DFSMS Basics: VSAM

VSAM RLS Best Practices

*How to leverage VSAM RLS best practices and performance*

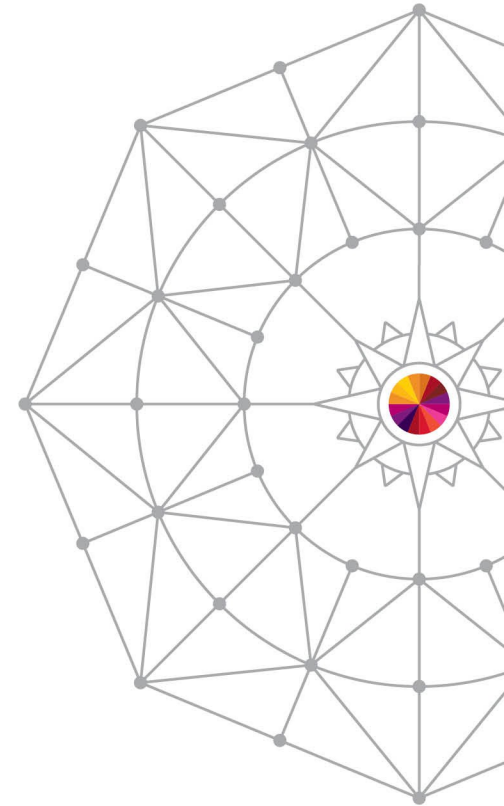*Neal Bohling, bohling@us.ibm.com*
Session : 15090

Introduction

- VSAM RLS is highly configurable, and these settings can drastically influence how SMSVSAM runs

- *Goals*:

  - Explain available configuration options
  - Give helpful guidance in setting options
  - Demonstrate useful measurements
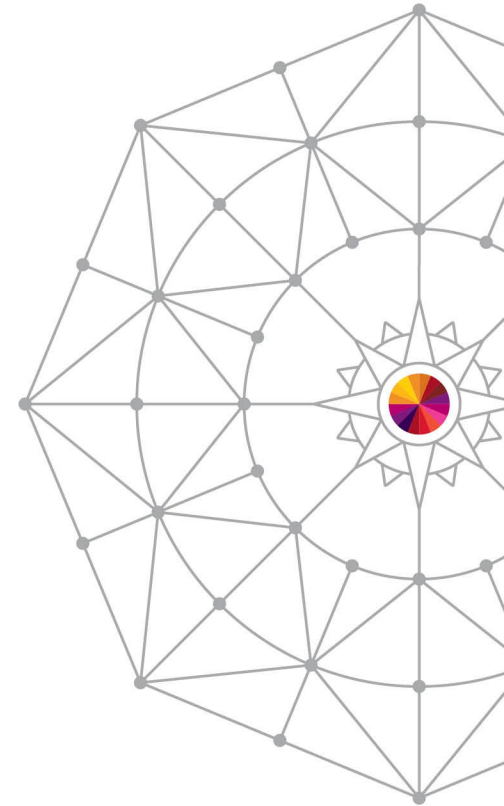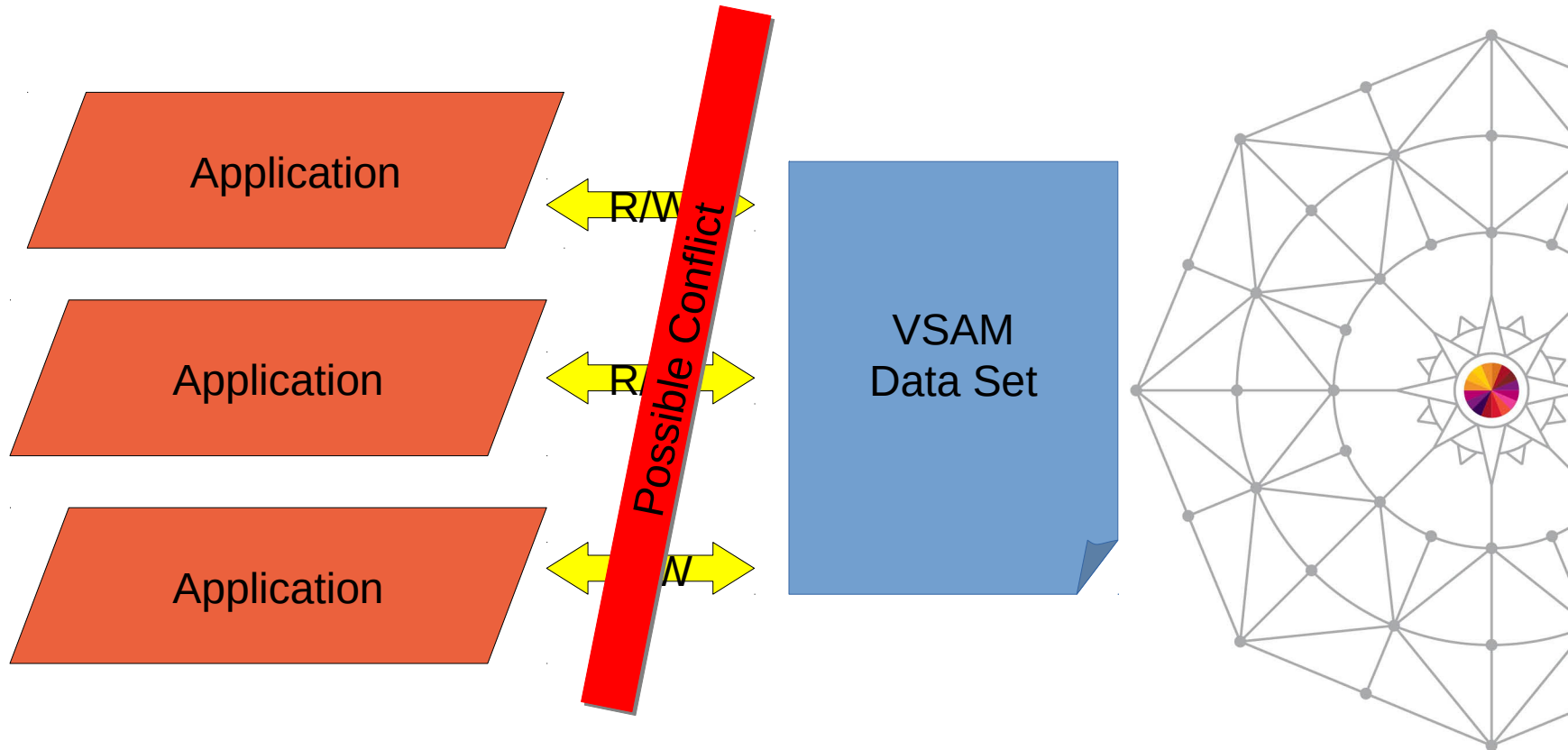 - Some guidance for troubleshooting / error reporting
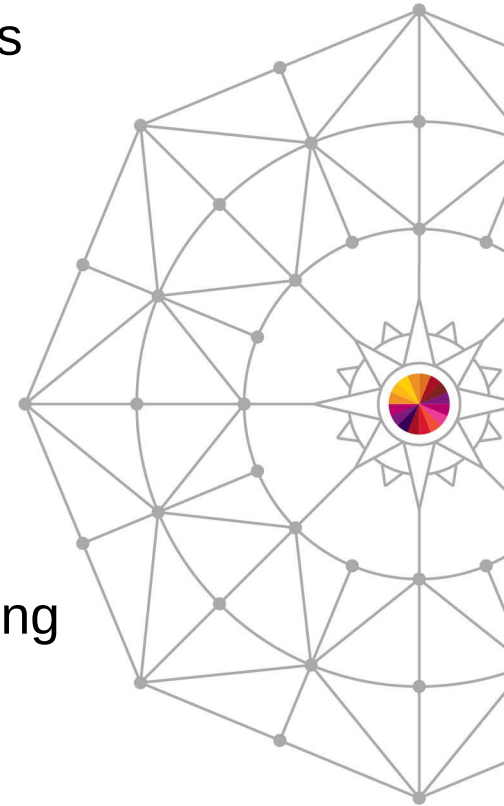
Agenda

- **Overview of RLS**
- Measurements Overview
- Locking and Structures
- Buffering
- Caching
- Data set parameters
- Request-level Parameters
- Diagnostics

RLS Overview

Regular VSAM Access

Application

R/W

Possible Conflict

VSAM
Data Set

Application

R/W

Application

R/W

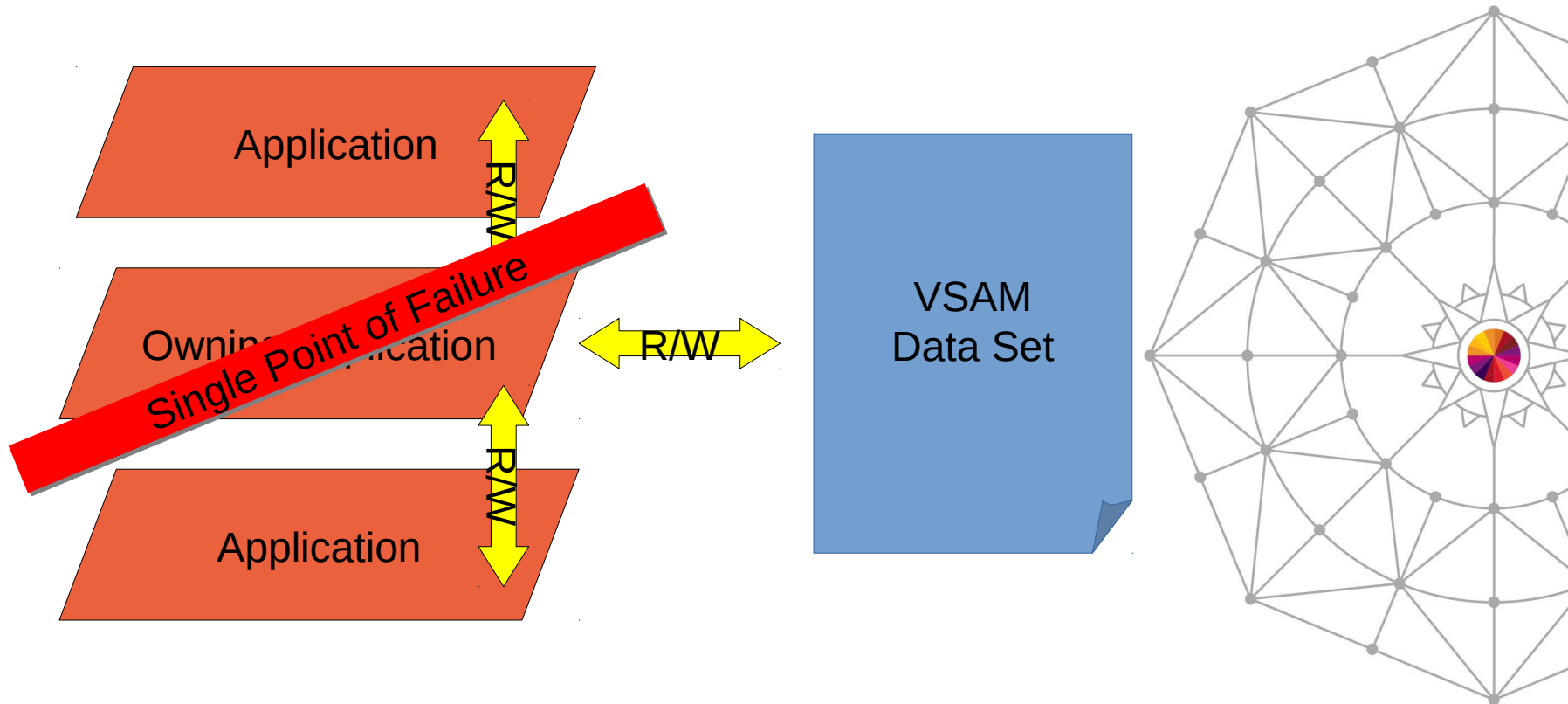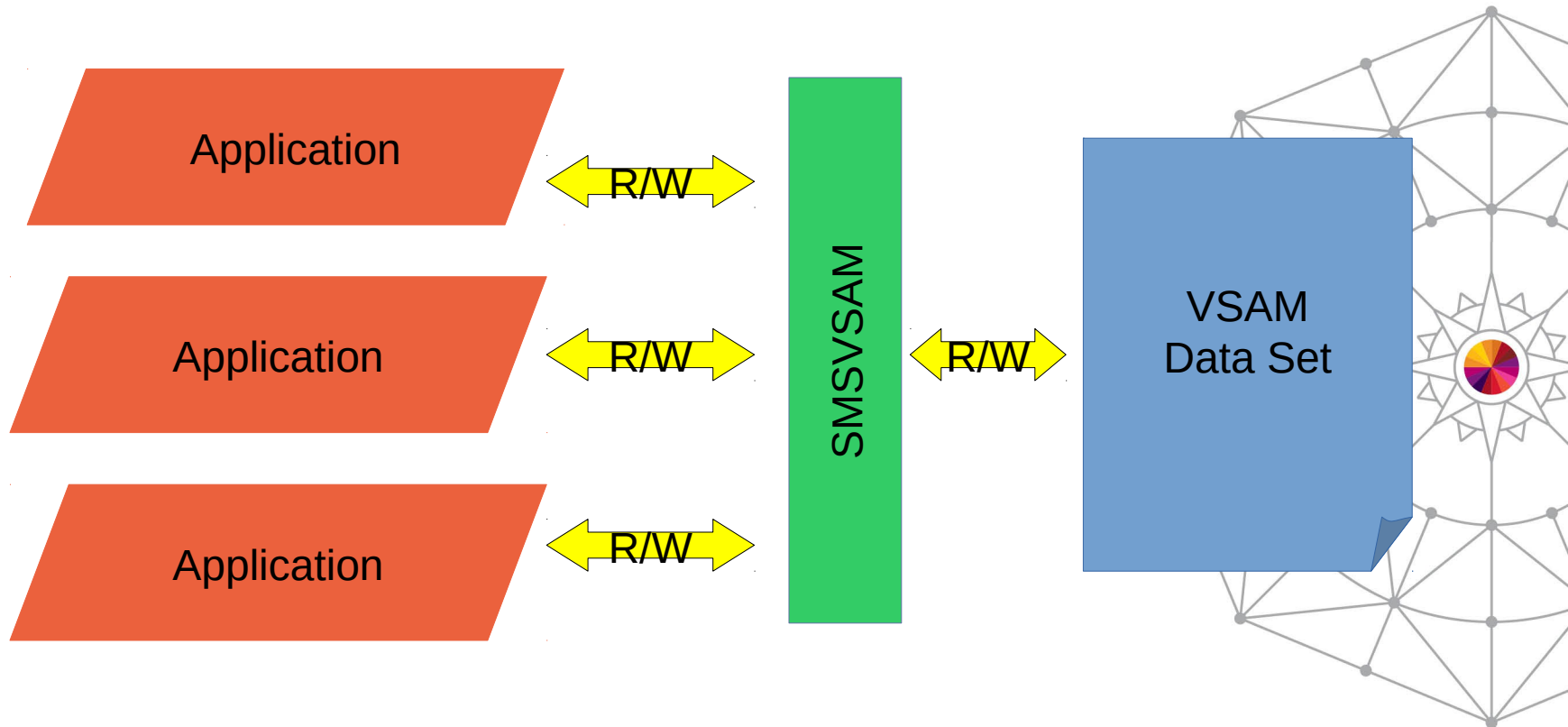*Applications are responsible for serialization (SHR >3)*
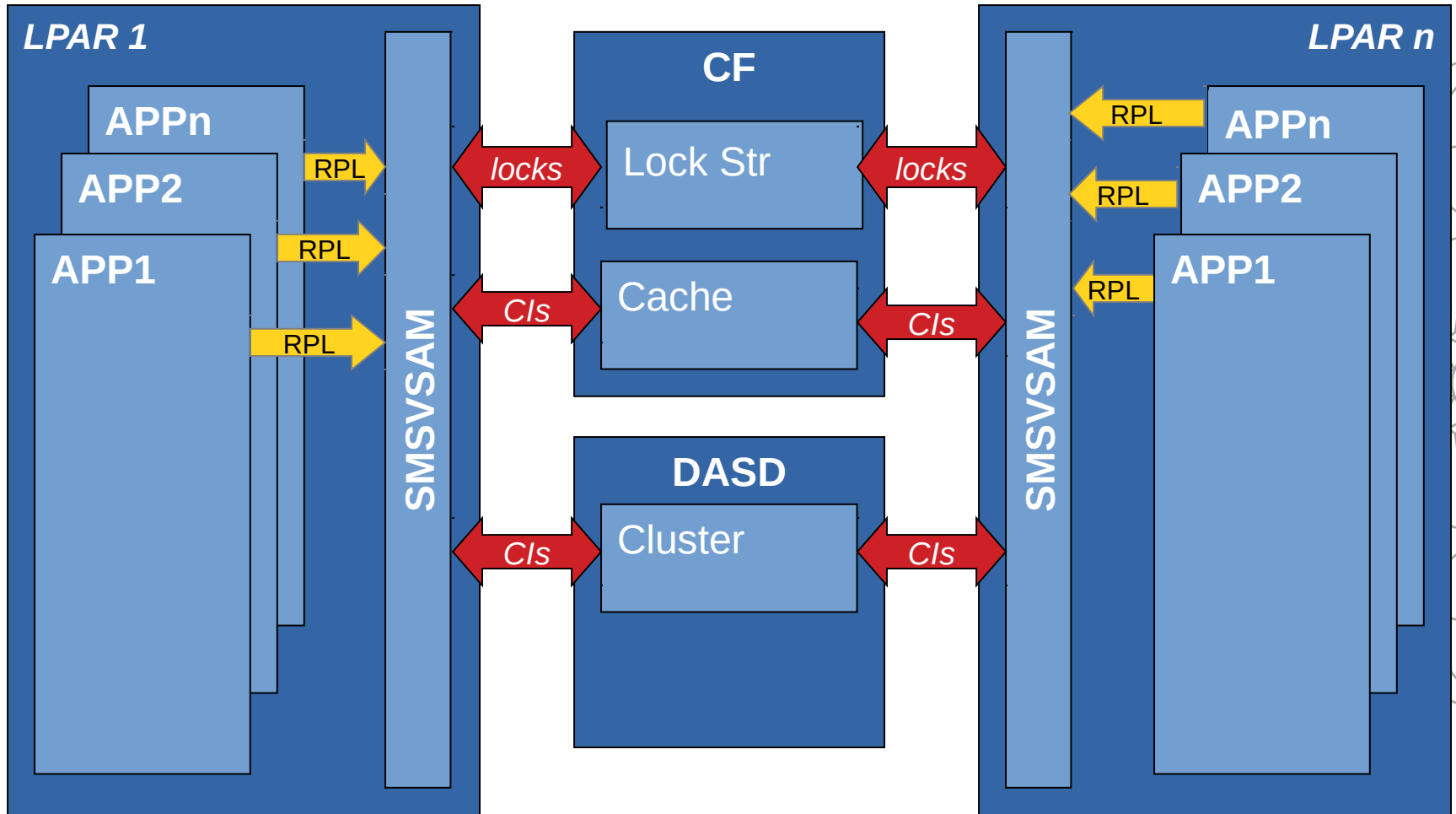
Single-owner model

Applications are responsible for serialization (SHR >3)

RLS VSAM Access

SMSVSAM is responsible for serialization.

RLS in More Detail

# VSAM RLS I/O Path – Sub Components

- VSAM Record Management (VRM)

  - Provides the VSAM interfaces:
    GET, PUT, POINT, ERASE, etc

  - Parameters passed to VRM are
    through the RPL control block

- Storage Management Locking Services (SMLS)

  - Interfaces with VRM and XCF locking services

  - Obtains, releases, and alters locks in the coupling facility
    lock structure(s)
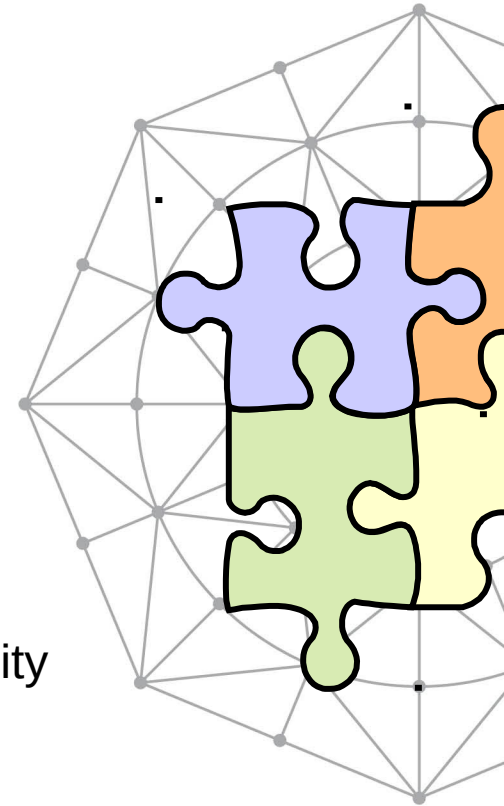
VSAM RLS I/O Path – Sub Components

- Sysplex Cache Manager (SCM)

  - Interfaces with Buffer Management Facility and XCF caching services

  - Obtains directory elements and read/write data elements in the coupling facility cache structures

- Buffer Management Facility (BMF)

  - Interfaces with VRM and SCM to locate/add buffers to the local pool

  - Supports buffering past close

    - Data sets reopened for RLS within 10 minutes can reuse valid buffers remaining in the pool

  - Manages the size of the local buffer pool via a Least Recently Used (LRU) manager
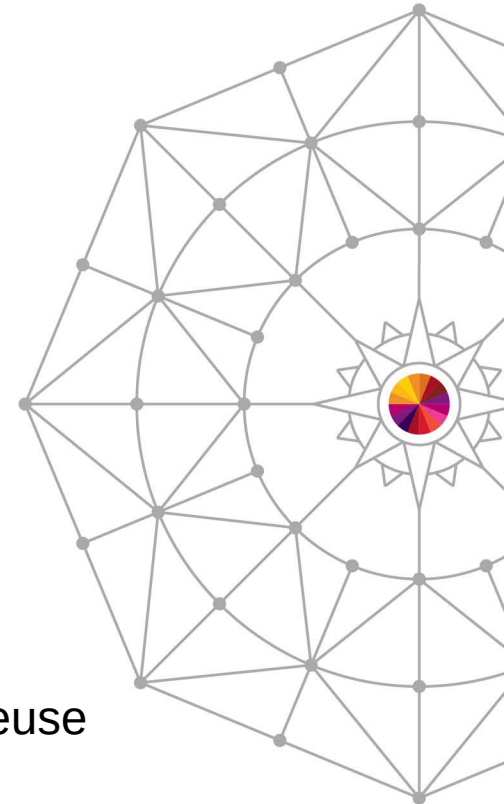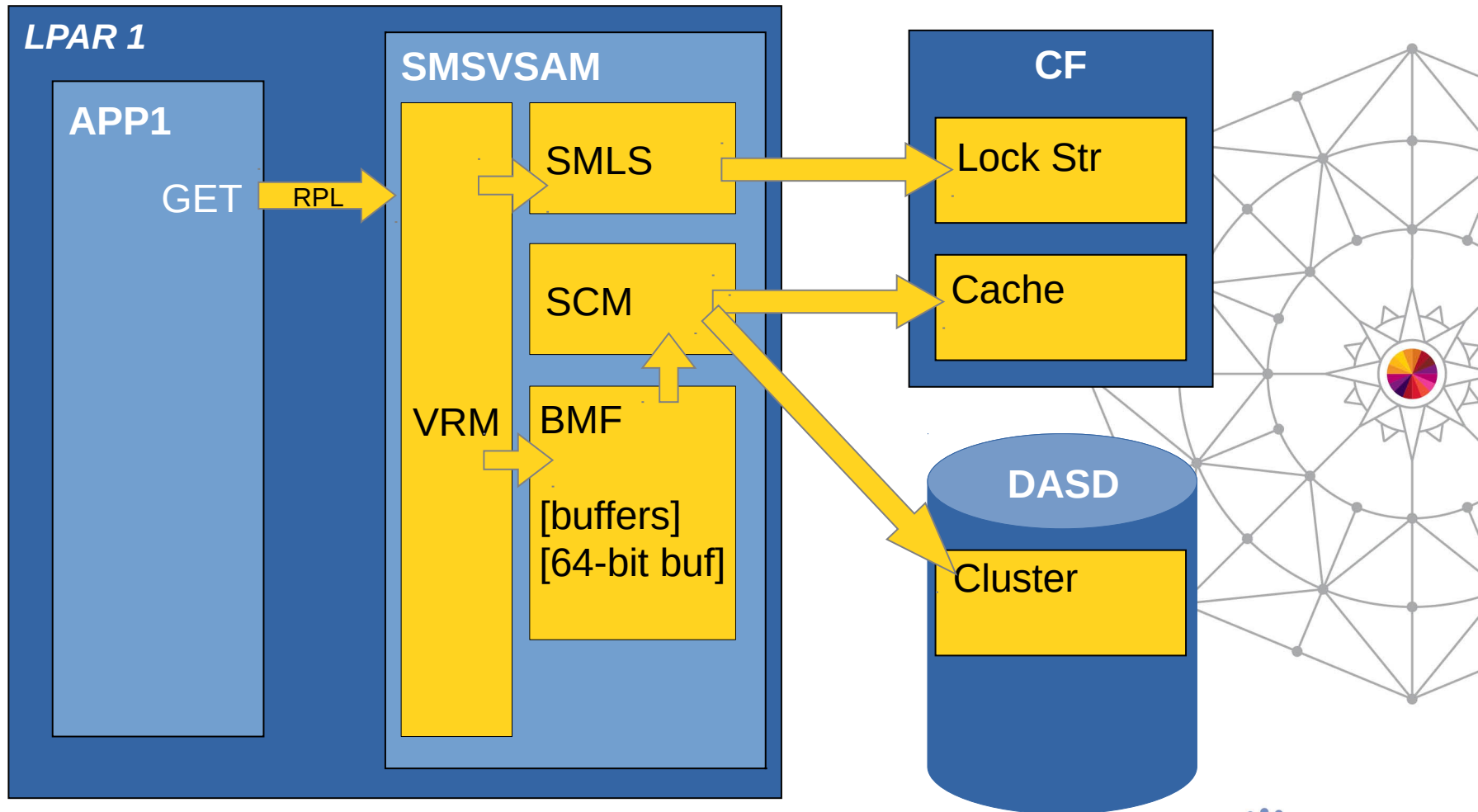
VSAM RLS I/O Path

Performance Goal: Spend the least amount of time in the I/O path!

GET/PUT ↔ VRM ↔ SMLS ↔ XCF Locking services

↔ BMF ↔ SCM ↔ XCF Caching Services

↔ Media Mgr Services (to DASD)

GET request – not buffered

LPAR 1

APP1

GET — RPL →

SMSVSAM

SMLS

SCM

VRM

BMF

[buffers]
[64-bit buf]

CF

Lock Str

Cache

DASD

Cluster

GET request – buffered
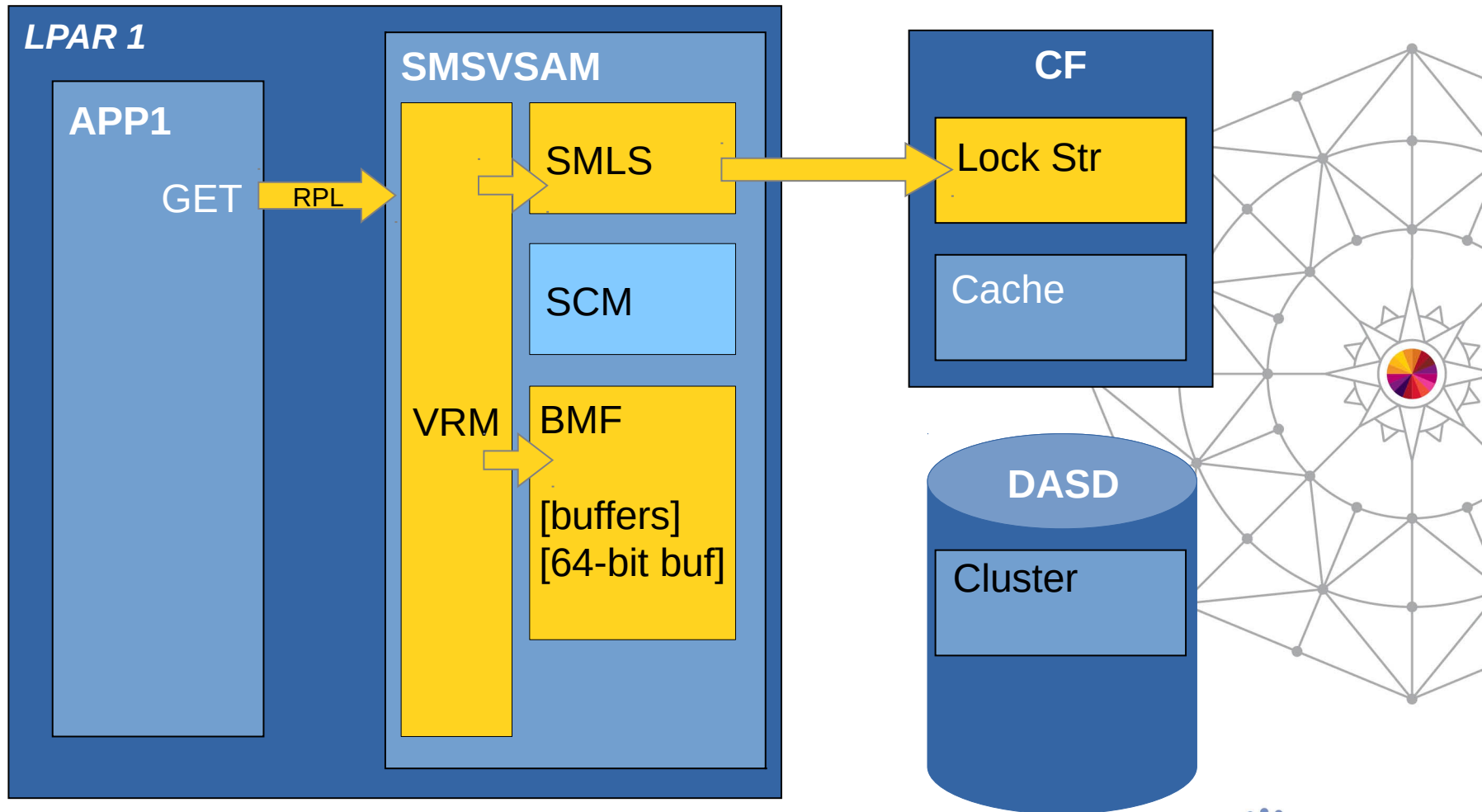
VSAM RLS I/O Path

Performance Goal: Spend as little time as possible I/O path!

GET/PUT ↔ VRM ↔ SMLS ↔ XCF Locking services
↔ BMF ↔ SCM ↔ XCF Caching Services
↔ Media Mgr Services (DASD)

Agenda

- Overview of RLS
- **Measurements Overview**
- Locking and Structures
- Buffering
- Caching
- Data set parameters
- Request-level Parameters
- Diagnostics

Measurements Overview

- SMF
  - Type 60-66 – Open / Close / Extend records
  - SMF 42:16-19 – RLS specific records
- RMF Monitor III
- Display Commands
  - RLS – CFLS, DIAG(C), QUIESCE
  - GRS
  - XCF
- Health Checker

Performance Measurements – SMF 42

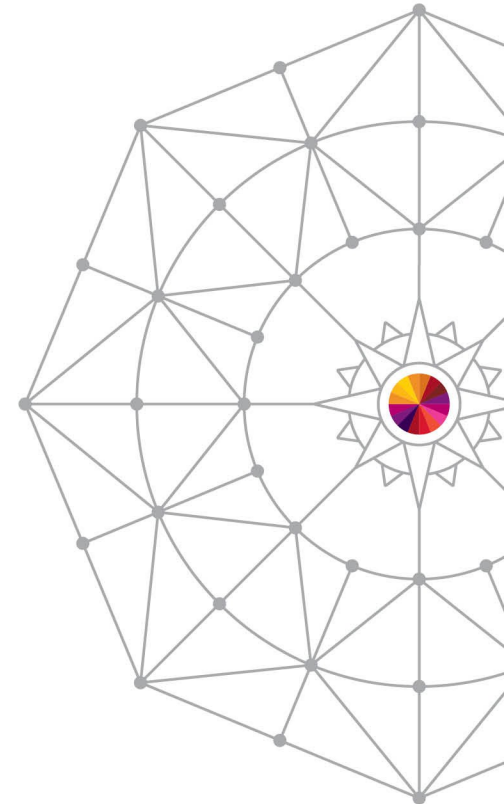- SMF 42 Subtypes

  - **Subytpe 15** - RLS statistics by storage class

  - **Subtype 16** - RLS statistics by data set*

  - **Subtype 17** - RLS CF lock structure usage

  - **Subtype 18** - RLS CF caching statistics

  - **Subtype 19** - Buffer Manager LRU statistics

- Note: Only one system in the sysplex collects the SMF 42 records. The system collecting the records is displayed in the D SMS,SMSVSAM operator command.

- * Must use V SMS,MONDS(spherename),ON to collect subtype 16 statistics

Performance Measurements – RMF

- RMF Monitor III

  - Statistics gathered from SMSVSAM

  - Mirrors and summarizes SMF42 data

  - Enabled via the VSAMRLS
    parm in ERBRMF04

  - RMF Option:
    - 3 – Monitor III
    - S – SYSPLEX

- See reference slides for notes on starting RMF
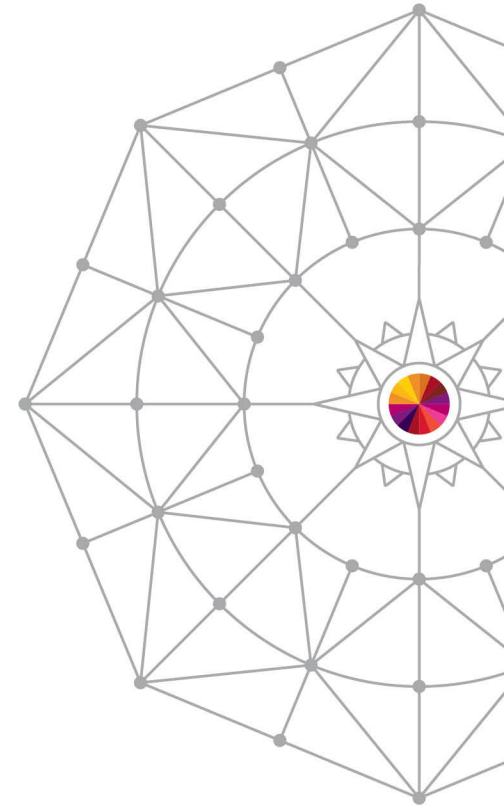
Performance Measurements – RMF

```
                    RMF Sysplex Report Selection Menu


 Enter selection number or command for desired report.


  Sysplex Reports
        1 SYSSUM    Sysplex performance summary            (SUM)
        2 SYSRTD    Response time distribution             (RTD)
        3 SYSWKM    Work Manager delays                    (WKM)
        4 SYSENQ    Sysplex-wide Enqueue delays             (ES)
        5 CFOVER    Coupling Facility overview              (CO)
        6 CFSYS     Coupling Facility systems               (CS)
        7 CFACT     Coupling Facility activity              (CA)
        8 CACHSUM   Cache summary                          (CAS)
        9 CACHDET   Cache detail                           (CAD)
       10 RLSSC     VSAM RLS activity by storage class     (RLS)
       11 RLSDS     VSAM RLS activity by data set          (RLD)
       12 RLSLRU    VSAM LRU overview                      (RLL)
```
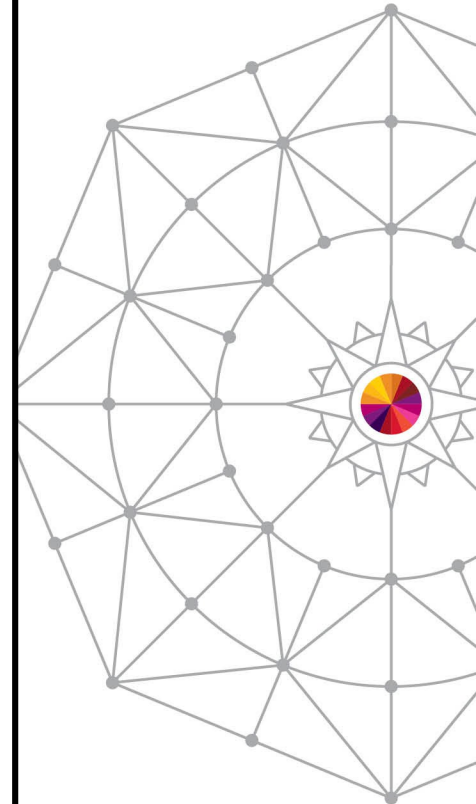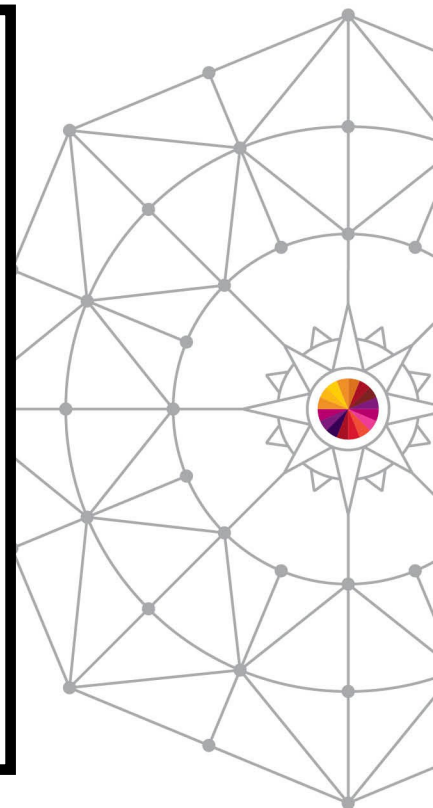
RLSSC - Sysplex Totals View

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


Samples: 120      Systems: 1     Date: 03/08/14  Time: 17.28.00  Range: 120    Sec


             < 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
RLS
  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  2303  99.4   0.0    0.6    100      0.00      1158
             SEQ   0.000  0.00   0.0   0.0    0.0    0.0      0.00      0.00
```
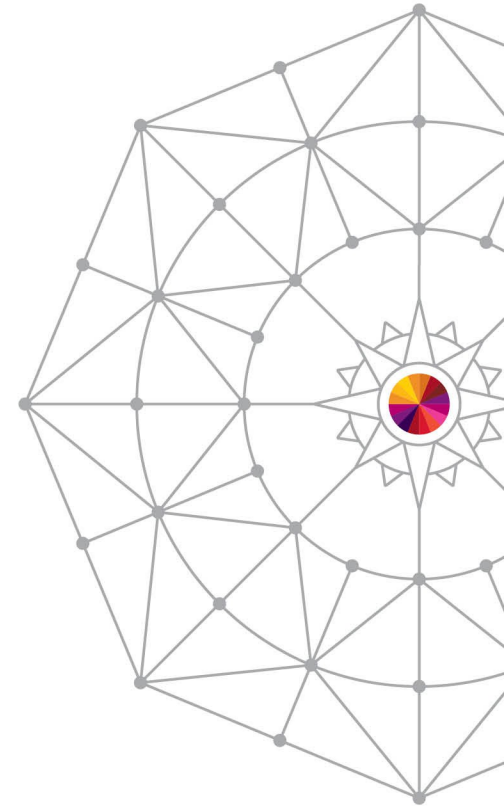
Health Checker

- z/OS Tool introduced in V1R4,
- SMSVSAM checks added starting in V1R7
- Periodically runs "checks" against system to watch for errors and attempt to prevent problems
- Checks **IBMVSAMRLS**
  - VSAMRLS_CFCACHE_MINIMUM_SIZE
  - VSAMRLS_CFLS_FALSE_CONTENTION
  - VSAMRLS_DIAG_CONTENTION
  - VSAMRLS_QUIESCE_STATUS
  - VSAMRLS_SHCDS_CONSISTENCY
  - VSAMRLS_SHCDS_MINIMUM_SIZE
  - VSAMRLS_SINGLE_POINT_FAILURE
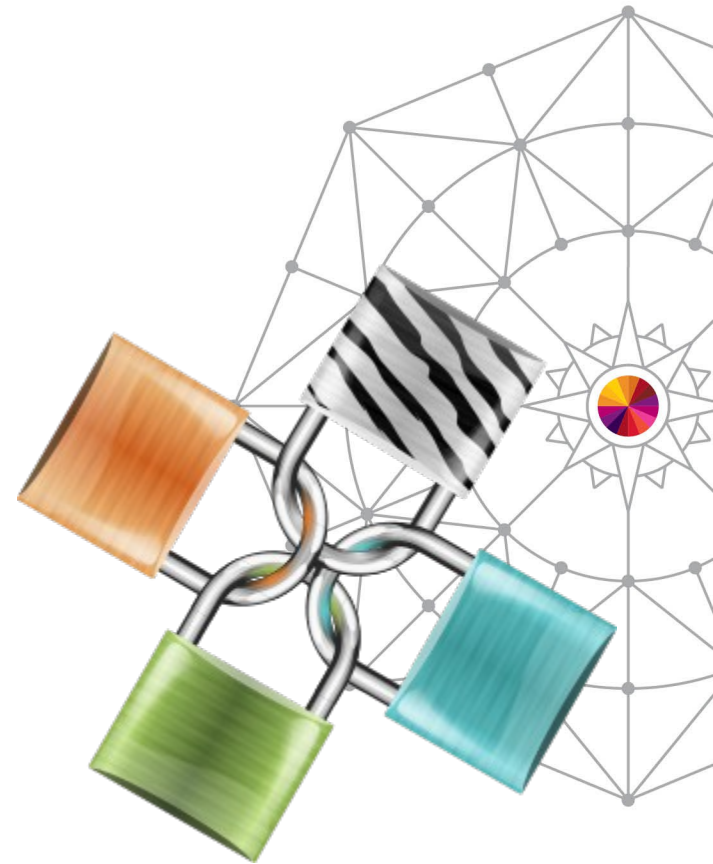 - VSAMRLS_TVS_ENABLED

Agenda

- Overview of RLS
- Measurements Overview
- **Locking and Structures**
- Buffering
- Caching
- Data set parameters
- Request-level Parameters
- Diagnostics

Lock Structures

- Locks
  - Records locks
  - Special locks
  - Global RLS parameters

- Lock Structures
  - Stores hash table of locks
  - Defined via CFRM Policy
  - Stored in Coupling Facility
  - Named IGWLOCK00

Lock Structures - Sizing

- **Size requirement depends on how many locks are active**

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

  - 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/
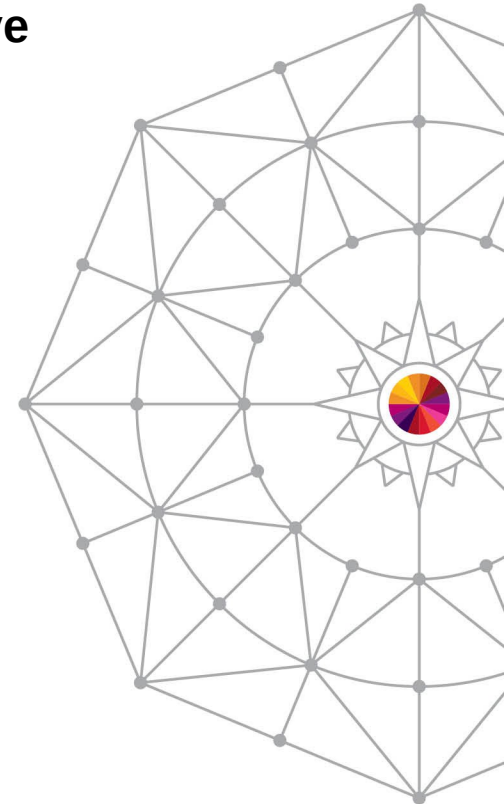
Lock Structures – Potential Problems

- **Running out of Lock Space**

  - IGW326W *** Warning *** DFSMS SMSVSAM RECORD TABLE IN IGWLOCK00 100% FULL.
  - Can cause abends and failed requests
    - RPLFDBK 090800BD

- **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**

  - HealthCheck
  - D SMS,CFLS command
  - RMF CF activity report
  - SMF 42 subtype 17

Health Check

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

- VSAMRLS_CFLS_FALSE_CONTENTION
  - Average of the last hour
  - Samples taken every second
  - Has configurable threshold
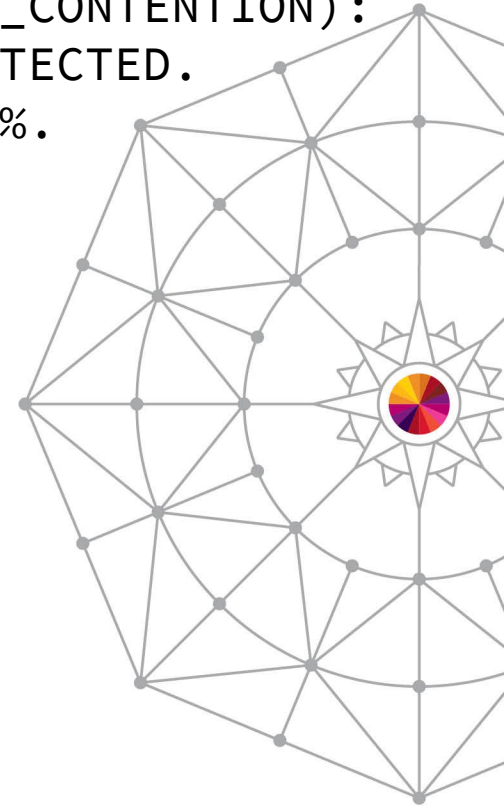
D SMS,CFLS

```
IGW320I 11:37:11 Display SMS,CFLS(IGWLOCK00 )
PRIMARY STRUCTURE:IGWLOCK00 VERSION:CCC6425A4759D195 SIZE:4096K
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
```
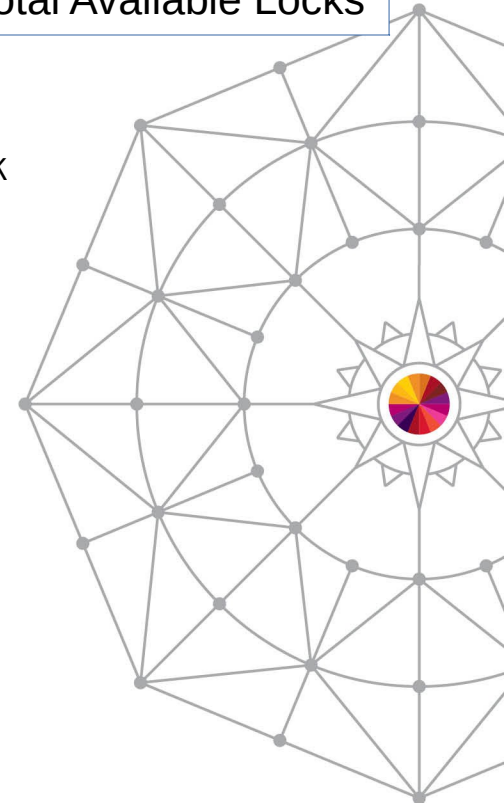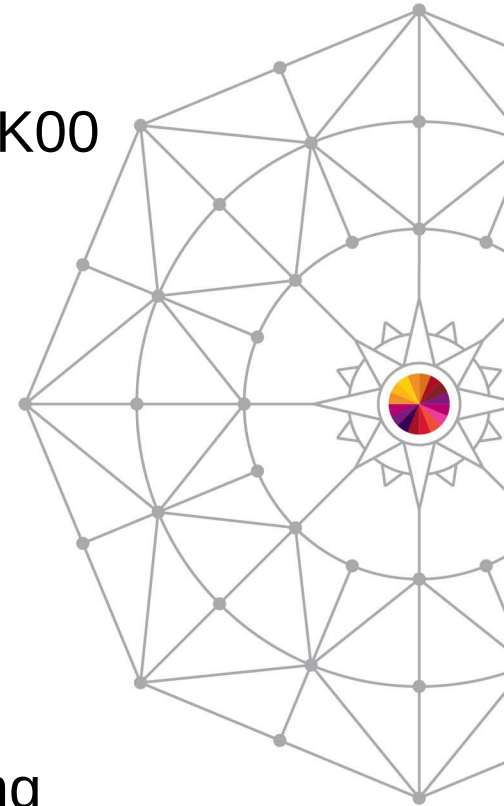
False Contention - Measuring

- Other ways to measure:
  - RMF Mon III – Structure details on IGWLOCK00
    - S.7 - "Coupling Facility Activity"
  - RMF Mon III – RLSSC and RLSDS
    - `                 < 2GB / > 2GB`
    - `LRU Status   : Good  / Good`
    - `Contention % :   0.0 /   0.0`
    - `False Cont % :   0.0 /   0.0`

  - SMF 42 subtypes 15,16,17
    - SMF42FOA – Number of Lock Requests
    - SMF42FOC – Number of Lock Req causing False Contention

Multiple Lock Structures

- Allows separation of workloads
- SMS allows up to 256 different lock sets
- Can use any name: IGWLOCKxx is easy
- Storage Class →
    SMS Lock Set →
        Lock Structure
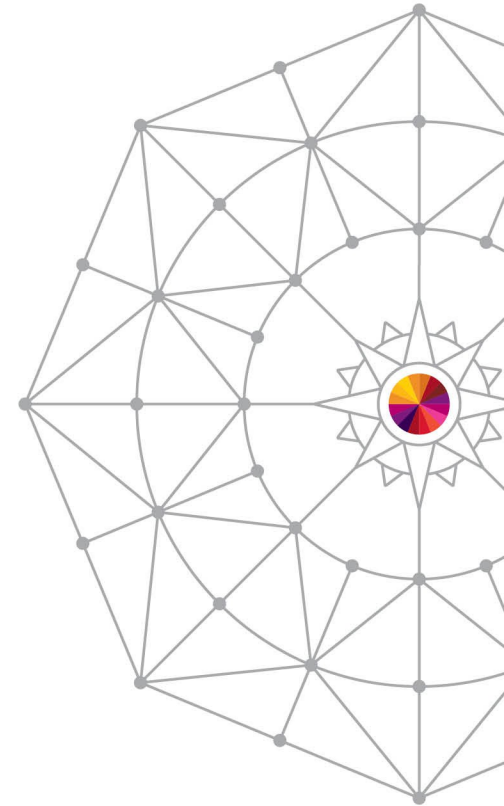- Does not include special locks (such as CI split lock)

Storage Class 1

Lock Set 1

Lock Structure 1

Storage Class 2

Lock Set 2

Lock Structure 2

Storage Class 3
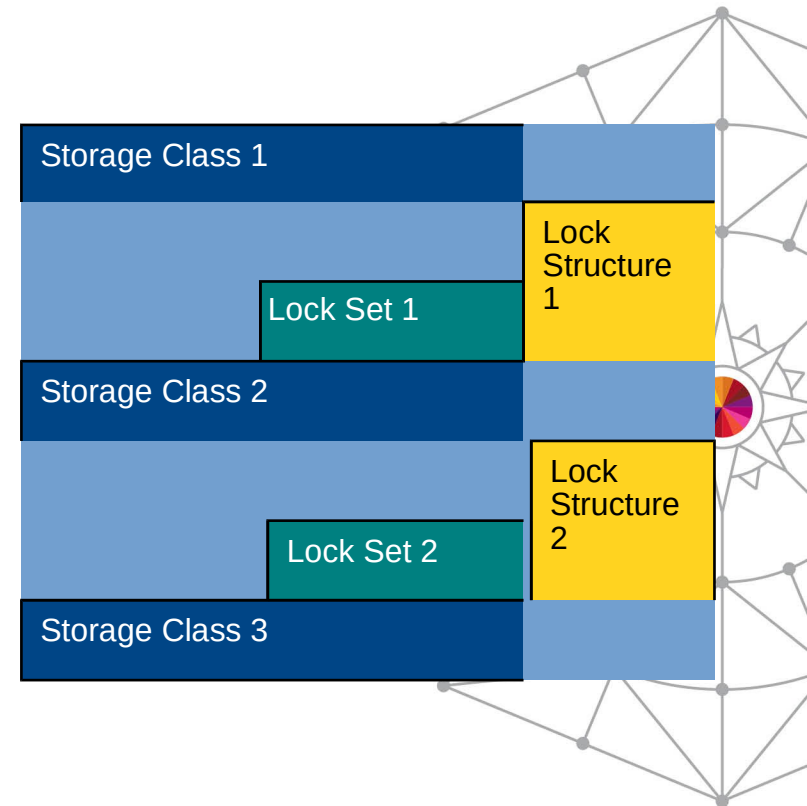
Locking Summary

- Lock size depends on active locks
- Recommended size is > 13mb,
  at least 20mb / system

- **Best Practice:** Watch False Contention to see
  if you need to increase lock structure size

- Multiple Lock Structures good for keeping
  workload lock activity separate

Agenda

- Overview of RLS
- Measurements Overview
- Locking and Structures
- **Buffering**
- Caching
- Data set parameters
- Request-level Parameters
- Diagnostics

Buffer Management Overview

- Data buffered at the CI level

- All RLS data is buffered

- As buffers age, LRU routine removes them

- Two possible pools:
  - 31 Bit Buffer Pool
    - Size: 10M to 1728M
    - Defined using the RLS_MAX_POOL_SIZE parameter
    - Same on all systems
  - 64 Bit Buffer Pool
    - Size: 0 or > 500MB
    - Defined using RLSAboveTheBarMaxPoolSize
    - Can be specified per system
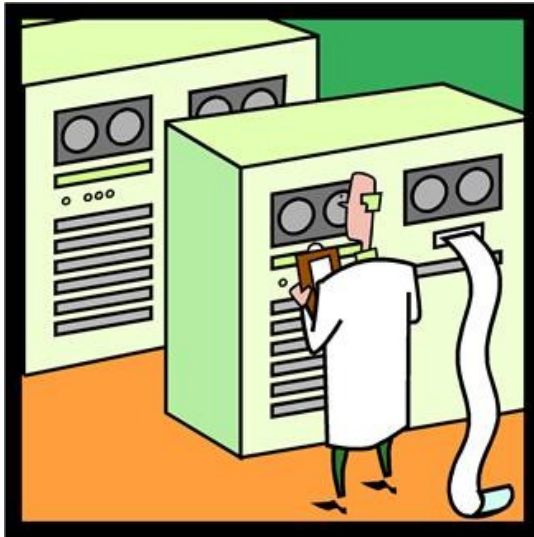
Below (2GB)
Maximum 1728

LRU Modes:
Accel: 120%
Panic: 200%

Above (> 2GB)
Limited by
available storage

LRU Modes:
Accel: 90%
Panic: 100%
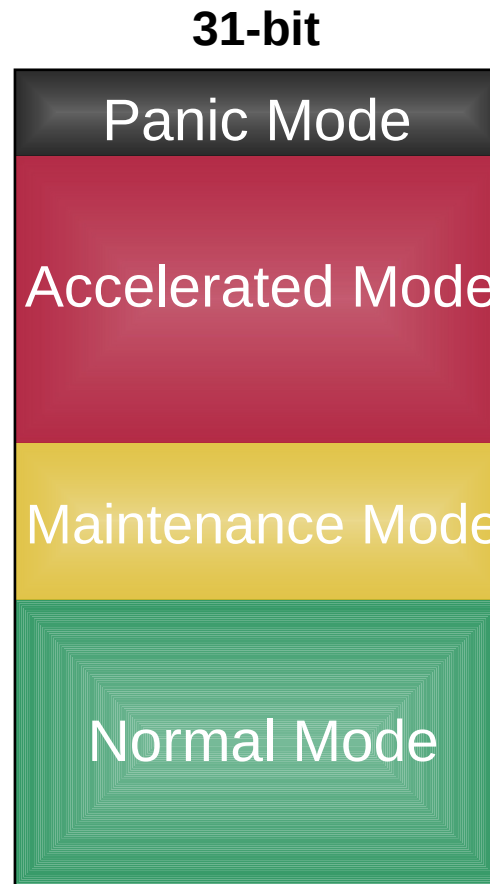
Performance and Tuning – Buffer Pools

- Pool Size values are a *goal*, not a limit

- If more buffers are required, the pool will exceed the goal

- Total size of buffer pools should not exceed amount of real storage.

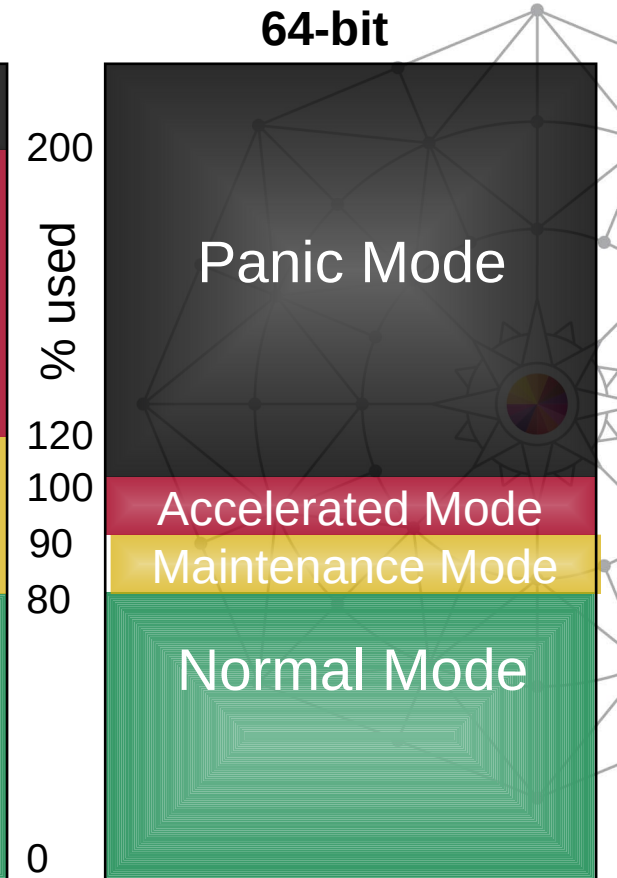- A paged out buffer is immediately freed by the LRU.

SMSVSAM LRU

- Runs every 15 seconds
- Throws out buffers
- **PANIC MODE**
  - Runs every 5 seconds
  - 31-bit threshold:
    240 – (panic cycles * 8)
  - 64-bit threshold: 5 min
  - Use stolen buffers
- Accelerated Mode
  - 31-bit threshold:
    240 – (accel cycles * 4)
  - 64-bit treshold: 30min
  - Buffer stealing starts
- Maintenance Mode
 - 32-bit: 240-(maint cycles)
  - 64-bit: 60min
- Normal Mode
  - 60 min

31-bit

Panic Mode

Accelerated Mode

Maintenance Mode

Normal Mode

RLS_MAX_POOL_SIZE

64-bit

Panic Mode

Accelerated Mode

Maintenance Mode

Normal Mode

RLSABOVETHEBARMAXPOOLSIZE

% used

200

120

100

90

80

0

Buffer Sizing Considerations

- **Consider**:
  - How much of your data are you going to want in-buffer?
  - How long do you want it there / how often is it referenced?
  - How much data do you have flowing through SMSVSAM?
  - How fast does it flow?

- **Buffers will be thrown out when one of the following is true:**
  - Over 60 minutes old (in normal mode)
  - Invalidated from the cache
  - Increased LRU mode causes them to be pitched sooner

- **No easy way to calculate a goal**
  - Manual states no smaller than 150% of combined non-RLS buffers
  - Needs to hold relevant data for a relevant time period
 - You want to avoid loading more than will fit in the buffer within 60 minutes
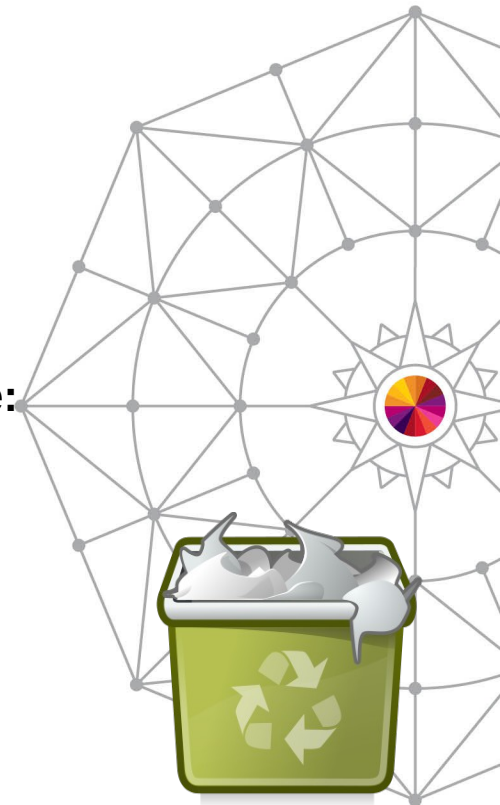
Specifying Buffer Pool Size

- ## SYS1.PARMLIB(IGDSMSxx)

  - **RLS_MAX_POOL_SIZE(nnnn)**
    - Maximum size of SMSVSAM local buffer pool
    - nnnn = 10 to 9999, with over 1500 treated as 1728
    - **Recommendation: RLS_MAX_POOL_SIZE(<= 850)**

  - **RLSAboveTheBarMaxPoolSize([sys1,val1;sys2,val2...] | [ALL,value])**
    - Amount of virtual storage allocated above 2G bar for RLS buffering
    - nnnn = 0, or 500M > size > 2,000,000M (2 TB)
    - **Recommendation: RLSAboveTheBarMaxPoolSize(<= 32768)**
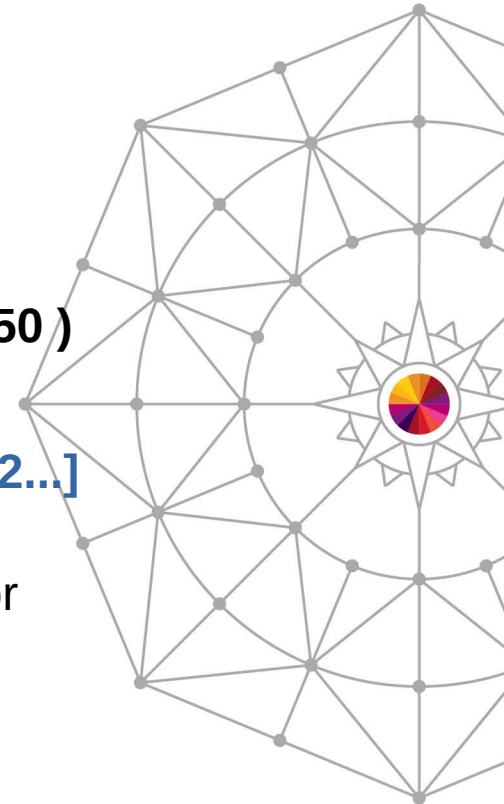
Performance and Tuning – Data Class

- **RLS Above the 2-GB Bar (YES | NO)**

    - Specifies whether SMSVSAM can use 64-bit virtual buffering for the data set

    - Must specify a non-zero RLSAboveTheBarMaxPoolSize
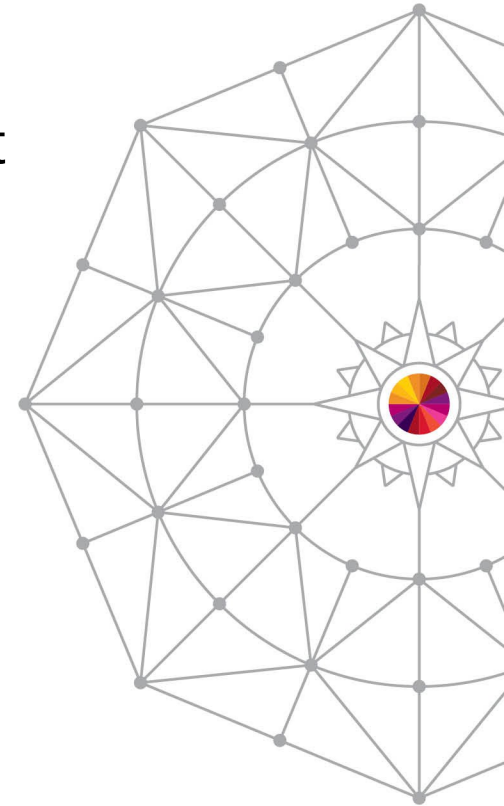
Monitoring Buffers

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

- ***Watch for:***
  - Accelerated or panic mode
  - Buffer usage vs buffer goal
  - Reads satisfied by BMF %

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
  - 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             Avg CPU  - Buffer Size -  Accel   Reclaim  ------ Read -----
System          Time      Goal   High      %        %      BMF%   CF%  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
RLS Workspaces
Buffer LRU System Detail Workspace
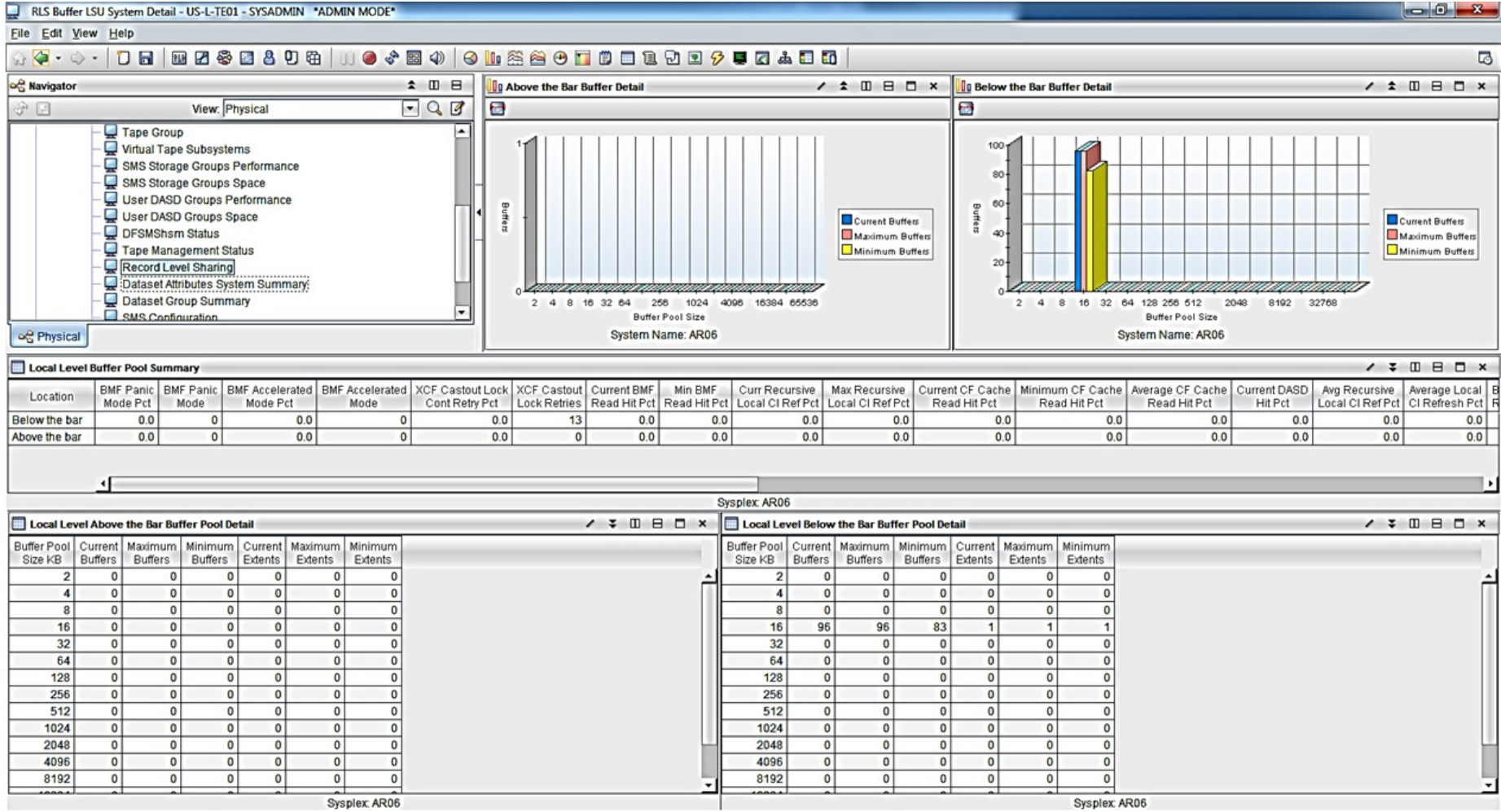
Buffer Pool Statistics for a single z/OS image

Buffering Summary

- Use adequate buffers to hold your data as long as it is needed
- Above-the-bar is available for large pools

- **Best Practice: Monitor buffers:**
  - Ensure LRU is OK
  - Watch for % BMF
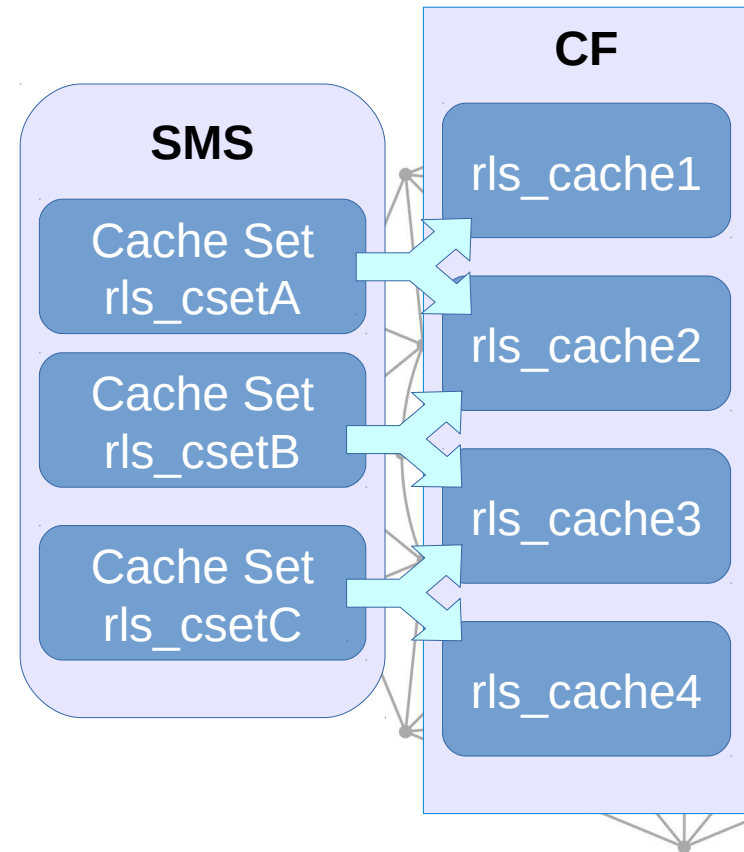 - Watch used vs goal

- *Make changes as needed*
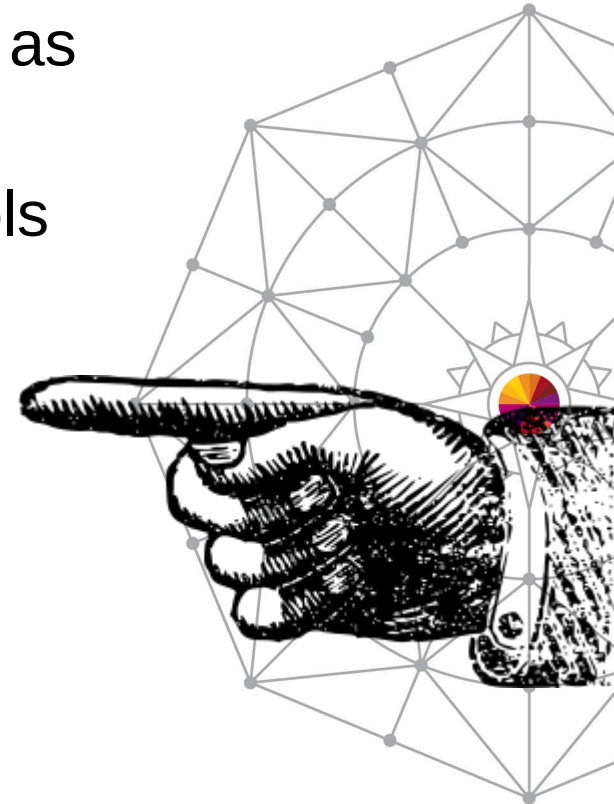
Agenda

- Overview of RLS
- Measurements Overview
- Locking and Structures
- Buffering
- **Caching**
- Data set parameters
- Request-level Parameters
- Diagnostics

Cache Structures

- Caches are:
  - Structures in the Coupling Facility
  - SMSVSAM uses them to store Cis for other systems to read
- Cache Sets are:
  - SMS constructs inside base configuration
  - Allows for partitioning of workloads and data sets
  - One cache set points to multiple caches
  - One cache set assigned to each storage class
- Proper cache structure sizing is required to ensure maximum buffer effectiveness!
- Caching at the CI level

CF

SMS

Cache Set rls_csetA

Cache Set rls_csetB

Cache Set rls_csetC

rls_cache1

rls_cache2

rls_cache3

rls_cache4

Cache Settings

- RLS_MaxCFFeatureLevel(A | Z)
  - Set in IGDSMSxx
  - A – caches up to 32k CIs
  - Z – caches up to 4k CIs
- RLS CF Cache Value
  - In the Data Class
  - Only valid if using mode A
  - Possible Values:
    - ALL – cache everything
    - NONE – cache only the index
    - UPDATESONLY – only WRITEs are cached
   - DIRONLY – Cache nothing, but use the cache as a way to track interest in CIs
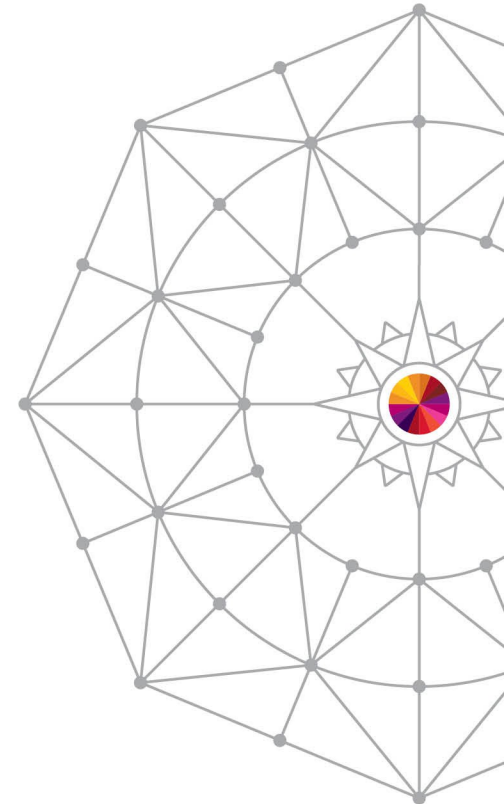
Cache Sizing

- Total cache size should be the size of all your shared data

- For best results, use Total Cache = Total Buffers
  - Sum of 31 bit and 64 bit buffer goals
    - 31 bit buffer goal =
      *RLS_Max_Pool_Size * Number of Systems)*

    - 64 bit buffer goal =
      (RLSAboveTheBarMaxPoolSize(system1) +
      RLSAboveTheBarMaxPoolSize(system2) + … +
      RLSAboveTheBarMaxPoolSize(systemN))

  - Assumptions:
    - RLS_MaxCFFeaturelevel(A): caching all data
    - No sharing of data across the sysplex
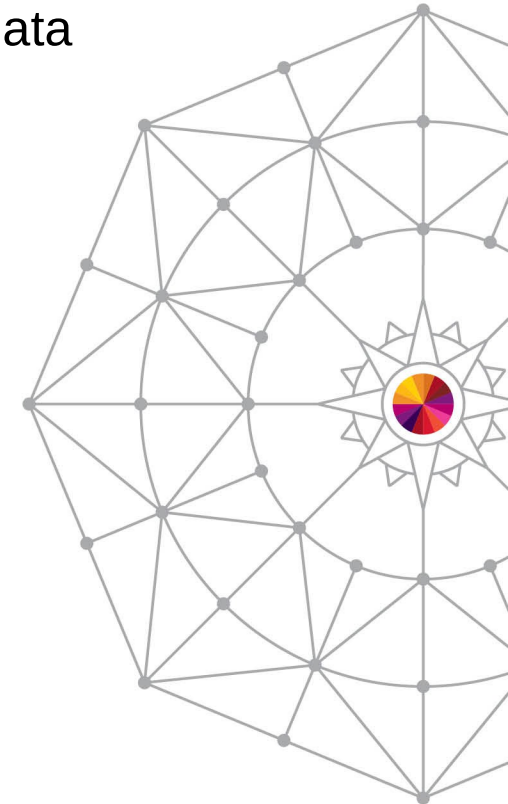
Caching

- Example:

  - RLS_Max_Pool_Size(850)

  - Number of Systems = 2

  - RLSAboveTheBarMaxPoolSize(System1,2048)

  - RLSAboveTheBarMaxPoolSize(System2,4096)

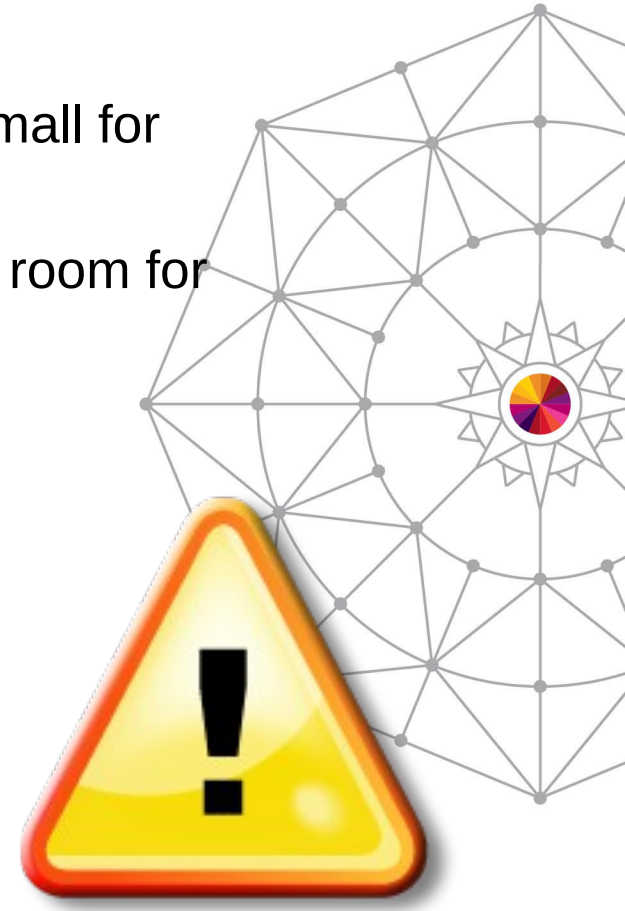 - **Cache Size = (850*2) + 2048 + 4096 = 7844M**
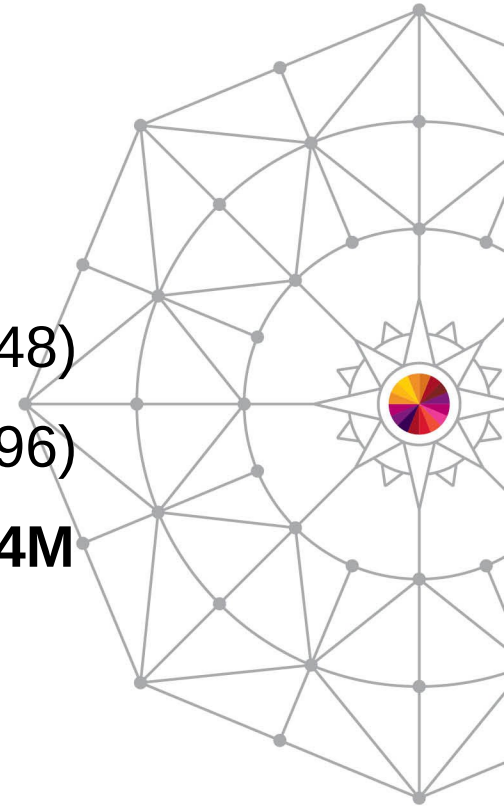
Caching

- Recommendations
  - Allocate different cache structures for different applications.
  - Allocate different structures to match internal structure usages.

- If using more than one cache structure:
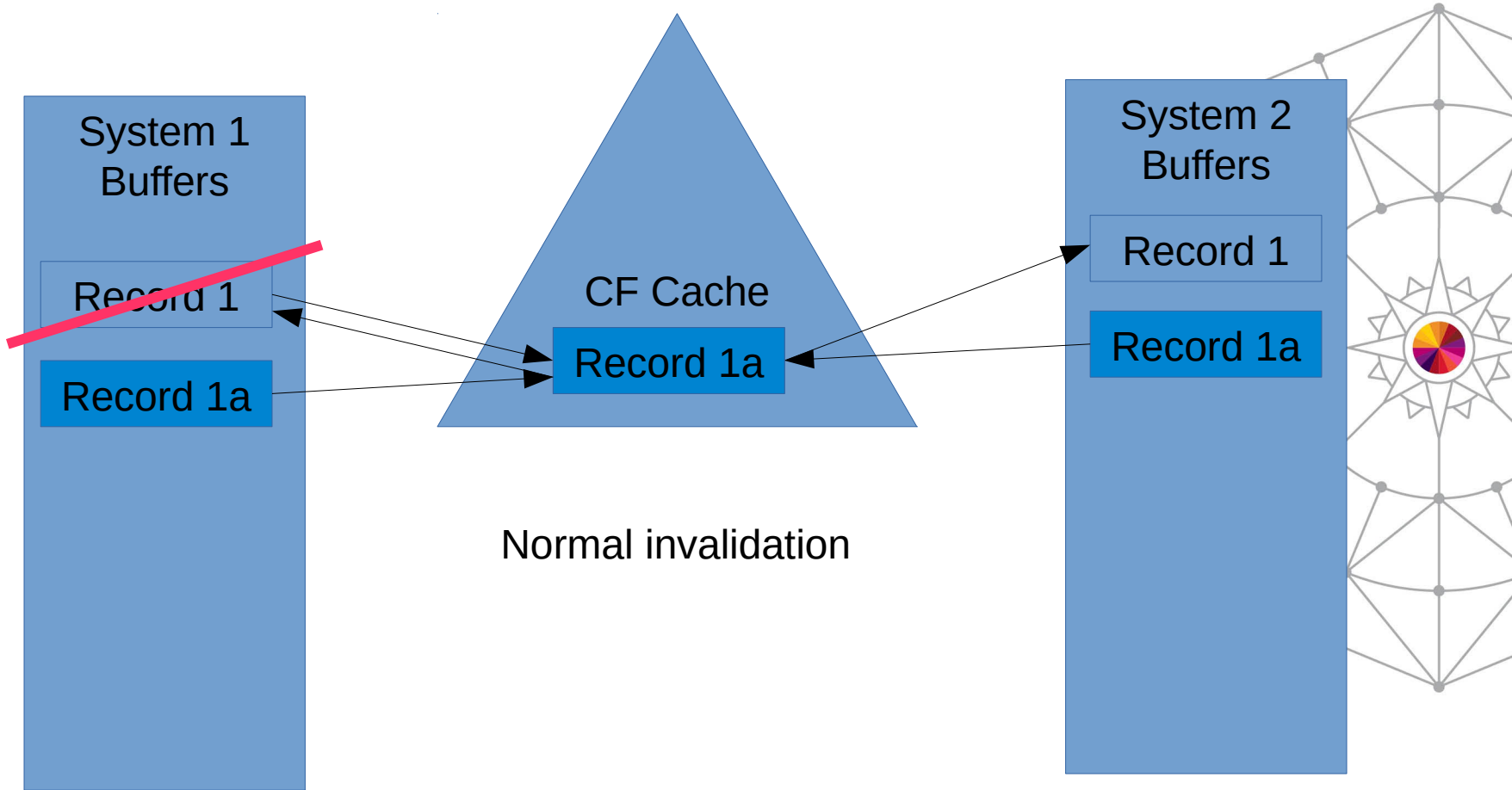 - Divide total space between caches
  - Base sizes on data set usage distribution
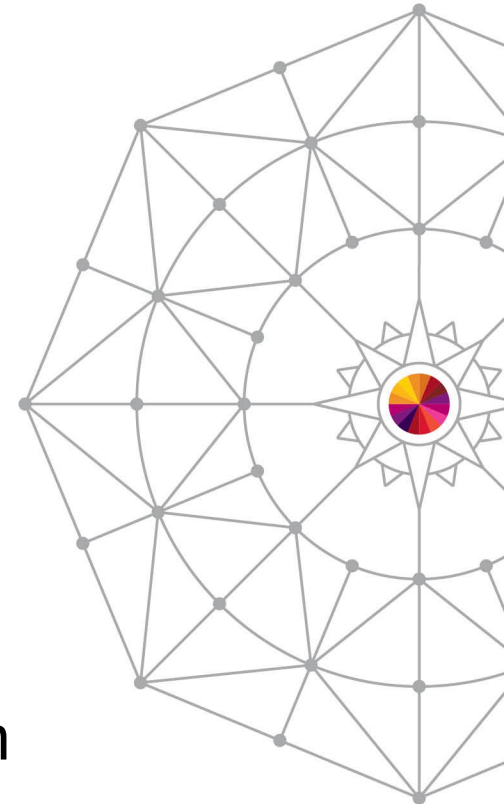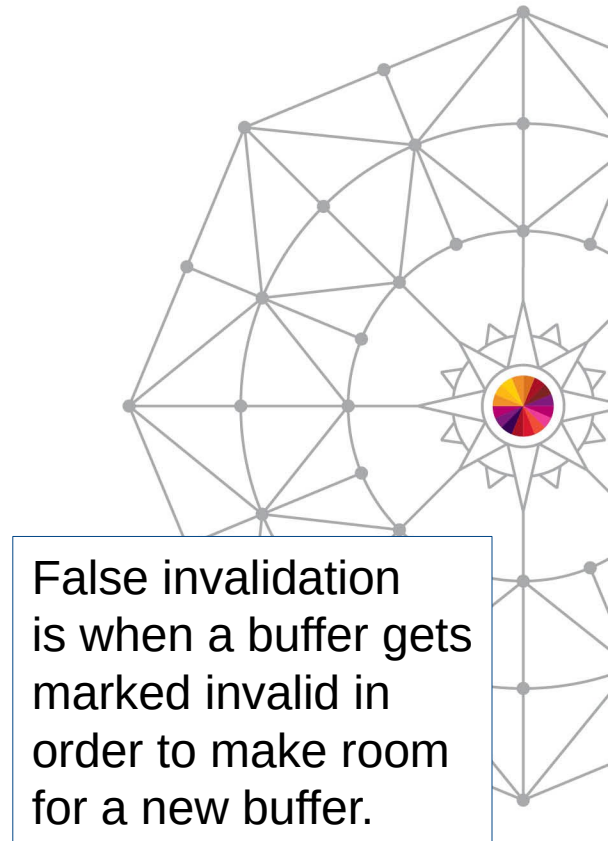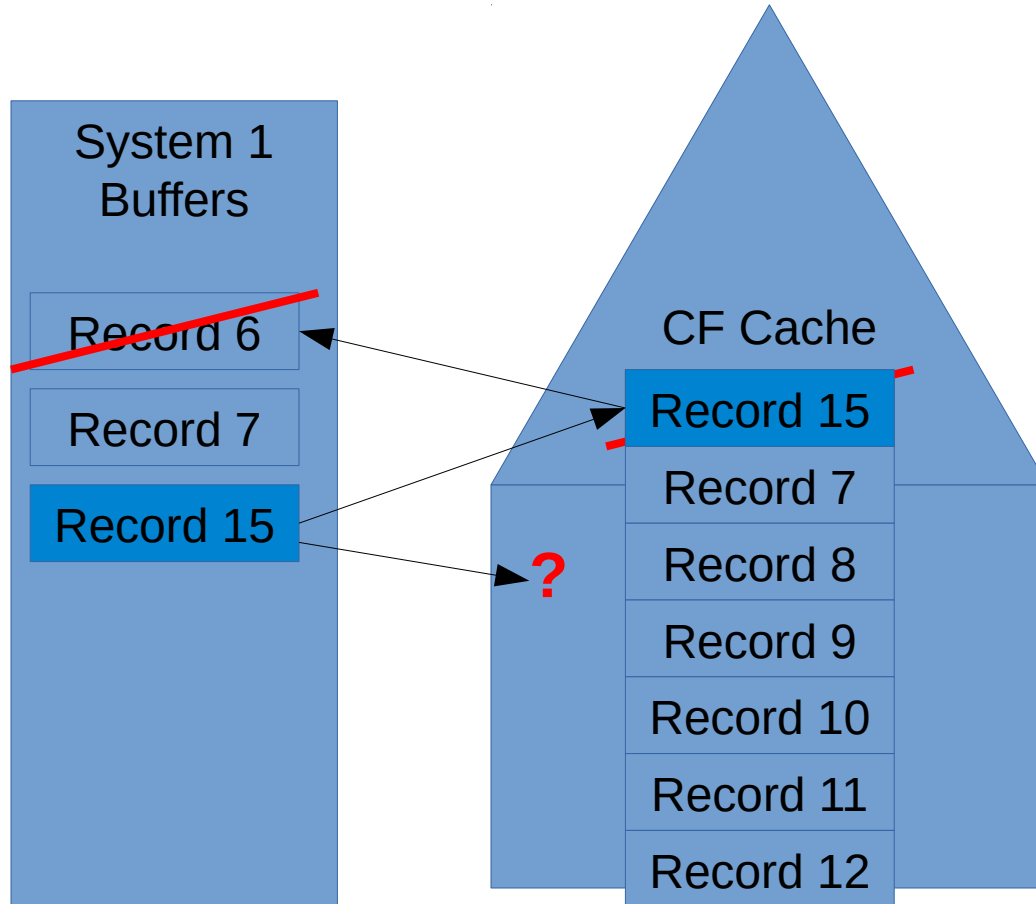
Cache Structures – False Invalids

- False Invalids:

  - Take place when a cache structure(s) too small for current buffer allocations

  - Cache structures invalidate entries to make room for new requests

  - Causes additional requests to DASD!

  - Tracked in

    - RMF panels RLSSC, and RLSDS
    - SMF 42 subtypes 15-16

Normal Invalidation Example

System 1
Buffers

Record 1

Record 1a

CF Cache

Record 1a

System 2
Buffers

Record 1

Record 1a

Normal invalidation

False Invalidation Example

System 1 Buffers

- Record 6 ~~(crossed out)~~
- Record 7
- Record 15

CF Cache

- Record 15
- Record 7
- Record 8
- Record 9
- Record 10
- Record 11
- Record 12

?

False invalidation is when a buffer gets marked invalid in order to make room for a new buffer.

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
RLS
  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  99.4   0.0   0.6    100     0.00      1137
             SEQ    0.000   0.00   0.0   0.0   0.0    0.0     0.00      0.00
```
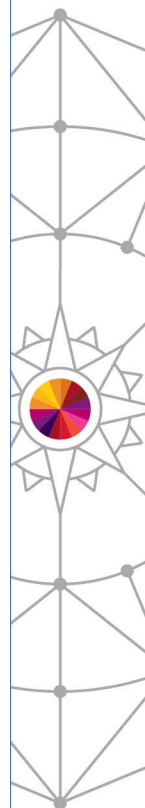
Cache Health Check

- **VSAMRLS_CFCACHE_MINIMUM_SIZE**
  - Verifies that caches are at least
    10% of optimal size
  - Calculated by adding up buffer pools
  - Can cause confusion if caches are not connected
- Messages:
  - `IGWRH0121I` The VSAM RLS CF cache size does not meet IBM's recommended minimum CF cache size for the current sysplex buffer pool configuration.
  - `IGWRH0122I` The IBM recommended minimum cache size for your current sysplex buffer pool configuration is CacheMinSize MB.
    According to the current configuration of the VSAM RLS buffer pool in the sysplex, the VSAM RLS CF cache optimal size should be CacheOptSize MB. CF cache size should be at least CacheMinPercent% of the CF cache optimal size. Thus, the CF cache size for the sysplex should be at least CacheMinSize MB.
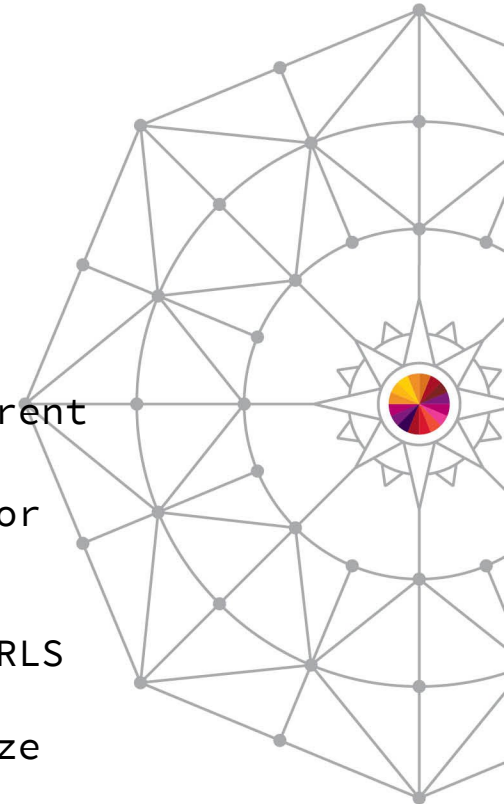
Buffer / Cache Dance

- A balance between buffer and cache allocations must be made for optimum performance

- Buffer space = large and cache space = small

  - Bottleneck in cache will occur between buffers and DASD

  - Cache will invalidate buffer entries to make room for new requests: thrashing

  - False Invalids occur when the cache structure needs more space than it has to satisfy CI requests.

- Buffer space = small and cache space = large

  - Potentially wasted coupling facility space is expensive

  - Cache will try to keep as much data in it as possible

  - Buffer LRU will be the constraining factor
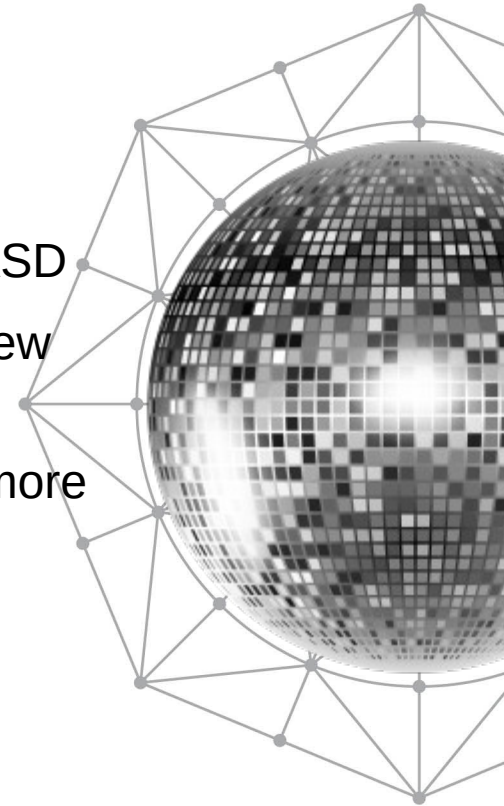
RLSSC - Sysplex Totals View

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


Samples: 120      Systems: 1     Date: 03/08/14  Time: 17.28.00  Range: 120    Sec


               < 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
RLS
  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  2303  99.4   0.0   0.6    100     0.00      1158
             SEQ    0.000  0.00   0.0   0.0   0.0    0.0     0.00      0.00
```
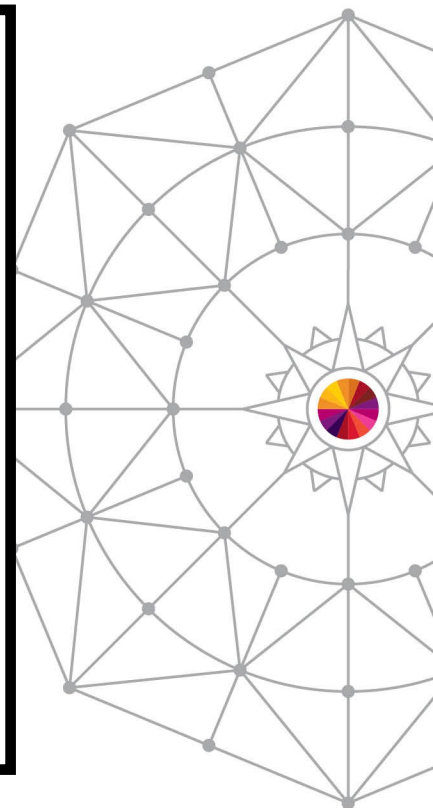
Caching Summary

- Caches should hold as much of your buffers as you can manage
- Use multiple caches to spread your workload
- Use RLS CF Cache Value wisely
- **Best Practice:** Monitor BMF False Contention as a health indicator for cache size
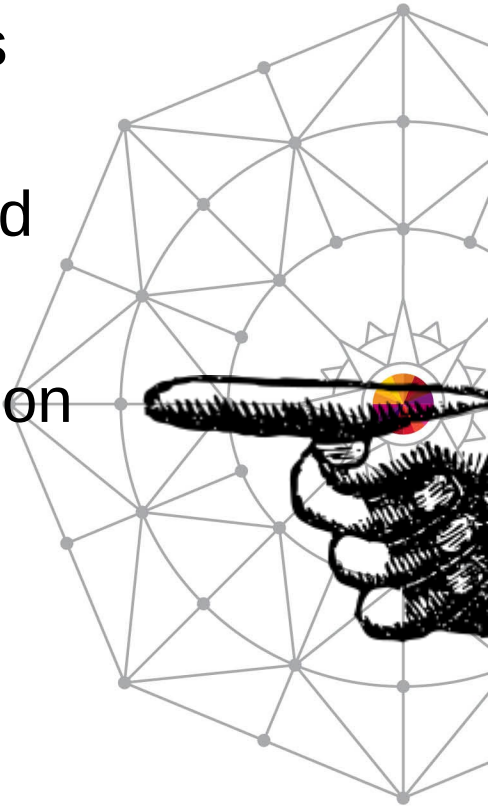
Agenda

- Overview of RLS
- Measurements Overview
- Locking and Structures
- Buffering
- Caching
- **Data set-level parameters**
- Request-level Parameters
- Diagnostics
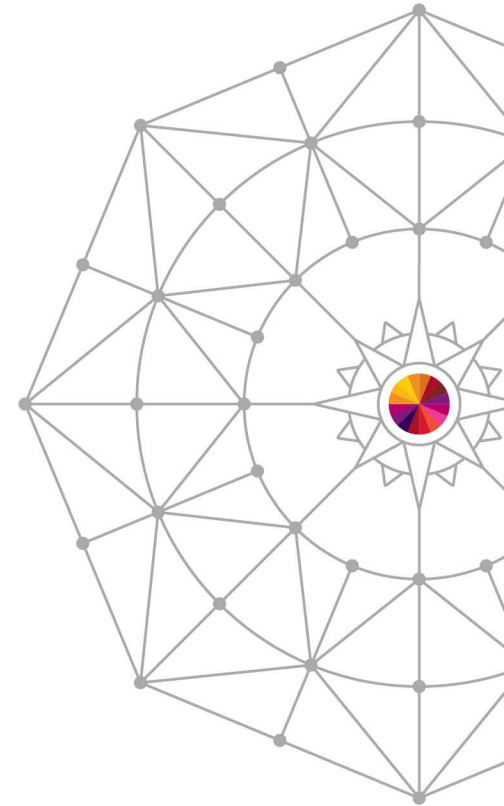
Data Set Parameters

- **CISIZE**
  - Currently, both RLS and base VSAM will serialize the *entire* data set when
    - CI / CA split is in progress
    - CI / CA reclaim is in progress
    - Accessing spanned records
  - **Best Practice**: When reasonable, increase CI size for data sets with higher insert activity.
    - Larger is better in terms of CPU
    - Larger also means larger buffers: performance vs. storage tradeoff
    - Larger CIs will contain more records which may lead to contention on DASD flush
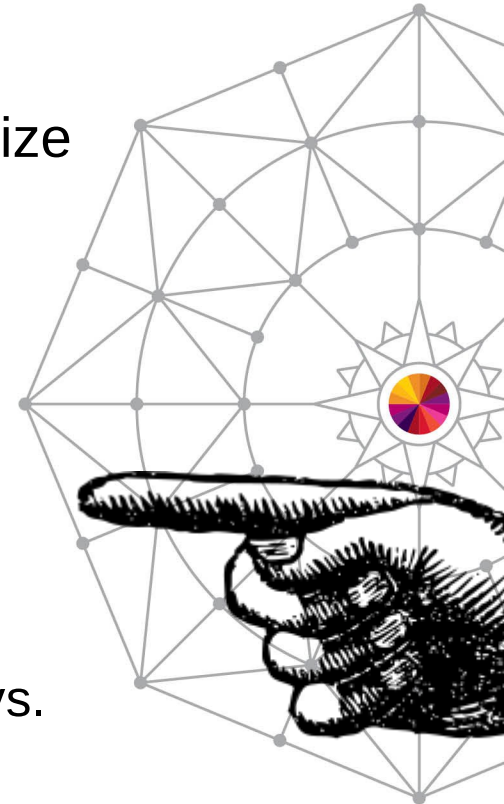
Data Class Parameters

- Extent Constraint Removal (YES | <u>NO</u>)

  - Specifies whether data sets can have > 255 extents

- CA Reclaim (<u>YES</u> | NO)

  - Specifies whether DASD space for empty CAs in KSDS data sets will be reused

  - Must be enabled in IGDSMSxx or with SETSMS command

  - Recommended to avoid data set fragmentation and to reduce reorg frequency
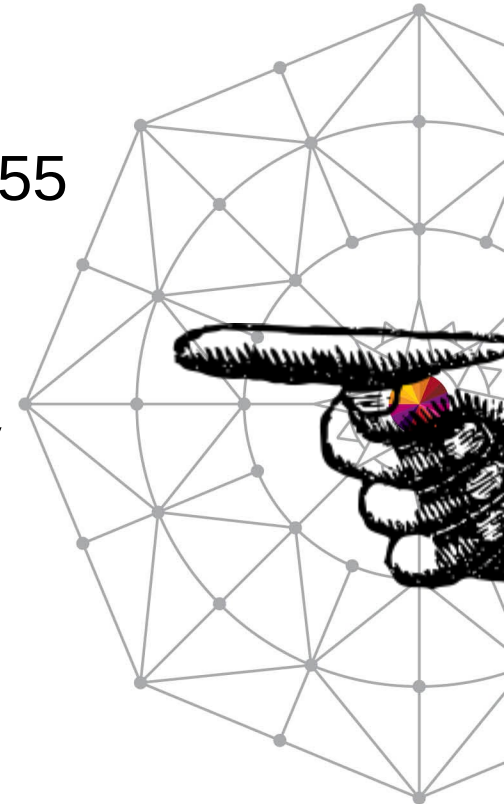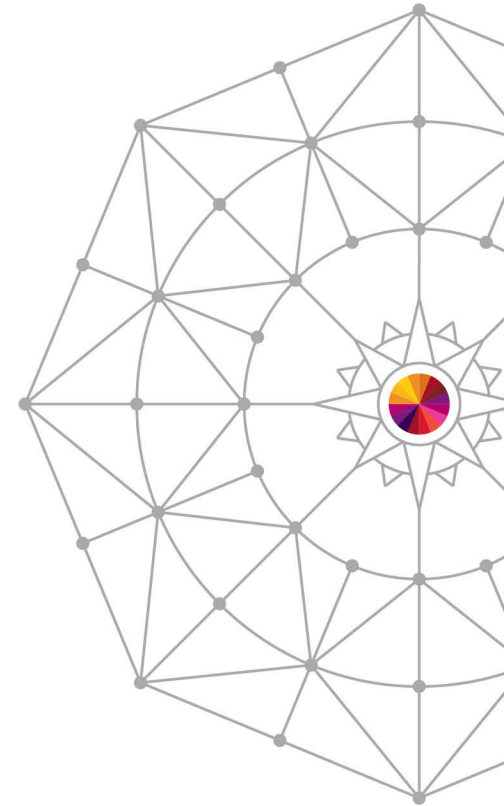
Data Class Parameters

- BWO (<u>NO</u> | TYPECICS | TYPEIMS)

  - BWO: Backup While Open

  - NO: BWO not used for CICS or IMS VSAM data sets

  - TYPECICS: BWO used for CICS VSAM file control data sets

    - Requires multiple SVC 26 calls to catalog
    - Component 1 lock held across VRM requests

  - TYPEIMS: BWO used for IMS VSAM data sets with IMS 6.1+

  - **Recommendation:** To maximize concurrent VRM performance, only use BWO when absolutely necessary.

Agenda

- Overview of RLS
- Measurements Overview
- Locking and Structures
- Buffering
- Caching
- Data set-level parameters
- **Request-level Parameters**
- Diagnostics

Request Level – ACB / GENCB / RPL

- RLSREAD / OPTCD (<u>NRI</u> | CR | CRE)

  - NRI – No Read Integrity
    - No Read Integrity
    - Can occasionally result in "dirty" read

  - CR – Consistent Read
    - Will get / release record lock

  - CRE – Consistent Read Extended
    - Will get record lock
    - Locks released at commit
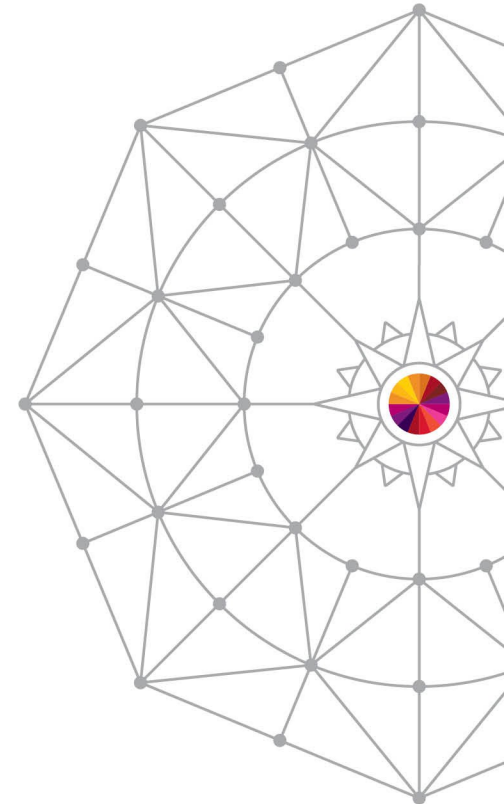
Request Level Considerations

- Be careful when using CRE
- Long-running transactions can hold locks, causing delays

- **Best Practice**: use GET NUP when browsing, and then use GET UPD before update. If consistency is required, use CR

Agenda

- Overview of RLS
- Measurements Overview
- Locking and Structures
- Buffering
- Caching
- Data set-level parameters
- Request-level Parameters
- **Diagnostics**
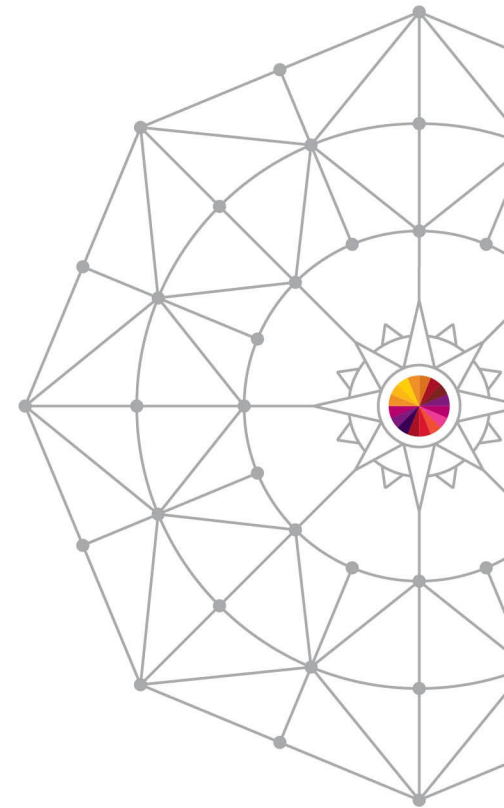
SMSVSAM Diagnostics

- Display commands to run

  - D SMS,SMSVSAM,ALL
  - D SMS,TRANVSAM,ALL   (if TVS is in use)
  - D SMS,SMSVSAM,DIAG(C)   (system scope)
  - D SMS,SMSVSAM,QUIESCE (system scope)
  - D GRS,C   (system scope)
  - D XCF,STR,STRNM=[IGWLOCK00 | lock_structure]
  - D XCF,STR,STRNM=[cachestructurename]
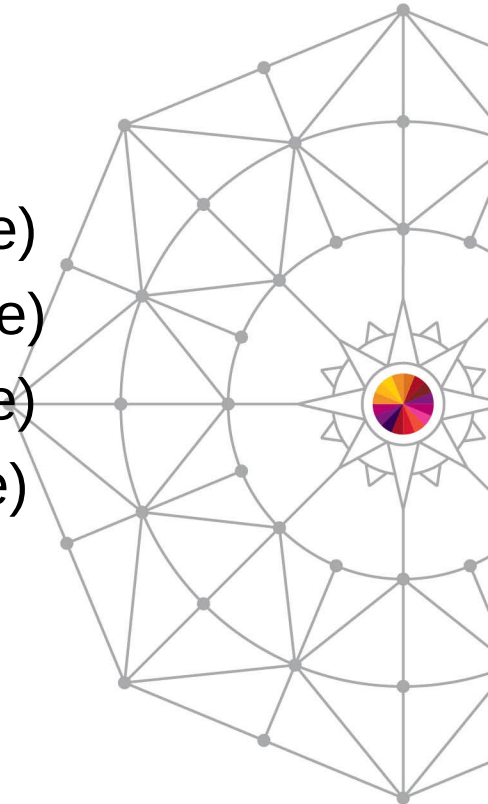
Gather Diagnostic Dumps

- Plex-wide dumps

    - Need SMSVSAM address and data spaces

    - Dump plex-wide if you can

- **Best Practice**: Create an IEADMCxx member for dumping SMSVSAM and perhaps CATALOG

    - Example of a plex-wide parmlib definition:

    ```
    COMM=(SMSVSAM hang)
    JOBNAME=(*MASTER*,SMSVSAM),
    DSPNAME=('SMSVSAM'.*),
    SDATA=(ALLNUC,COUPLE,CSA,GRSQ,LPA,LSQA,
    PSA,RGN,SUM,SQA,SWA,TRT,XESDATA),
    REMOTE=(SYSLIST=(*('SMSVSAM')),DSPNAME,SDATA)
    ```
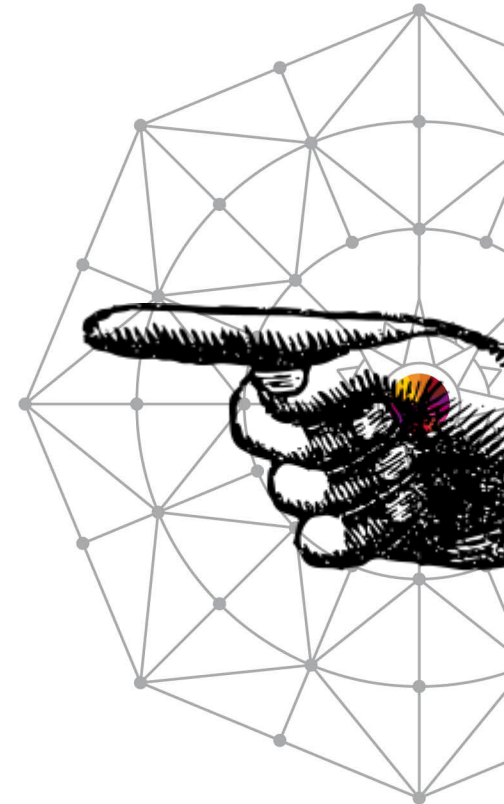
For more on diagnosis, dig up session 14153 from Aug 2013 Share Boston:

"Advanced RLS Diagnostics and Recovery"

Link: http://goo.gl/vBHuX9
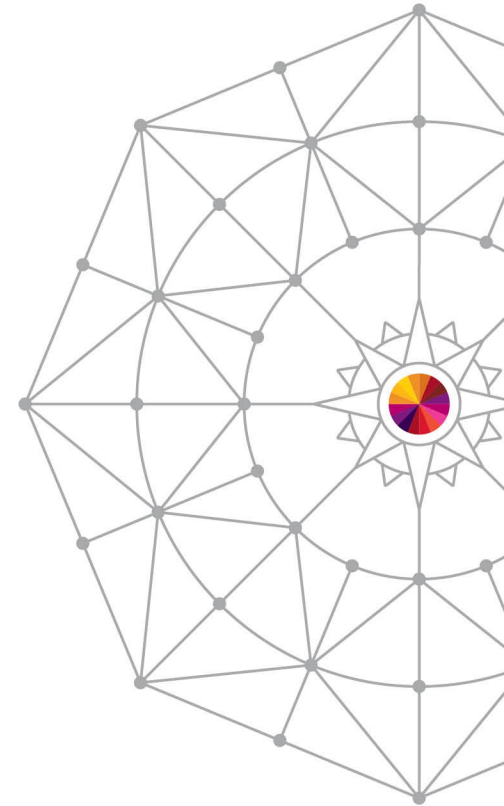
Maintenance – PSP Bucket

- The RLS PSP bucket is on the web at:
  - http://www14.software.ibm.com/webapp/set2/psearch/search?domain=psp
- 'Search for: SMSVSAM'
- Maintenance organized by release
  - Upgrade ZOSV2R1, Subset DFSMS
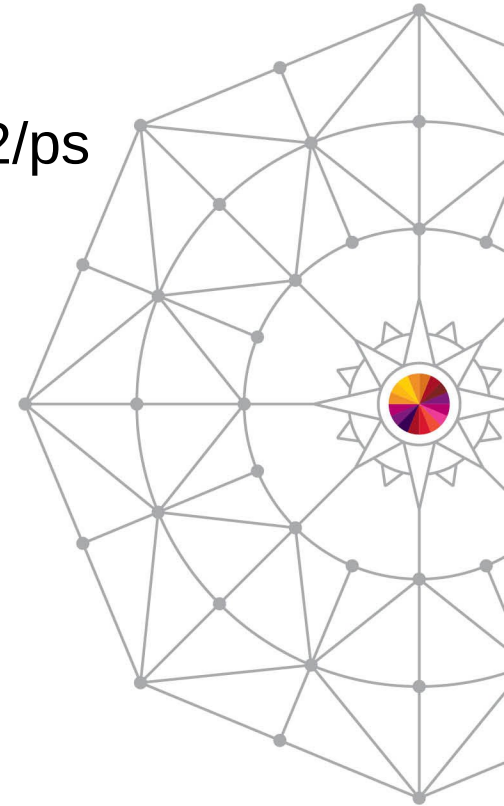
Agenda

- Overview of RLS
- Measurements Overview
- Locking and Structures
- Buffering
- Caching
- Data set-level parameters
- Request-level Parameters
- Diagnostics

References

- VSAM Demystified
  - SG24-6105-01
- DFSMSdfp Storage Administration
  - SC26-7402
- DFSMS: Using Data Sets
  - SC26-7410
- z/OS MVS System Management Facilities
  - SA22-7630

Trademarks

DFSMSdfp, DFSMSdss, DFSMShsm, DFSMSrmm, IBM, IMS, MVS, MVS/DFP, MVS/ESA, MVS/SP, MVS/XA,

OS/390, SANergy, and SP are trademarks of International Business Machines Corporation in the United States, other countries, or both.

AIX, CICS, DB2, DFSMS/MVS, Parallel Sysplex, OS/390, S/390, Seascape, and z/OS are registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Domino, Lotus, Lotus Notes, Notes, and SmartSuite are trademarks or registered trademarks of Lotus

Development Corporation. Tivoli, TME, Tivoli Enterprise are trademarks of Tivoli Systems Inc. in the United

States and/or other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

Other company, product, and service names may be trademarks or service marks of others.

Notices & Disclaimers

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 any other 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.
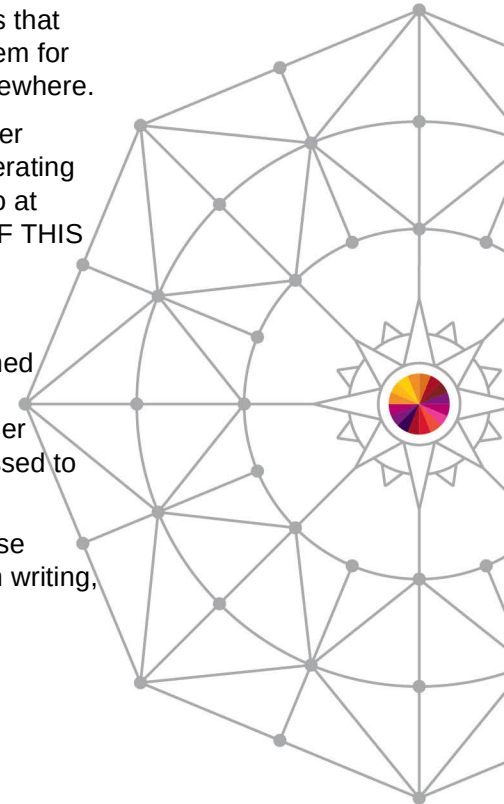
S H A R E
Technology · Connections · Results

DFSMS Basics: VSAM

VSAM RLS Best Practices

How to leverage VSAM RLS best practices and performance

Neal Bohling, bohling@us.ibm.com
Session : 15090

SHARE
in Anaheim

Configuring RMF Monitor III

- SYS1.PARMLIB(ERBRMF04) keywords
  - CFDETAIL – detailed CF information
  - VSAMRLS – enables RLS data gathering
  - VSAMRLS(ADD(DSMASK))
    - enables data-set level statistics

- To start
  - S RMF
  - F RMF,S III
  - Issue TSO RMF