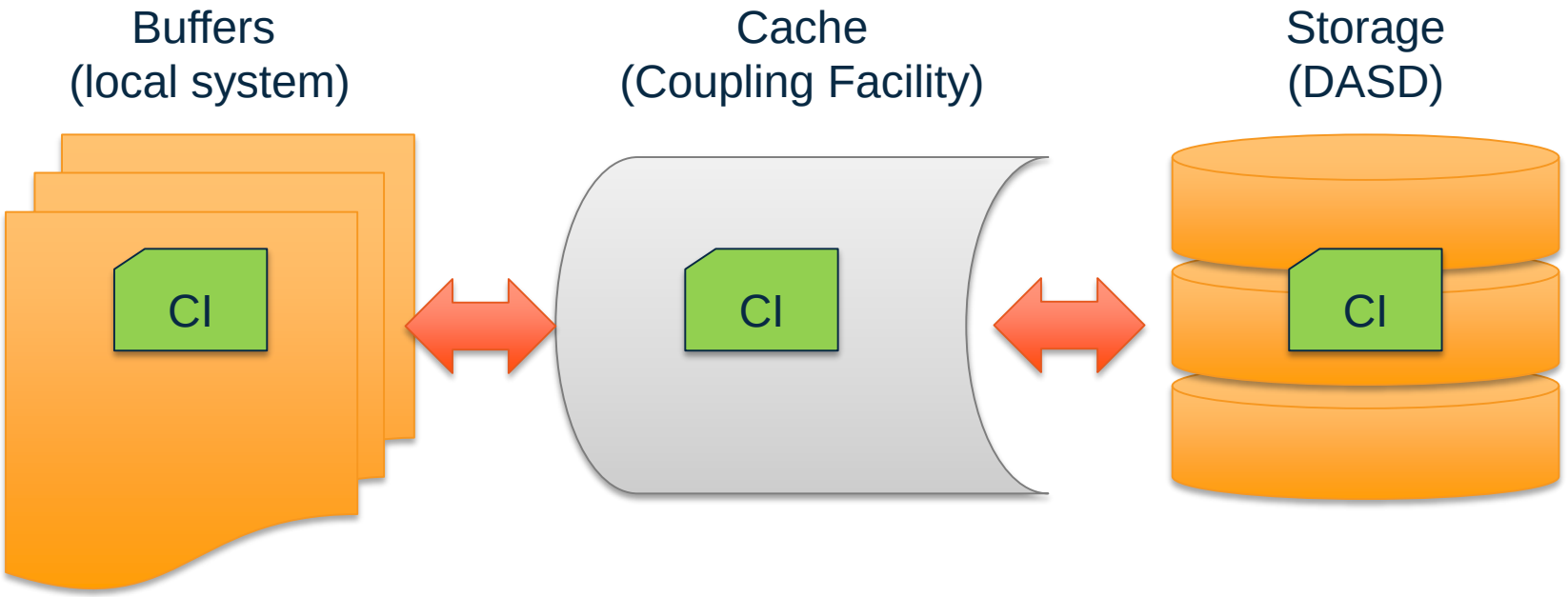
SMSVSAM Overview

- SMSVSAM can handle all that serialization
- Serialization is at RECORD level instead of DATA SET



SMSVSAM Data Storage

- Data is staged across 3 layers
- Stored as full Control Intervals



SMSVSAM General Benefits

- Record-level serialization
  - Higher concurrent throughput
  - Multiple updates / erases / reads simultaneously
- Uses CF Locks
  - Maintains serialization, even past close in case of failure
  - Allows for transaction-level locking
- Cross-PLEX data sharing
  - Buffers and caching provide shortest path to data
  - Reduces I/O
- No single point of failure

Benefits for HSM CDS

- Reduces CDS contention during Primary Space Management and Automatic Backup
- Takes advantage of the data caching / reduces read time
- Avoids STRNO limits

- Performance Improvement*:
  (actual customer data, comparing NonRLS to RLS, 1yr elapsed)

Function	GB Moved Increase	Window Size Decrease
Auto Backup	33%	-25%
Migrate -> ML2	18%	-36%

- Actual Data #2 – AUDIT Processing*
  - Before RLS: Could not complete in 24 hours
  - After RLS: Finished within 4 hours

*source: Glenn Wilcock, *DFHSM Best Practices*, Anaheim 2014, Session 15075

Benefit for Catalog

- **Reduced contention**
  - Eliminates SYSIGGV2 'UCAT' ENQ contention
  - Plans to remove the SYSIGGV2 'sphere' ENQ
  - No need to split catalogs to lower contention
- **Higher throughput**
  - Significant improvement in elapsed time & CPU
  - Much shorter wait times
- **Improved control**
  - Suspend / resume ALL catalogs, PLEX-wide
  - Prevents un-serialized updates

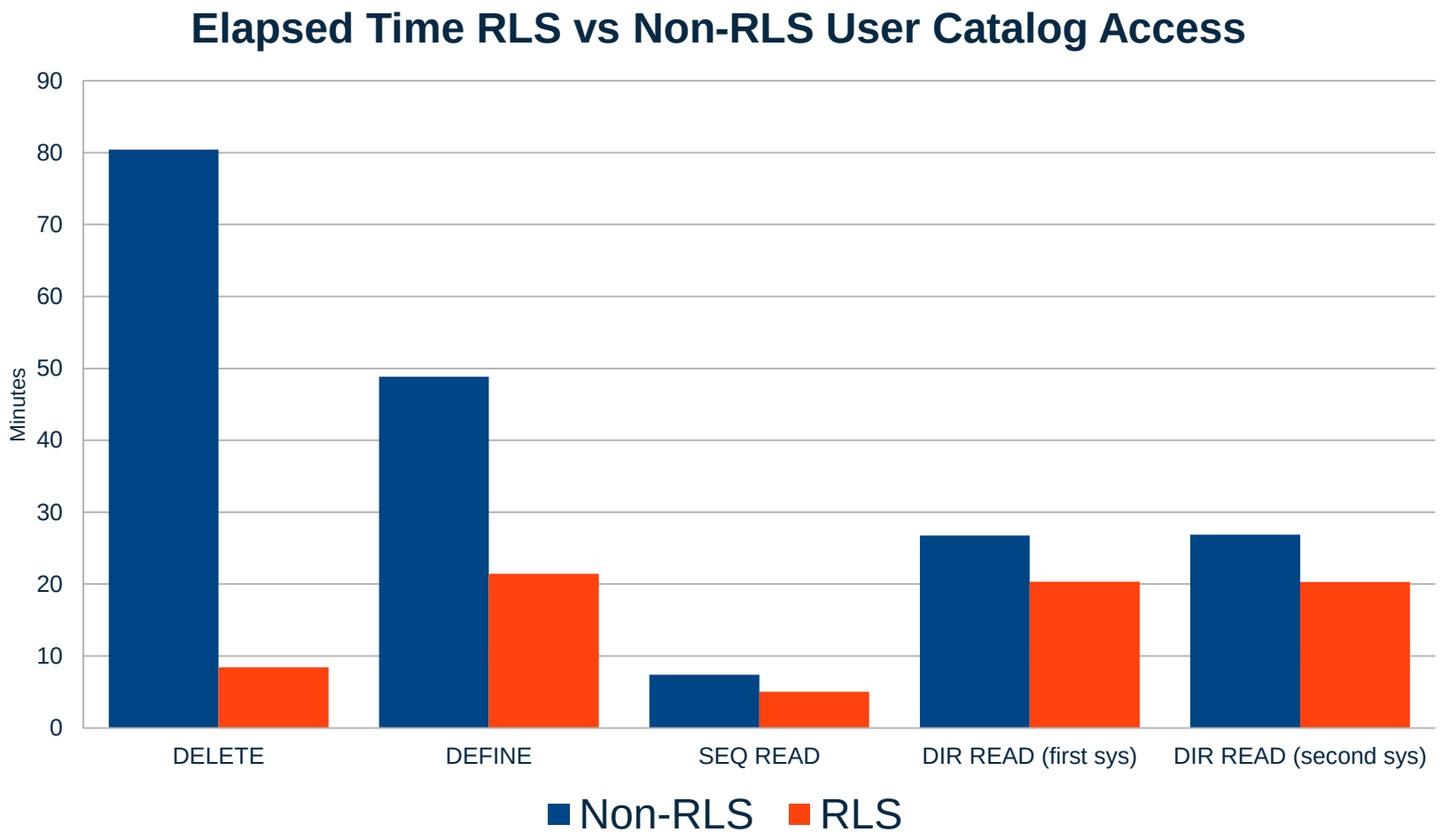
RLS Catalog Performance Benchmark Test

	Elapsed Time (min)		CPU* (sec)		Deltas	
Test	**Non-RLS**	**RLS**	**Non-RLS**	**RLS**	**Elapsed**	**CPU***
DELETE	80.42	8.42	1269.3	298.7	**89.5%**	**77.0%**
DEFINE	48.84	21.42	685.6	130.8	**56.1%**	**80.9%**
SEQ READ	7.40	5.03	65.1	75.2	**32.0%**	**-15.5%**
DIR READ (first sys)	26.77	20.33	94.0	109.6	**24.1%**	**-16.6%**
DIR READ (second sys)	26.86	20.29	95	109.9	**24.5%**	**-15.7%**

***CPU in GRS, CATALOG may see a small increase – best to compare per request**

Test environment: Z10 2097 E12, 3 LPARs, 7 CPUs, 1 CF, z/OS 2.1
Catalog parms: TASKMAX=180, CISIZE(32768) and CISIZE(4096), STRNO(255)
RLSABOVETHEBAR(NO) RLSCFCACHE(ALL) RLSMAXPOOLSIZE(100M) CF Cache size 1G
Catalog RLS vs Catalog VLF at z/OS 2.1
Tests: 300,000 data sets, 100 jobs using 1000 data sets on each LPAR
Source: “Unclog your Systems with z/OS 2.1 – Something New and Exciting for Catalog” by Terri Menendez, IBM
SHARE San Francisco, Spring 2013 Session #12977, 12978

Performance Benchmark Test



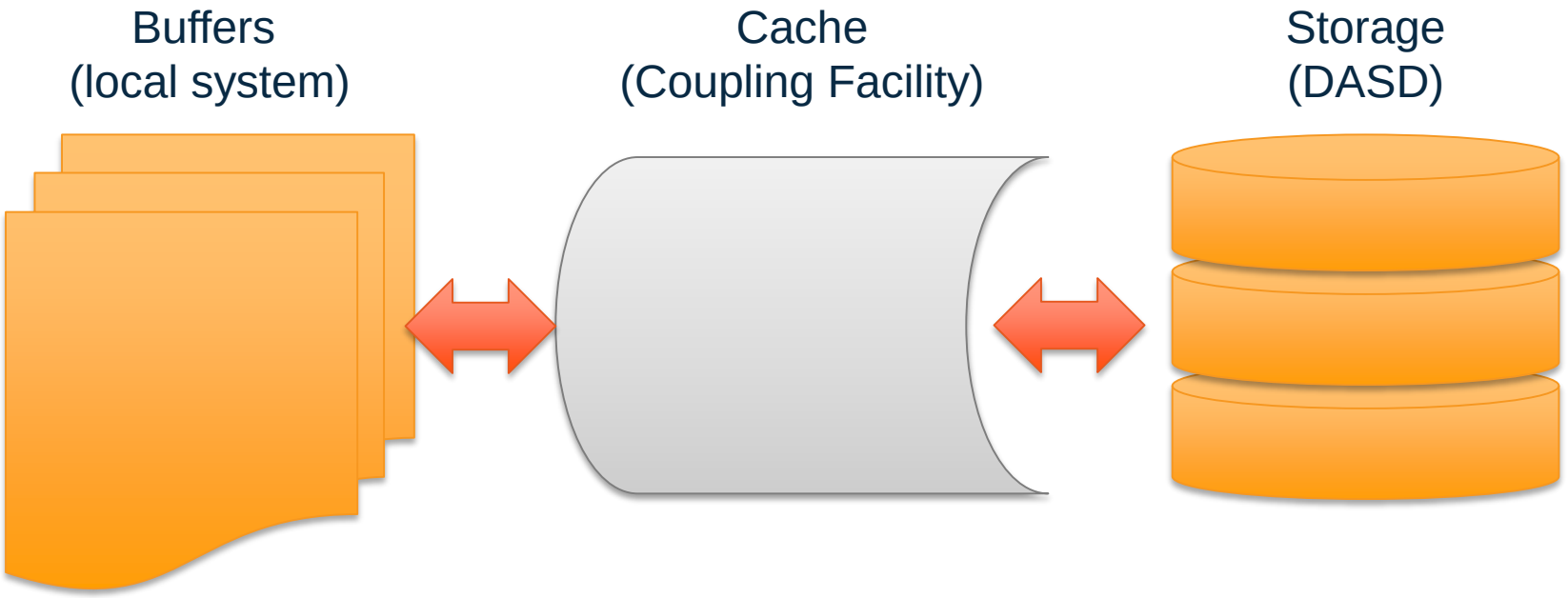
Making the Changes

Gather information

- SMSVSAM
  - Current buffer goal / size
  - Current cache names
- HSM
  - Size of CDS
  - Expected growth
- Catalog
  - Number of catalogs to use RLS
  - Size of catalogs
  - Expected growth

SMSVSAM Changes

- We'll probably need to expand Buffers and Cache
- May also want to expand the lock structure

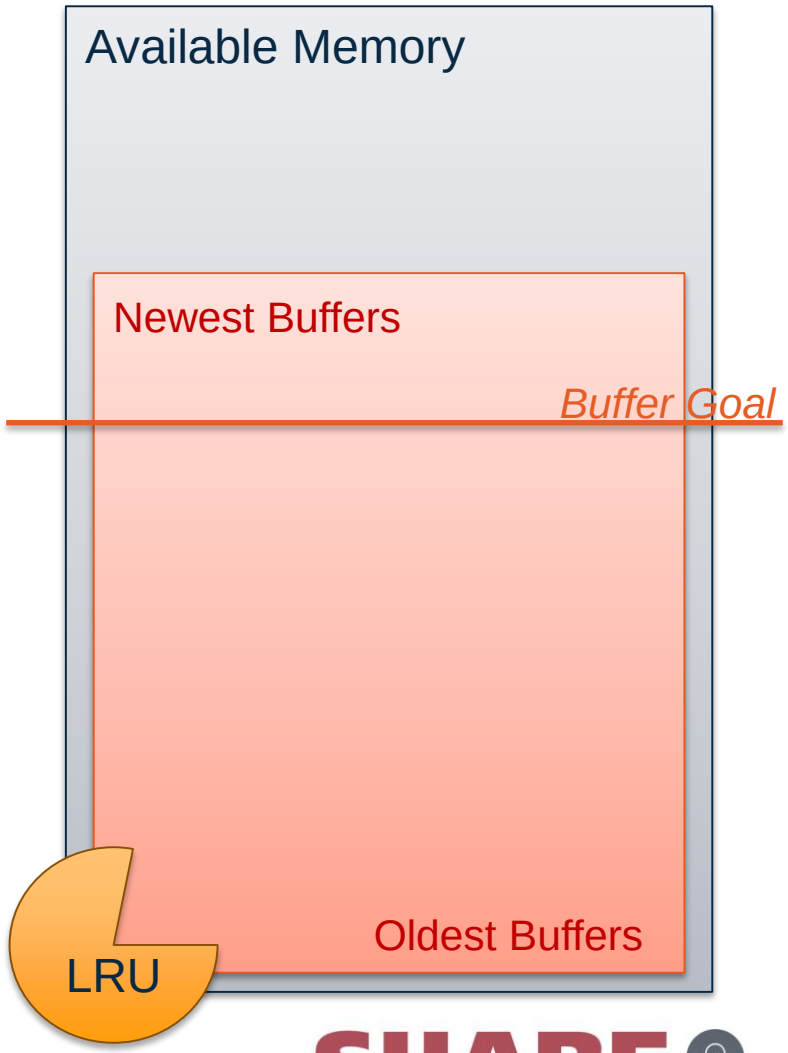


Buffering

Buffer Overview

- Once buffers obtained, they are put on the top of the stack
- New buffers pile on top
- Re-referencing a buffer will pull it back to the top

- RLS Least Recently Used (LRU) routine purges old buffers
- As usage approaches or crosses the goal, LRU speeds up
- Below-the-bar: Panic @ 200%
  Above-the-bar: Panic @ 100%

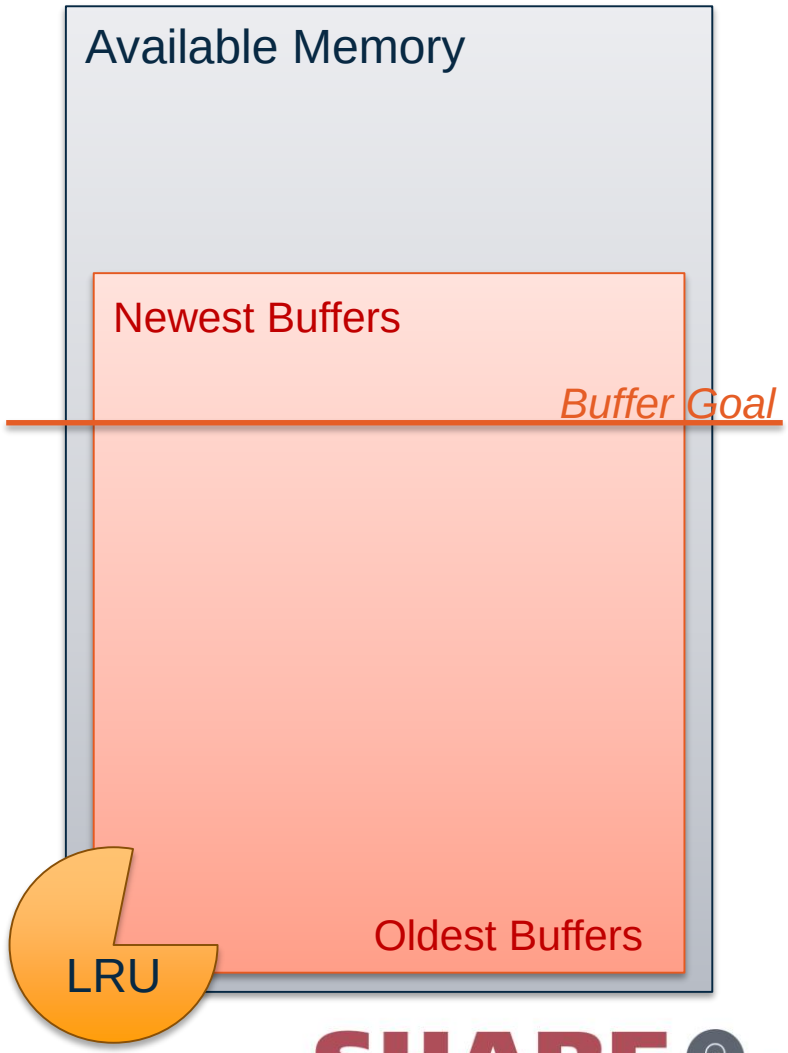


Buffer Overview

- ***What if I don't change my buffer sizes?***

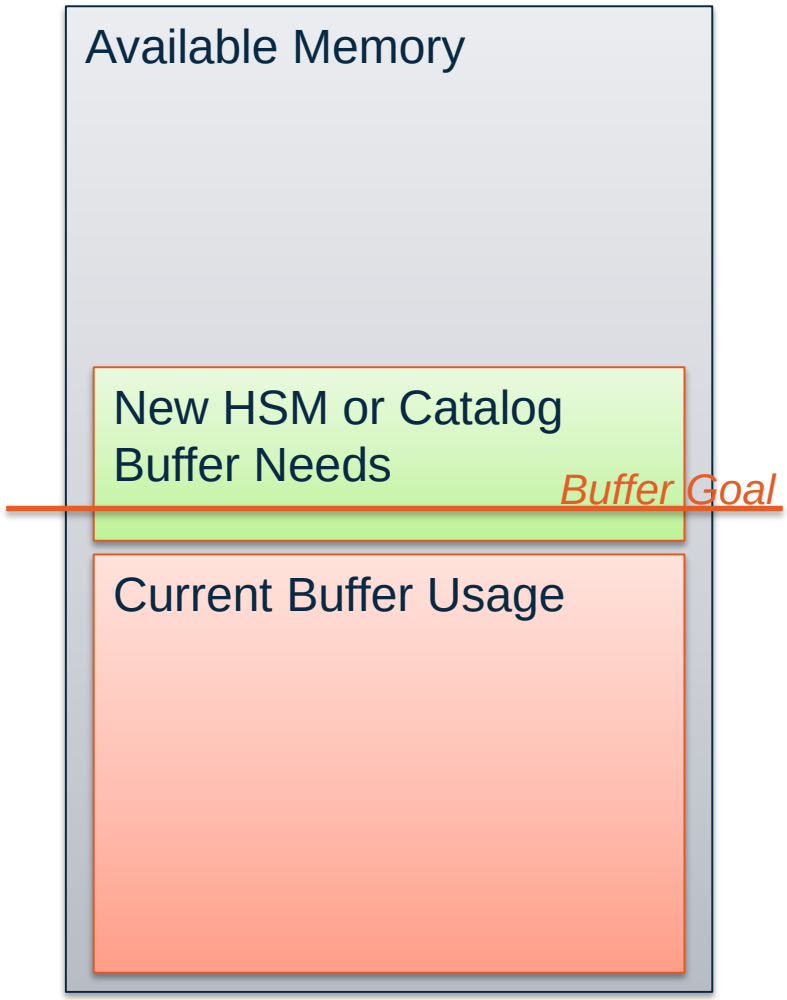
- More data fighting for the same space

- Buffers will be flushed more quickly
- RLS LRU CPU may rise
- Extra caching requests
- Extra I/O



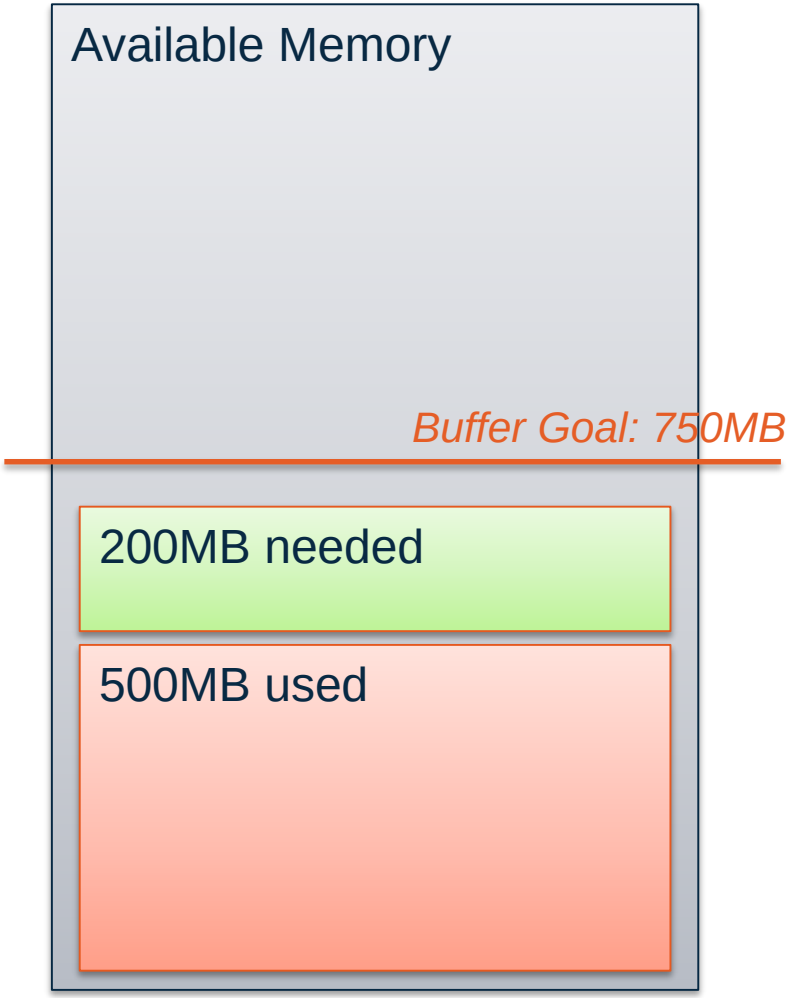
Size Your Buffers

- Calculate Needed Size
  - No easy formula
  - How big are your CDS / Catalogs?
  - Prefer big enough to hold:
    - DS Index
    - Several Data CI
  - Note: buffers are shared
- Add New Size to Old Usage
  - If you are below goal, you may not need to change
  - If you are above goal already, you might want to increase even more



Example #1

- Current usage:
  - Goal: 750MB
  - Usage: 500MB
- New Usage:
  - 5 Catalogs: 100MB
  - HSM CDS: 100MB
- No Need to Change Buffer Goal

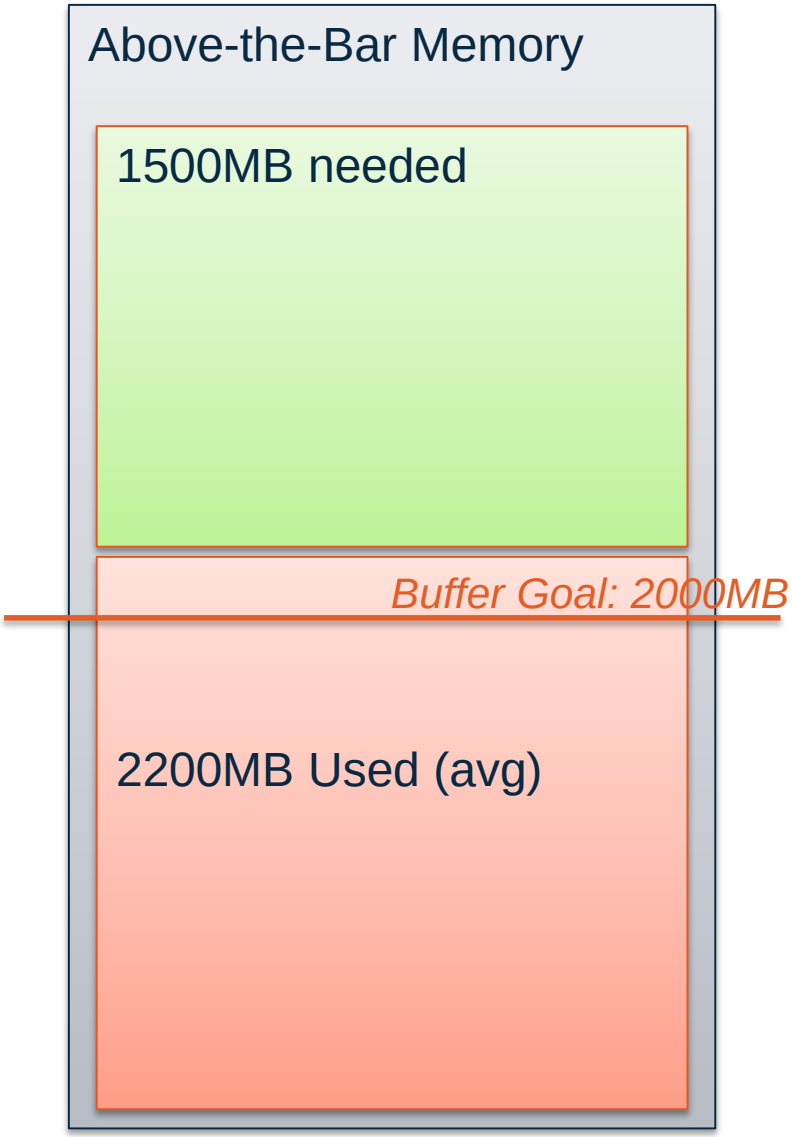


Example #2

- Current usage:
  - Goal: 2000MB
  - Usage: 2200MB
- New Usage:
  - 5 Catalogs: 500MB
  - HSM CDS: 1000MB

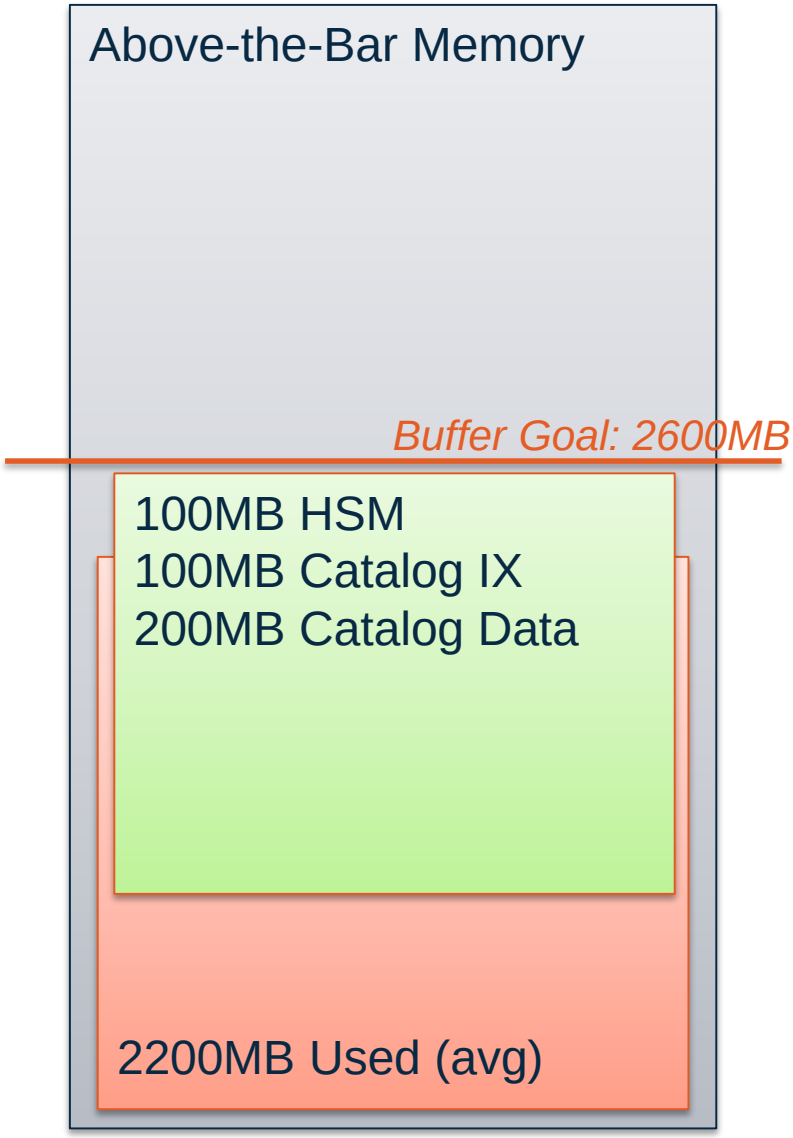
- Solution 1:
  - Raise Goal to 3500MB

- That's a big change.. Consider:
  - How much HSM data is needed concurrently?
  - How much Catalog data is needed concurrently?
  - Does HSM run at the same time as your other workload?



Example #2

- **Solution 2:**
  - HSM PSM runs overnight when other RLS data is offline / batch processing
  - 100MB of concurrent data needed during day for HSM
  - Catalog index total: 100MB
  - Highly active catalogs, so make space for 50% of data
- **Final Changes:**
  - Raise goal to 2600MB
  - Monitor performance



Buffer Sizing

- **RLS_MAX_POOL_SIZE({nnnn|100})** in IGDSMSxx
  - Below the bar GOAL
  - One size for all LPARs
  - Set by first system to start SMSVSAM
  - 10 to 999, Recommendation: 850MB

- **RLSABOVETHEBARMAXPOOLSIZE({ALL,size})** in IGDSMSxx
  - Above the bar GOAL
  - Can set per-system using (SYS1,size1),(SYS2,size2)
  - 500MB to 2TB.. Recommended to keep < 32GB
  - Relies on real storage

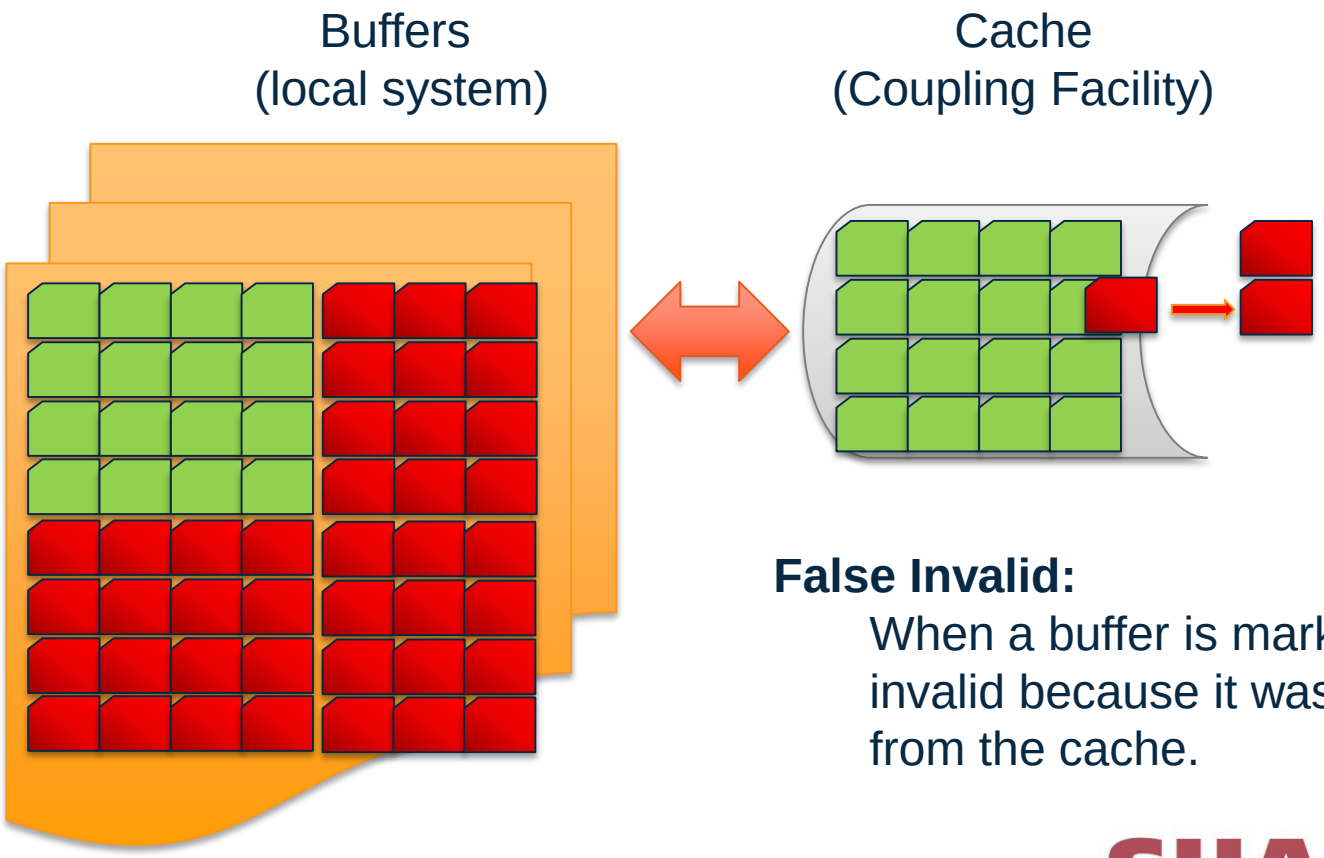
- **RLSFIXEDPOOLSIZE({ALL,size})** in IGDSMSxx
  - Amount of the total real storage, both above and below the bar, that will be permanently fixed (pinned)
  - Does not allow paging of the buffers

- **RLS Above the 2-GB Bar (YES | NO)** in Data Class definition
  - Must be YES to use above the bar buffering
  - Defaults to NO

Caching

Cache Needs

- Cache needs multiply with more systems
- Undersized cache reduces buffer effectiveness

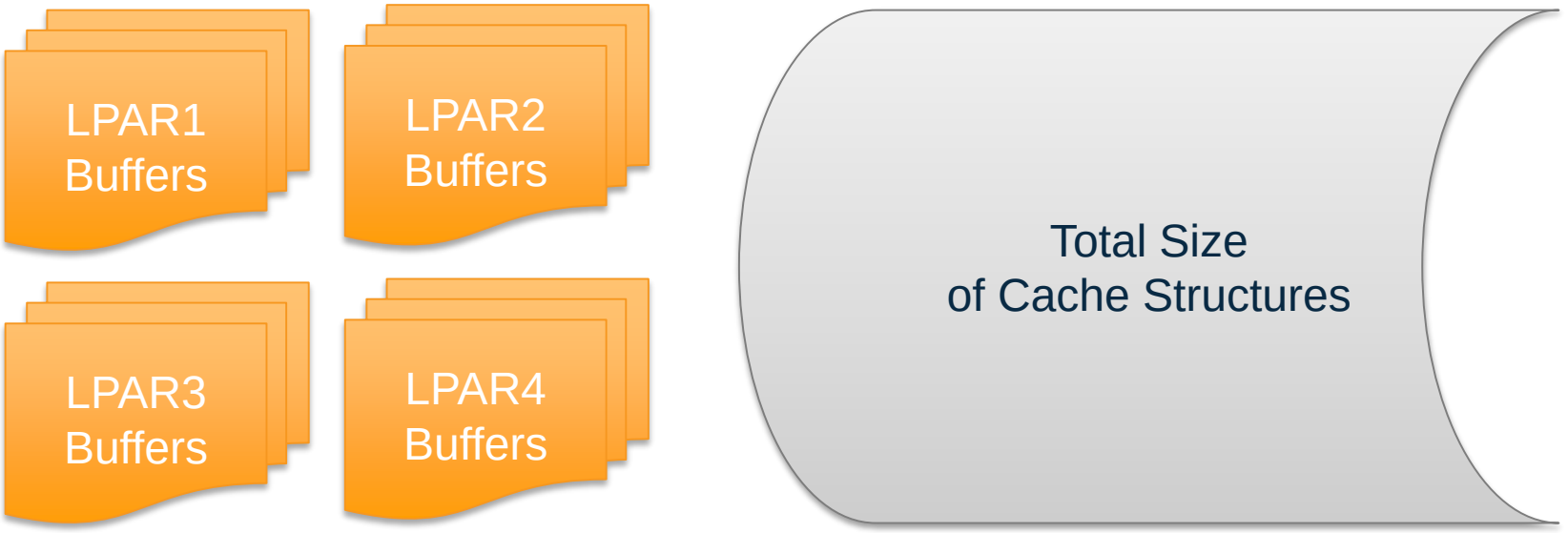


False Invalid:

When a buffer is marked invalid because it was purged from the cache.

Cache Needs

- **"Best" Sizing:**
  - Buffer Size x Number of Systems
- **Optimized Sizing:**
  - Sized of Shared Data + Size of Unique Data from each LPAR



RLS Caching Modes

ALL	Data CIs and Index CIs stored READ or WRITE will add CI to cache
NONE	Cache on index Cis READ or WRITE will add CI to cache
UPDATES ONLY	Data Cis and Index CIs stored Only WRITES will update cache
DIRONLY	No CI data is stored READ or WRITE will update interest

Inside a Cache

Directory Entry

- Holds control information
- Holds interest information
- One per CI
- Relatively Small

Data Element

- Holds the stored data
- Up to 2k in size
- Many per CI (depending on CISIZE)
- Ex: 6k CI would require 3 DE

Caching Modes

Mode: ALL or UPDATESONLY

Dir Entries	Data Elements		
Data CI Registered	Data CI Part 1	Data CI Part 2	Data CI Part 3
Index CI Registered	Index CI		

Caching Modes

Mode: NONE

Dir Entries	Data Elements
Data CI Registered	
Index CI Registered	Index CI

Caching Modes

Mode: DIRONLY

Dir Entries	Data Elements
Data CI Registered	
Index CI Registered	

Caching Modes

- Can have a big impact on how cache is used
- Can reduce cache needs

- **RLS CF Cache Value** in Data Class
  - A = All
  - N = None
  - U = Updates
  - D = Directory Only

- Requires Rls_MaxCFFeatureLevel(A)
- Toleration: OA36443, OA36415

Create New Structures

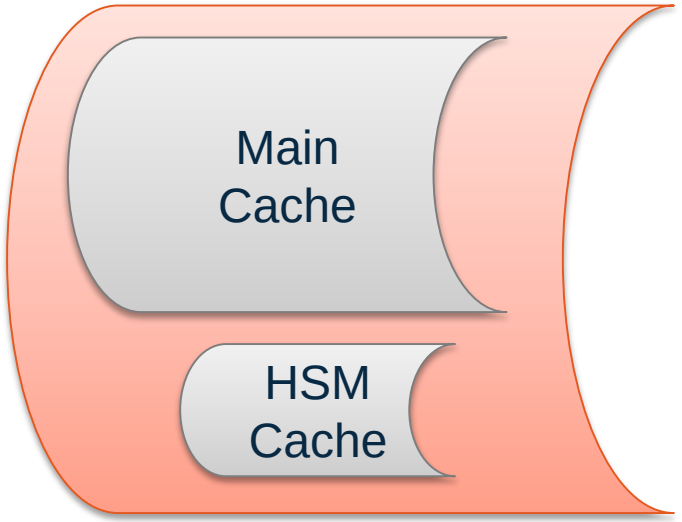
- **Recommendation: Use new cache structures**
  - Allows for separation between workload and CDS
  - Keeps load off of current structures
  - Makes sizing easier

- Define in CFRM Policy
  - Note: RLS does not grow structures, so use INITSIZE=MAXSIZE
- Update cache sets / storage classes in SMS via ISMF
  - May want to create new cache set
  - May want to create new Storage Class

- **Recommendation: Separate Catalog / HSM / Work SC**

Caching Example

- Two Systems
- RLS CF Cache = ALL
- Shared Data: 2048MB
- Unique Data:
  - System1: 400MB
  - System2: 2448MB
- New HSM Data = 100MB



- Total Cache Size:
  - 2048 (shared) + 400 + 2448 (unique) = 4896MB (~4.8GB)
- CICS/Online Cache Size = 4896MB
- HSM Cache Size = 100MB

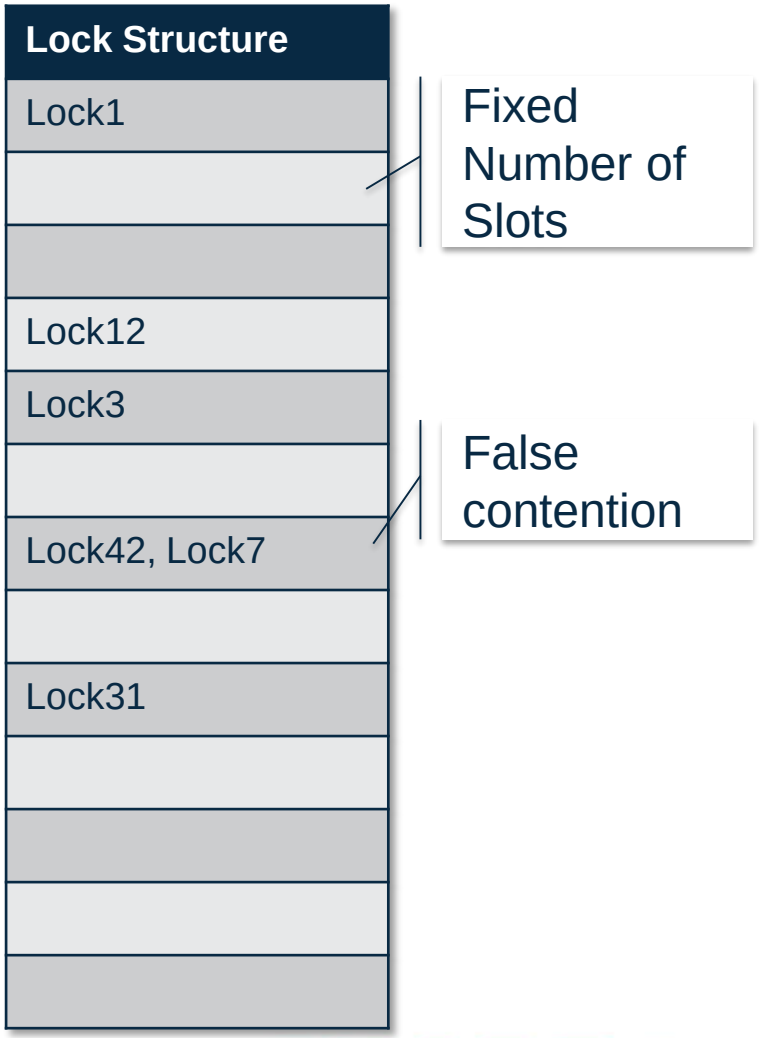
Lock Structures

Lock Structure Considerations

- Lock Structure
  - Divided into slots
  - Uses hash to place lock
  - Too few slots results in hash collisions
  - "False contention"

- **Recommendation: Use secondary lock structures**
  - TEST, WORK, HSM, Catalog
  - Prevents overload in one from infecting others

- Monitor False Contention rates



Multiple Lock Structures

- Allows separation of workloads
- SMS allows up to 256 different lock sets
- Can use any name. IGWLOCK01 is easy

- Update SMS:
  - Storage Class → SMS Lock Set → Lock Structure

- Holds all record locks for that data set
  (does not hold Component or Special locks)

SHCDS considerations

SHCDS Considerations

- Share Control Data Sets (SHCDS)
- Maintains data set level information in case of failure
- Used to build Lost Locks information
- Hold:
  - Data set / cache set / lock structure pairing info
  - Data set / region pairings

- You may want to grow the SHCDS **only** if you are adding a lot of data sets (lots of catalogs)

SMSVSAM Changes Summary

- Examined each in detail
- Made changes based on new information

Buffers
(local system)

Cache
(Coupling Facility)

Lock
Structure

Lock Structure
Lock1
Lock12
Lock3
Lock42, Lock7

HSM Changes

HSM Changes

- Update HSM Data Sets:
  - IDCAMS ALTER cdsname LOG(NONE)
  - Consider SHAREOPTIONS(2 3) if you want non-RLS read access while open
  - Make sure Storage Class points to a Cache Set

- Startup Keyword **CDSSHR** = {YES | **RLS** | NO}

- Make sure SYSVSAM ENQs are SYSTEMS (not RNL excluded)

Catalog Changes

Catalog Changes

- **To ENABLE for a single catalog:**
  - IDCAMS ALTER ucat LOG(NONE)
  - Ensure that Storage Class points to Cache Set
  - F CATALOG,RLSENABLE(ucat)
  - IEC352I MODIFY CATALOG cat.name TO STATE RLSENABLE SUCCESSFUL
  - F CATALOG,RLSQUIESCE(ucat)

- **Notes:**
  - Requires that SMSVSAM be up and active
  - Only available on z/OS 2.1 and up

IDCAMS Tools Support

- **REPRO, PRINT, IMPORT, EXPORT** supported
- To use, specify
  - **RLSSOURCE**(YES | NO | QUIESCE)
  - **RLSTARGET**(YES | NO | QUIESCE)

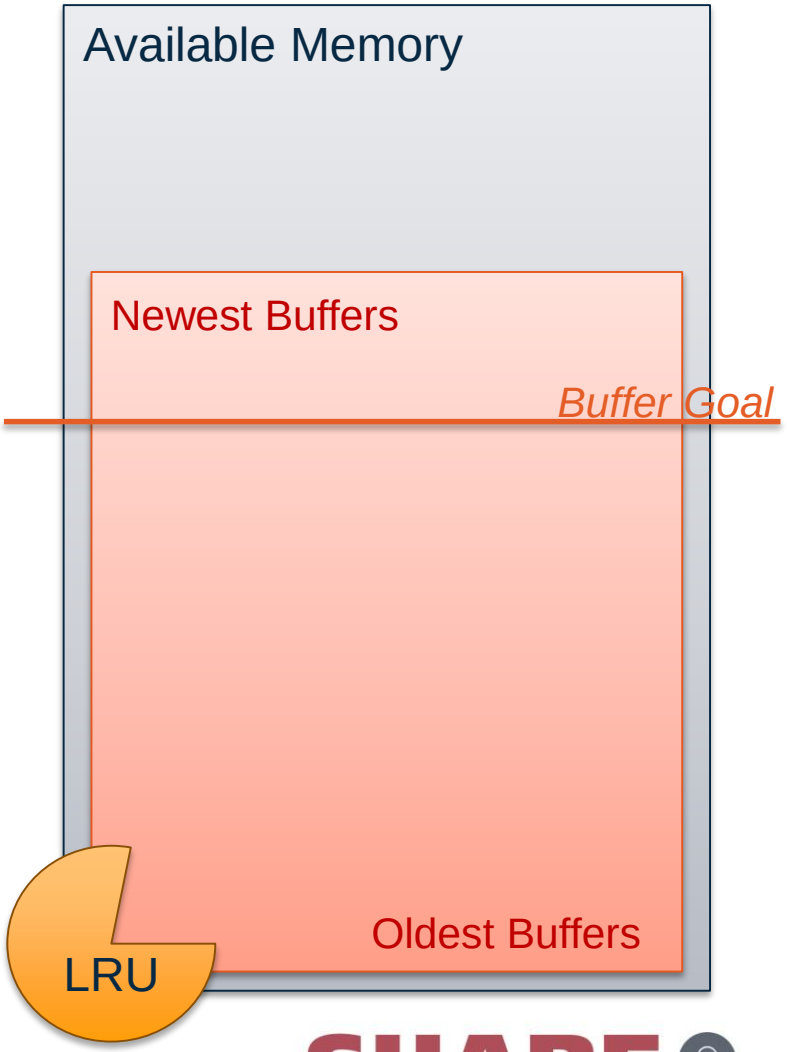
- Options:
  - YES – use RLS mode to access data set
  - NO – use Non-RLS (regular VSAM) to access data set
  - QUIESCE – Use Non-RLS mode, but QUIESCE first.

Monitoring and Tuning

Monitoring Buffers

- Measurements to Watch:
  - LRU Mode
  - Buffer usage vs buffer goal
  - BMF Hit Rate
  - Reclaim Rate

- SMF 42 subtype 15, 16, 19
- RMF Monitor III –
  - RLSSC panel
  - RLSLRU panel
- OMEGAMON XE



SMF42 Buffer-related Fields

- **Subtype 19 – LRU Summary**
  - Available at SYSPLEX-level or system-level
  - Split into below-the-bar and above-the-bar sections
  - **Buffer goal:** SMF42JQI
  - **Buffer used:** SMF42JQM
  - **Accelerated/panic mode rate:** SMF42JPH / SMF42JPG

- **Subtype 15, 16 – Storage Class & Data Set**
  - Also split into PLEX / system
  - Also split into below / above the bar
  - Additionally split into SEQ / DIR
  - **BMF Hit rate = SMF42FII / SMF42FIG**

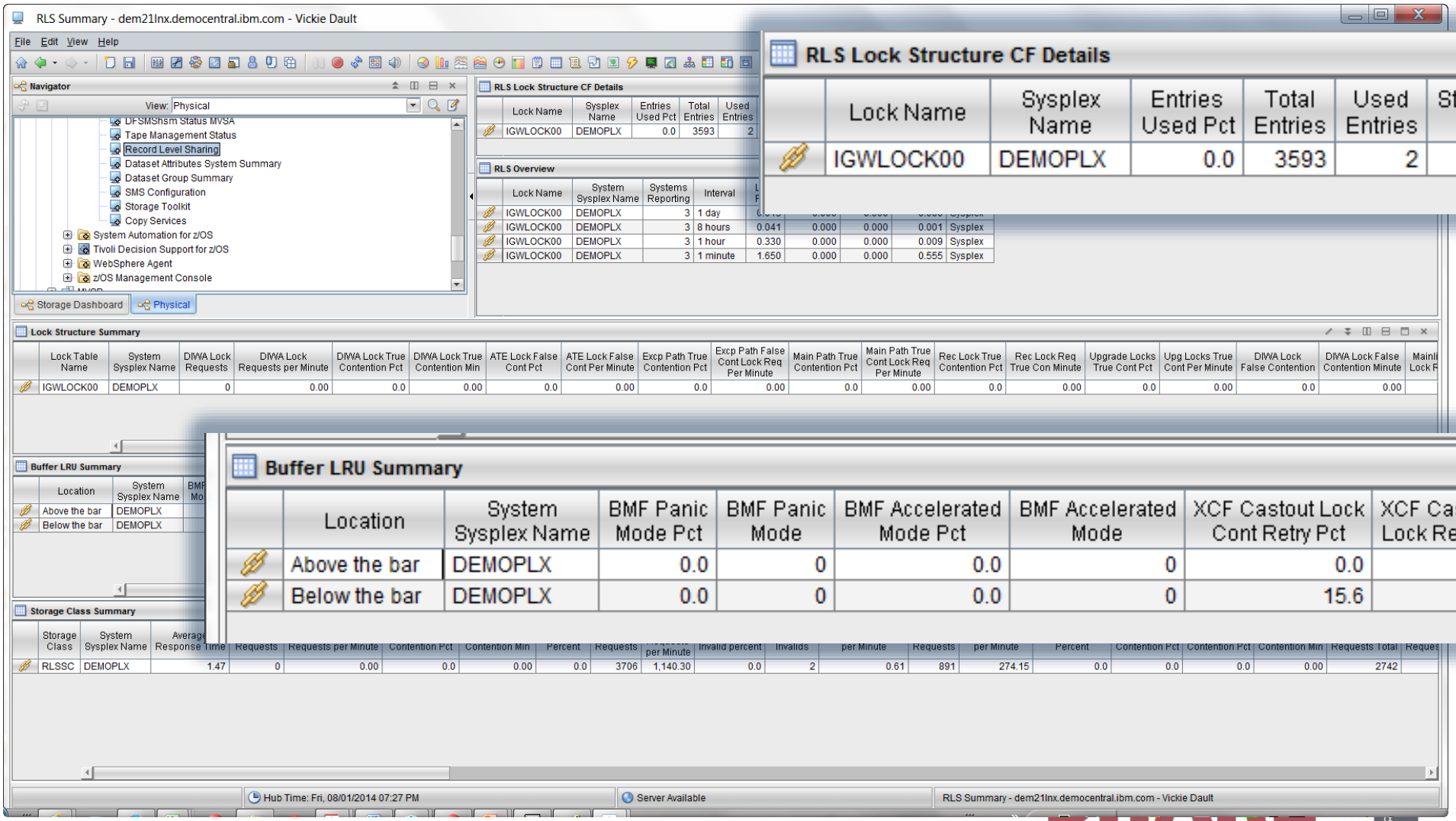
RMF MON III - RLSLRU

RMF V2R1 VSAM LRU Overview - CAPTKEN1 Line 1 of 3

Samples: 100 Systems: 1 Date: 03/08/14 Time: 15.23.20 Range: 100 Sec

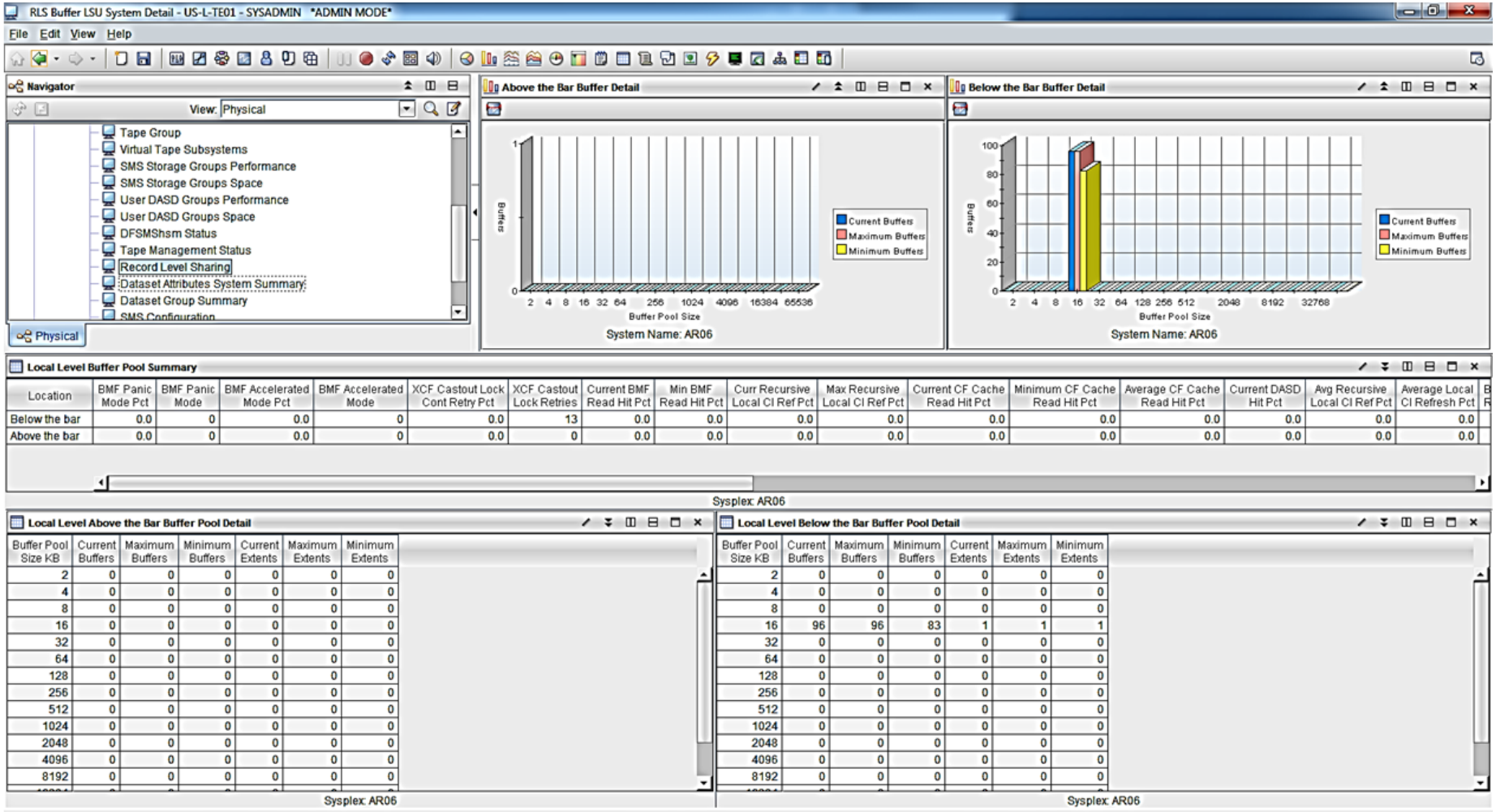
MVS System	Avg CPU Time	Buffer Size - Goal	Buffer Size - High	Accel %	Reclaim %	Read ------ BMF%	Read ----- CF%	Read ----- DASD%
SYSTEM1								
Below 2GB	<.001	100M	10M	0.0	0.0	0.0	0.0	0.0
Above 2GB	<.001	**1000M**	**328M**	**0.0**	0.0	**99.4**	0.0	0.6

OMEGAMON XE for Storage



OMEGAMON XE for Storage

RLS Workspaces - Buffer LRU System Detail Workspace

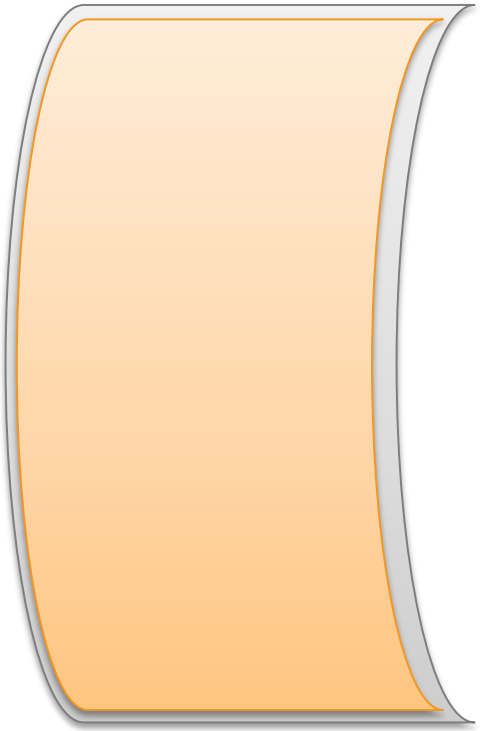


Buffer Pool Statistics for a single z/OS image

Monitoring the Cache

Monitoring the Cache

- Caches:
  - Entries claimed until structure is full
  - Once full, entries are RECLAIMed when needed
  - Even after close, cache entries are "used" until they are reclaimed
  - Reclaims based on age like buffers



- As a results:
  - Percent Full is not relevant
  - Reclaim Rate is not relevant

- **Measurement to Watch: BMF FALSE INVALIDS**
  - **If > 5%, time to raise cache sizes**

RMF MON III – RLSSC & RLSDS

```
                    RMF V2R1   VSAM RLS Activity  - CAPTKEN1        Line 1 of 5

Samples: 100     Systems: 1    Date: 03/08/14  Time: 15.23.20  Range: 100   Se

                < 2GB / > 2GB
LRU Status    : Good  / Good
Contention % :   0.0 /   0.0
False Cont % :   0.0 /   0.0

Stor Class  Access  Resp   -------- Read ----------  ------ BMF -------  Write
                    Time    Rate  BMF%   CF%  DASD%  Valid%  False Inv%  Rate
RLSSTOR
  Below 2GB  DIR    0.000   0.00   0.0   0.0   0.0    0.0      0.00      0.00
             SEQ    0.000   0.00   0.0   0.0   0.0    0.0      0.00      0.00
  Above 2GB  DIR    0.001   2254  90.4   9.0   0.6    100      0.10      1137
             SEQ    0.000   0.00   0.0   0.0   0.0    0.0      0.00      0.00
```

Monitoring the Lock Structure

Monitoring Lock Structures

- **False Contention**
  - Causes performance degradation
  - Two locks incorrectly stored in the same spot
  - Indicates that Lock Structure is too small
  - Target rate < 5%

- To review false contention
  - Health Check
  - D SMS,CFLS command
  - RMF CF activity report
  - SMF 42 subtype 17

Health Check

- VSAMRLS_CFLS_FALSE_CONTENTION
  - Average of the last hour
  - Samples taken every second
  - Has configurable threshold

```
HZS0002E CHECK(IBMVSAMRLS,VSAMRLS_CFLS_FALSE_CONTENTION):
IGWRH0131E FALSE LOCK CONTENTION HAS BEEN DETECTED.
  THE CURRENT FALSE CONTENTION RATE IS 7.120%.
```

D SMS,CFLS

```
IGW320I 11:37:11 Display SMS,CFLS(IGWLOCK00 )
  PRIMARY STRUCTURE:IGWLOCK00 VERSION:CCC6425A4759D195 SIZ
  RECORD TABLE ENTRIES:10358 USED:2
  SECONDARY STRUCTURE:IGWLOCK00 VERSION:CCC6425AD06E219A SIZE:4096K
  RECORD TABLE ENTRIES:10358 USED:2
  LOCK STRUCTURE MODE: DUPLEXED  STATUS: ENABLE
  System    Interval    LockRate    ContRate  FContRate    WaitQLen
  SYSTEM1   1 Minute       585.6       0.000      0.031        0.00
  SYSTEM1    1 Hour         95.1       0.000      0.000        0.00
  SYSTEM1    8 Hour         11.9       0.000      0.000        0.00
  SYSTEM1     1 Day          8.9       0.000      0.000        0.00
  *** No other systems provided data

  ***************** LEGEND ******************
    LockRate = number of lock requests per second
    CONTRATE = % of lock requests globally managed
    FCONTRATE = % of lock requests falsely globally managed
    WaitQLen = Average number of requests waiting for locks
```

Total Available Locks

Monitoring False Contention

- RMF Mon III – Structure details on IGWLOCK00
  - S.7 - “Coupling Facility Activity”
- RMF Mon III – RLSSC and RLSDS
- SMF 42 subtypes 15,16,17
  - SMF42FOA – Number of Lock Requests
  - SMF42FOC – Number of Lock Requests causing False Contention

RMF MON III – RLSSC & RLSDS

```
                    RMF V2R1   VSAM RLS Activity  - CAPTKEN1        Line 1 of 5

Samples: 100     Systems: 1    Date: 03/08/14  Time: 15.23.20  Range: 100   Se

                < 2GB / > 2GB
LRU Status    : Good  / Good
Contention % :   0.0 /   0.0
False Cont % :   0.0 /   0.0

Stor Class  Access  Resp    -------- Read ----------  ------ BMF -------  Write
                    Time     Rate  BMF%   CF%  DASD%  Valid%  False Inv%  Rate
RLSSTOR
  Below 2GB  DIR    0.000    0.00   0.0   0.0    0.0    0.0       0.00     0.00
             SEQ    0.000    0.00   0.0   0.0    0.0    0.0       0.00     0.00
  Above 2GB  DIR    0.001    2254  90.4   9.0    0.6    100       0.10     1137
             SEQ    0.000    0.00   0.0   0.0    0.0    0.0       0.00     0.00
```

Monitoring Lock Structure

- If structure fills, you'll see these messages:
  - IEC161I 248-ccc – indicates failure during OPEN
  - IGW326W *** Warning *** DFSMS SMSVSAM RECORD TABLE IN lockstructurename IS percent % FULL.
- Once full, you'll need to increase the size and rebuild
- Track usage via D SMS,CFLS command

D SMS,CFLS – Monitoring Lock Structure

```
IGW320I 11:37:11 Display SMS,CFLS(IGWLOCK00 )
  PRIMARY STRUCTURE:IGWLOCK00 VERSION:CCC6425A4759D195 SIZ
  RECORD TABLE ENTRIES:10358 USED:2
  SECONDARY STRUCTURE:IGWLOCK00 VERSION:CCC6425AD06E219A SIZE:4096K
  RECORD TABLE ENTRIES:10358 USED:2
  LOCK STRUCTURE MODE: DUPLEXED  STATUS: ENABLE
  System    Interval    LockRate    ContRate  FContRate    WaitQLen
  SYSTEM1   1 Minute       585.6       0.000      0.031        0.00
  SYSTEM1    1 Hour         95.1       0.000      0.000        0.00
  SYSTEM1    8 Hour         11.9       0.000      0.000        0.00
  SYSTEM1     1 Day          8.9       0.000      0.000        0.00
  *** No other systems provided data

  ***************** LEGEND ******************
    LockRate = number of lock requests per second
    CONTRATE = % of lock requests globally managed
    FCONTRATE = % of lock requests falsely globally managed
    WaitQLen = Average number of requests waiting for locks
```

Total Available Locks

Using SHCDS

IDCAMS LISTSUBSYS(ALL)

Show who holds locks

```
 SHCDS LISTSUBSYS(ALL)
----- LISTING FROM SHCDS ----- IDCSH03
-------------------------------------------------------------------------------
                                 RECOVERY          LOCKS          LOCKS          LOCKS
SUBSYSTEM NAME   STATUS          NEEDED            HELD           WAITING        RETAINED
-------------    -------------   -------------   ----------------------------------------
IYNXV            ONLINE--ACTIVE  NO                      200              0              0
    DATA SETS IN LOST LOCKS-----------        0
    DATA SETS IN NON-RLS UPDATE STATE--       0
    TRANSACTION COUNT-----------------        1
IYNX5            ONLINE--ACTIVE  NO                        0              3              0
    DATA SETS IN LOST LOCKS-----------        0
    DATA SETS IN NON-RLS UPDATE STATE--       0
    TRANSACTION COUNT-----------------        1
SMSVSAM          BATCH---ACTIVE  NO                        0              0              0
    DATA SETS IN LOST LOCKS-----------        0
    DATA SETS IN NON-RLS UPDATE STATE--       0
    TRANSACTION COUNT-----------------        0
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
```

Using SHCDS

IDCAMS LISTSUBSYSDS(ALL)

```
 SHCDS LISTSUBSYSDS(RETLK05A)
----- LISTING FROM SHCDS ----- IDCSH04
-------------------------------------------------------------------------
SUBSYSTEM NAME---- RETLK05A      SUBSYSTEM STATUS----ONLINE--FAILED


DATA SET NAME /                          LOCKS              NON-RLS     PERMIT
CACHE  STRUCTURE /  RETAINED   LOST      NOT       RECOVERY UPDATE      FIRST TIME
LOCK STRUCTURE      LOCKS      LOCKS     BOUND     REQUIRED PERMITTED   SWITCH
---------------     --------   -----     -------   -------- ---------   -----
SYSPLEX.KSDS.RETLK05.ABC
CACHE01
TESTLOCK01          YES        NO        NO        NO       NO          NO
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
```

Show region / data set pairings

Monitoring HSM

- **DFSMSrmm Report Generator**
  - Generate reports via ISPF panels
  - Customizable reports
  - Reports based on FSR, WWFSR, Inventory via DCOLLECT
  - See sessions 17592, 17717 (Monday) for more information

- **FSRSTAT**
  - REXX program that formats FSR records
  - Shipped with HSM
  - Easy to use and customize

Monitoring Catalog

F CATALOG,ALLOCATED

```
IEC348I ALLOCATED CATALOGS 118
*CAS*************************************************************
* FLAGS -VOLSER-USER-CATALOG NAME                              % *
* YSU-R- XP0301 0001 BOHLING.RLS.UCAT                          1 *
* Y-I--- USRPAK 0001 SYS1.MVSRES9.MASTCAT                      1 *
*******************************************************************
* Y/N-ALLOCATED TO CAS, S-SMS, V-VLF, I-ISC, C-CLOSED, D-DELETED,  *
* R-SHARED, A-ATL, E-ECS SHARED, K-LOCKED, U-RLS, W-SUSPENDED      *
*CAS*************************************************************
```

Monitoring Catalog

F CATALOG,REPORTS,PERFORMANCE

```
IEC359I CATALOG PERFORMANCE REPORT 127
*CAS*************************************************
*  Statistics since 21:45:12.84 on  03/07/2014        *
*  -----CATALOG EVENT----   --COUNT--  ---AVERAGE---  *
*  Entries to Catalog            840     27.624 MSEC  *
*  BCS ENQ Shr                   636      0.036 MSEC  *
*  BCS ENQ Shr Sys               380      9.266 MSEC  *
*  BCS ENQ Excl                    5      0.013 MSEC  *
*  BCS ENQ Excl Sys                5      9.830 MSEC  *
*  BCS DEQ                     1,255      3.879 MSEC  *
*  BCS Allocate                   13     58.862 MSEC  *
*  SMF Write                      35      0.020 MSEC  *
*  CAS MLA Lock                    1      0.288 MSEC  *
*  VVDS Format                     2     10.147 MSEC  *
*  MVS Allocate                    6    127.281 MSEC  *
*  SMS Active Config               3      0.073 MSEC  *
*  SYSVSAM S ENQ Excl             13     36.419 MSEC  *
*  SYSVSAM S DEQ                  13      8.585 MSEC  *
*  SYSVSAM D ENQ Shr              13      4.689 MSEC  *
*  SYSVSAM D DEQ                  13     10.067 MSEC  *
*CAS*************************************************
       * (truncated to fit)
```

Monitoring Catalog

F CATALOG,REPORT,CATSTATS

```
IEC359I CATALOG I/O STATS REPORT 143
*CAS*****************************************************************
*     ADDS  UPDATES     GETS   GETUPD  DELETES  BUFNI  BUFND STRNO *
*                                                                   *
* BOHLING.RLS.UCAT                                                  *
*       4        1       36        3        0      1      2   180 *
* SYS1.MVSRES9.MASTCAT                                              *
*       2        3    2,352        3        0      4      4     2 *
*CAS*****************************************************************
```

Summary

- Benefits of RLS
- Gathered Information
- Calculated Changes
- Made Changes
- Monitored Performance

Preparing SMSVSAM for HSM and Catalog

VSAM/RLS Performance and Tuning

Neal Bohling
IBM Development
bohling@us.ibm.com

Session #17832

Appendix - Lock Structures - Sizing

- Size requirement depends on how many locks are active

- **Lock_Structure_Size =**
  10M * Number_Of_Systems * Lock_Entry_Size

  - Where Lock_Entry_Size, depends on the MAXSYSTEM
    - MAXSYSTEM <= 7 Lock_Entry_Size = 2
    - MAXSYSTEM >= 8 & <24 Lock_Entry_Size = 4
    - MAXSYSTEM >=24 & <=32 Lock_Entry_Size = 8

- Example: MAXSYSTEM = 16, with 8 systems in sysplex:
- Lock_Structure_Size = 10M * 4 * 8 = 320M

  - Note: Minimum size of 13M is recommended
  - You can also use CFSIZER to help determine appropriate size: http://www-947.ibm.com/systems/support/z/cfsizer/vsamrls/

Notices & Disclaimers

Product information and data has been reviewed for accuracy as of the date of initial publication. Product information and data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the products and/or programs described herein at any time without notice.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Consult your local IBM representative or IBM Business Partner for information about the product and services available in your area.

Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectually property rights, may be used instead. It is the user's responsibility to evaluate and verify the operation of any non-IBM product, program or service.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS"WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. IBM is not responsible for the performance or interoperability of any non-IBM products discussed herein.

Notices & Disclaimers

The performance data contained herein was obtained in a controlled, isolated environment. Actual results that may be obtained in other operating environments may vary significantly. While IBM has reviewed each item for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere.

The responsibility for use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's or user's ability to evaluate and integrate them into their operating environment. Customers or users attempting to adapt these techniques to their own environments do so at their own risk. IN NO EVENT SHALL IBM BE LIABLE FOR ANY DAMAGE ARISING FROM THE USE OF THIS INFORMATION, INCLUDING BUT NOT LIMITED TO,
LOSS OF DATA, BUSINESS INTERRUPTION, LOSS OF PROFIT OR LOSS OF OPPORTUNITY.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not necessarily tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or another claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Trademarks