# VSAM Record Level Sharing

Page 1 of 2

- VSAM Record Level Sharing (RLS) is a major extension to VSAM.
  NOTE: Although it was designed for use by CICS, it can be used by ***any*** application.
- VSAM RLS extends the DFSMS storage hierarchy to support a data-sharing environment across multiple systems in a Parallel Sysplex..
  - This support is primarily for VSAM data sets that online-transaction-processing applications use.
  - It provides multi-system VSAM data sharing at a record level for CICS applications.
  - The VSAM RLS product is designed to exploit z/OS' Coupling Technology.
- VSAM RLS provides read-only access by non- CICS applications concurrent with read/write by CICS applications to recoverable data sets.
  - It allows any number of users within your Parallel Sysplex to share your existing VSAM data sets.
  - It can provide full data integrity (read and write).
  NOTE: The serialization is at record level. However, to implement recoverable VSAM data sets the users must have their own backout log, as CICS has.
- VSAM RLS does not introduce new types of VSAM data sets; rather, it introduces a new way of accessing existing data sets.
  - Apart from the need to open data sets in RLS mode, the same VSAM record management interfaces are used.
- You can specify the RLS mode in the MACRF parameter of the ACB macro that you use to open the data set in your program or you can also specify the RLS mode by using the *new keyword RLS in your DD card* that points to your VSAM data set in your JCL.
  NOTE: RLS mode can be used with KSDS, RRDS, VRRDS, and ESDS VSAM data sets. PATH (AIX) access for KSDS and ESDS is allowed with RLS.
  - Extended format, extended addressability, spanned, and compression are also supported by RLS.
  - Beginning with z/OS 1.12, VSAM stripping is also supported for VSAM data sets being accessed in RLS mode.

⚠ RLS and non-RLS VSAM data sets can coexist.

NOTE: Access method services do not use RLS when performing an IDCAMS EXPORT, IMPORT, PRINT, or REPRO command. If the RLS keyword is specified in the DD statement of a data set to be opened by access method services, the keyword is ignored and the data set is opened and accessed in non-RLS mode.

- Both CICS and non-CICS jobs can have concurrent read or write access to non-recoverable data sets.
  NOTE: there is no coordination between CICS and non-CICS, so data integrity can be compromised.
  - Non-CICS jobs can have read-only access to recoverable data sets concurrently with CICS jobs, which can have read or write access.
- VSAM RLS uses a coupling facility to perform data set-level locking, record locking, and data caching.
  - VSAM RLS uses the conditional write and cross-invalidate functions of the coupling facility cache structure, thereby avoiding the need for control interval (CI) level locking. VSAM RLS uses the coupling facility caches as store-through caches.
    NOTE: When a control interval of data is written, it is written to both the coupling facility cache and to direct access storage device (DASD). This ensures that problems occurring with a coupling facility cache do not result in the loss of VSAM data.
- VSAM RLS uses a Coupling Facility to perform data-set-level locking, record locking, and data caching.
  - VSAM RLS uses the conditional write and cross-invalidate functions of the Coupling Facility cache structure, thereby avoiding the need for control interval (CI) level locking.
- VSAM RLS uses the Coupling Facility caches as store-through caches.
  - When a control interval of data is written, it is written to both the Coupling Facility cache and the direct access storage device (DASD).
  - This ensures that problems occurring with a Coupling Facility cache do not result in the loss of VSAM data.
- RLS does not support the following options and capabilities:
  - Linear, KEYRANGE, IMBED, and temporary data sets
  - Addressed access to KSDS data sets
  - Control Interval Access
  - User Buffering (UBF)
  - GETIX and PUTIX requests
  - MVS Checkpoint/Restart facility
  - ACBSDS (system data set) specification
  - Hiperbatch
  - Catalogs, VVDS, the JRNAD exit, and any JCL AMP= parameters in JCL
  - Data that is stored in z/OS UNIX System Services
  - Requests that are issued while the caller is running in these modes**:**
    Cross-memory mode, SRB mode, or under an FRR.

CICS **function shipping** allows for writing CICS Application programs without regard to the location of the requested resources although this technique *limits* program independence

VSAM RLS is a data set access mode that enables multiple address spaces, CICS application-owning regions on multiple systems, and batch jobs to access recoverable VSAM data sets at the same time.

VSAM RLS processing involves support from multiple products:
- ➢ CICS Transaction Server
- ➢ CICS VSAM Recovery (CICSVR)
- ➢ DFSMS

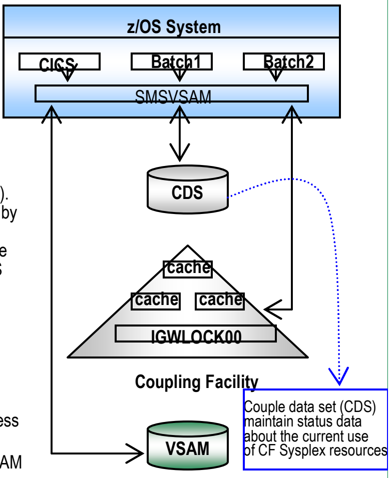
If you need to share a data set among address spaces, multiple systems, or both, consider using record-level sharing (RLS) instead of GSR.

RLS is not supported for control interval mode access or for HFS files.

VSAM RLS requires that the data sets be System Managed Storage (SMS) data sets.

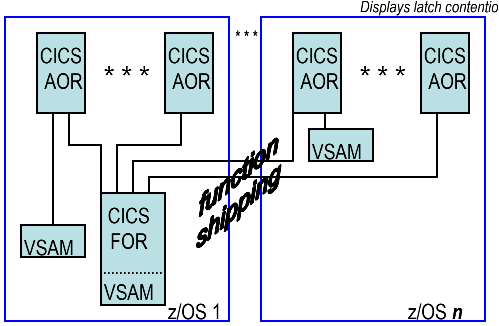
Example - RLS diagnostic command **See Page2**
D SMS,SMSVSAM,DIAG(contention)
*Displays latch contention on SMSVSAM resources*

- You can access VSAM data sets in RLS mode from users that run in just one z/OS image (monoplex).
  NOTE: Even in a monoplex environment a Coupling facility is required. The reason for doing that is to use the better serialization that is provided by RLS. For example, in a non-RLS mode two jobs cannot simultaneously access for update a VSAM data set with total read and write integrity. That is, for SHAREOPTIONS (1,3), just one can open the VSAM data set. The other must wait for the first to close the data set. The granularity of the serialization is at data set level.
- If the two jobs access the data set in RLS mode, both the OPENs succeed.
  - Both in parallel can then update the VSAM data set.
  - The serialization is at logical record level.
- VSAM SHAREOPTIONS are ignored when you access your data set in RLS mode.
  - You can have any number of read/write users who can access the VSAM data set in RLS mode.
  NOTE: There is one exception to this rule about SHAREOPTIONS(2 x). VSAM data sets defined with SHAREOPTIONS(2 x) can be accessed by any number of read/write users in RLS mode. They can also be accessed by any number of read users in non-RLS mode. However, the non-RLS users do not have any read integrity. A read user in non-RLS mode can be a batch job for printing the data set. Or it can use the IDCAMS repro to create a copy of the data set for test purposes.
  - An RLS client is any address space that starts an RLS function that results in a call to the SMSVSAM address space.
  Examples of RLS functions are OPEN, CLOSE, GET, PUT, and DELETE.
  Examples of application that can be registered as RLS client address spaces are CICS, batch jobs, and DFSMShsm.
  - A record lock is an XES lock resource that is obtained by SMSVSAM on behalf of a user and associated with a logical record.
  - The lock resource name is based on a 16-byte hashed version of the record's key (or RBA, or RRN), and the data set name and component name.
  - There are also other locks to serialize CI/CA splits.
  NOTE: SMSVSAM maintains two different types of record-level locks in its coupling-facility lock structures: Exclusive locks, and shared locks. Exclusive locks are used for any update request, whereas shared locks are used to support read integrity. The lock includes the name of the corresponding lock owner, which is the application ID (CICS, DFSMStvs or any other), and the unit of work ID.



Couple data set (CDS) maintain status data about the current use of CF Sysplex resources

**RLS in a Monoplex**



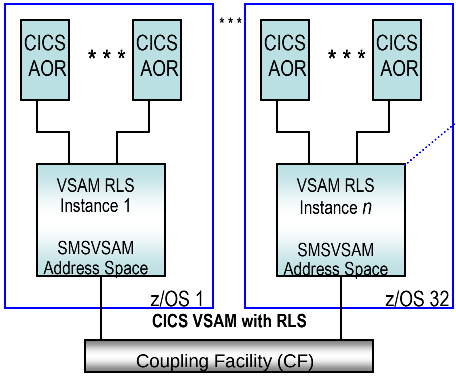
**CICS VSAM *non*-RLS**

Prior to VSAM RLS, a CICS VSAM data set was owned and directly accessed by one single CICS. Shared access across CICS Application Owning Regions (AORs) to a single VSAM data set was provided by CICS *function shipping*. With function shipping, one CICS File Owning Region (FOR) accesses the VSAM data sets on behalf of other CICS regions

**Problems**

There are a couple of problems with this CICS configuration:
- ➢ CICS FOR is a single point of failure.
- ➢ Multiple system performance is not acceptable.
- ➢ Lack of scalability.
- ➢ Over time the FORs became a bottleneck because CICS environments became increasingly complex. CICS required a solution to have direct shared access to VSAM data sets from multiple CICSs.
- ➢ Need to commit remotely.
- ➢ VSAM's HURBA/HARBA are updated only when CICS closes the data set, causing conflicts along shared updates.

NOTE: These issues are fully addressed by VSAM RLS.



**CICS VSAM with RLS**

**Benefits of VSAM RLS**

The benefits of VSAM RLS are:
- ➢ Enhances cross-system data sharing - scope is sysplex
- ➢ Improves performance and availability in CICS and also non-CICS VSAM environments
- ➢ Provides data protection after a system failure
- ➢ Provides automation for data recovery
- ➢ Provides full read/write integrity to your existing VSAM files; the user does not need to serialize using ENQ/DEQ macros
- ➢ Allows CICS to register as a recoverable subsystem, which will automate recovery processing as well as protect the data records to be recovered.

The *SMSVSAM address space* automatically starts at IPL when the RLSINIT (YES) keyword is specified in the IGDSMSxx member of SYS1.PARMLIB.

VSAM RLS does not support record locks for a single VSAM sphere (index / data components) to be placed in multiple lock structures.

VSAM internally performs cross-address space accesses and linkages between requestor address spaces and the SMSVSAM server address space. The SMSVSAM server owns two data spaces. One data space is called the SMSVSAM data space. It contains some VSAM RLS control blocks and a system-wide buffer pool. The other data space is used by VSAM RLS to collect activity monitoring information used to produce SMF records.

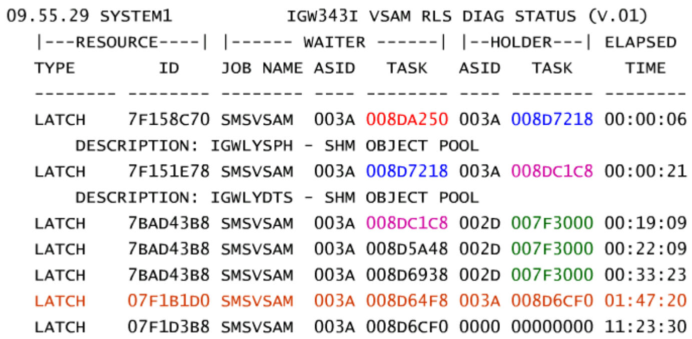
- VSAM RLS data buffers occupy the largest share of the SMSVSAM data-space storage.
  - In some cases, storage limits on the data buffers may create performance slowdowns in high-volume transaction environments.
    NOTE: To avoid any storage limits and potentially enhance performance VSAM RLS offers the option to move RLS data buffers into 64-bit addressable virtual storage. This option can be activated by assigning VSAM data sets to a data class with ISMF that specifies RlsAboveTheBar(YES). IBM recommends that you use this option, especially for applications with a high rate of critical CICS transactions.
- VSAM record-level sharing allows multiple levels of CF caching for DFSMS cache structures that are defined in the active storage management subsystem (SMS) configuration.
- VSAM RLS has multiple levels of CF caching.
  - The value of the SMS DATACLAS RLS CF Cache Value keyword determines the level of CF caching.
  - The default value, ALL, indicates that RLS caches both the data and index parts of the VSAM data set in the coupling facility.
    NOTE: If you specify NONE, then RLS caches only the index part of the VSAM data set. If you specify UPDATESONLY, then RLS caches data in the coupling facility only during write operations.
  - All active systems in a sysplex must have the greater than 4K CF caching feature before the function is enabled.
- When an application opens a VSAM data set, RLS processing determines which lock structure to use by checking the storage class defined for the data set.
  - If the storage class specifies a secondary lock structure, RLS processing uses the secondary lock structure for serializing access to records in the data set. Otherwise, RLS processing uses IGWLOCK00 for all record locking (see illustration above).

# Two RLS Diagnostics Commands

## D SMS,SMSVSAM,DIAG(contention)

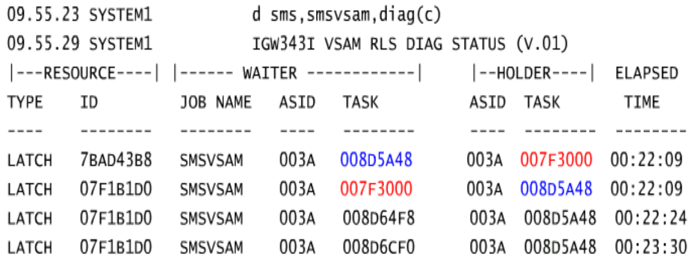
This helps determine which systems are actually experiencing a hang, as opposed to which are victims.

- ➢ Will display any latch contention on SMSVSAM resources
- ➢ Latches are 8 bytes of storage used as a logical means to serialize resources. They are either held, or waited on; there are no shared requests.
- ➢ The contention displayed will show the latch address, the holder and any waiters if the latch is in contention.
- ➢ The display will show elasped time for how long a latch was in contention.



```
09.55.29 SYSTEM1           IGW343I VSAM RLS DIAG STATUS (V.01)
   |---RESOURCE----| |------ WAITER ------| |--HOLDER---| ELAPSED
   TYPE         ID     JOB NAME ASID    TASK    ASID    TASK      TIME
   -------- -------- -------- ---- -------- ---- -------- --------
   LATCH    7F158C70 SMSVSAM  003A 008DA250 003A 008D7218 00:00:06
      DESCRIPTION: IGWLYSPH - SHM OBJECT POOL
   LATCH    7F151E78 SMSVSAM  003A 008D7218 003A 008DC1C8 00:00:21
      DESCRIPTION: IGWLYDTS - SHM OBJECT POOL
   LATCH    7BAD43B8 SMSVSAM  003A 008DC1C8 002D 007F3000 00:19:09
   LATCH    7BAD43B8 SMSVSAM  003A 008D5A48 002D 007F3000 00:22:09
   LATCH    7BAD43B8 SMSVSAM  003A 008D6938 002D 007F3000 00:33:23
   LATCH    07F1B1D0 SMSVSAM  003A 008D64F8 003A 008D6CF0 01:47:20
   LATCH    07F1D3B8 SMSVSAM  003A 008D6CF0 0000 00000000 11:23:30
```

*Sample display fro DIAG command #1*



```
 09.55.23 SYSTEM1           d sms,smsvsam,diag(c)
 09.55.29 SYSTEM1           IGW343I VSAM RLS DIAG STATUS (V.01)
 |---RESOURCE----| |------ WAITER ------------|     |--HOLDER----|  ELAPSED
 TYPE    ID        JOB NAME   ASID   TASK           ASID  TASK        TIME
 ----    --------  --------   ----   --------       ----  --------   --------
 LATCH   7BAD43B8  SMSVSAM    003A   008D5A48       003A  007F3000   00:22:09
 LATCH   07F1B1D0  SMSVSAM    003A   007F3000       003A  008D5A48   00:22:09
 LATCH   07F1B1D0  SMSVSAM    003A   008D64F8       003A  008D5A48   00:22:24
 LATCH   07F1B1D0  SMSVSAM    003A   008D6CF0       003A  008D5A48   00:23:30
```

*Sample display fro DIAG command #2*

**| - - RESOURCE - - |**
TYPE: Resource in contention (Almost always a Latch)
ID: Address location of the 8 byte latch

**| - - WAITER - -|**
JOB NAME: Job name of Waiter
ASID: Address space ID of Waiter
TASK: TCB of Waiter

**| - - HOLDER - - |**
ASID: Address Space ID of Latch Holder
TASK: TCB of Latch Holder

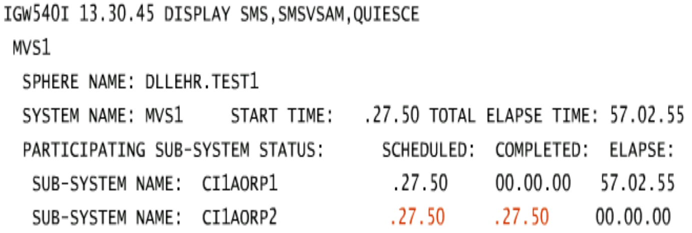
**| - ELAPSED TIME - - |**
The amount of time that the latch was in contention.

NOTES:
- • DIAG should be used in conjunction with D GRS,C to determine if any of the latch holders are waiting on ENQs.
- • Any latch contention with an elasped time of more than a few seconds is most likely stuck.
- • Sometimes the holding ASID id not SMSVSAM, but another address space like a CICS region. Canceling that region could avoid an SMSVSAM wide outage.

## D SMS,SMSVSAM,QUIESCE

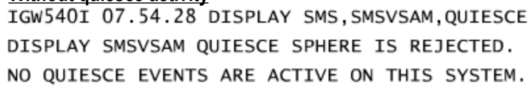
- ➢ The quiesce display wil show any outstanding quiesce events
- ➢ A quiesce event is a decision to allow unhindered CICS access to an RLS dataset.
- ➢ Any CICS regions that are part of the quiesce event will show up in the display.
- ➢ If there is not an event in progress you will receive an IGW540I message rejecting the command.

**With quiesce activity**



```
IGW540I 13.30.45 DISPLAY SMS,SMSVSAM,QUIESCE
 MVS1
  SPHERE NAME: DLLEHR.TEST1
  SYSTEM NAME: MVS1     START TIME:   .27.50 TOTAL ELAPSE TIME: 57.02.55
  PARTICIPATING SUB-SYSTEM STATUS:       SCHEDULED:  COMPLETED:  ELAPSE:
   SUB-SYSTEM NAME:  CI1AORP1              .27.50     00.00.00   57.02.55
   SUB-SYSTEM NAME:  CI1AORP2              .27.50     .27.50     00.00.00
```

**Without quiesce activity**



```
IGW540I 07.54.28 DISPLAY SMS,SMSVSAM,QUIESCE
DISPLAY SMSVSAM QUIESCE SPHERE IS REJECTED.
NO QUIESCE EVENTS ARE ACTIVE ON THIS SYSTEM.
```

- • **SPHERE NAME:** Name of dataset being quiesced/unquiesced.
- • **SYSTEM NAME:** System where command was issued.
- • **SUB-SYSTEM NAME:** Name of CICS region involved in quiesce event.
  - - SCHEDULED: Time when the quiesce event was issued.
  - - COMPLETED: Time when CICS region responded to event.
  - - ELAPSE: Time between when the quiesce event was issued and it was finally responded to by CICS.

NOTES:
- • The Quiesce command is invaluable to determine what CICS regions are holding up a quiesce request.
- • Any CICS regions that have 00.00.00 in the completed section are most likely in trouble and should be dumped along with SMSVSAM before termination.
- • In most scenarios, the CICS region, instead of SMSVSAM, can be terminated to allow the quiesce event to finished. Thus saving a SMSVSAM wide outage.