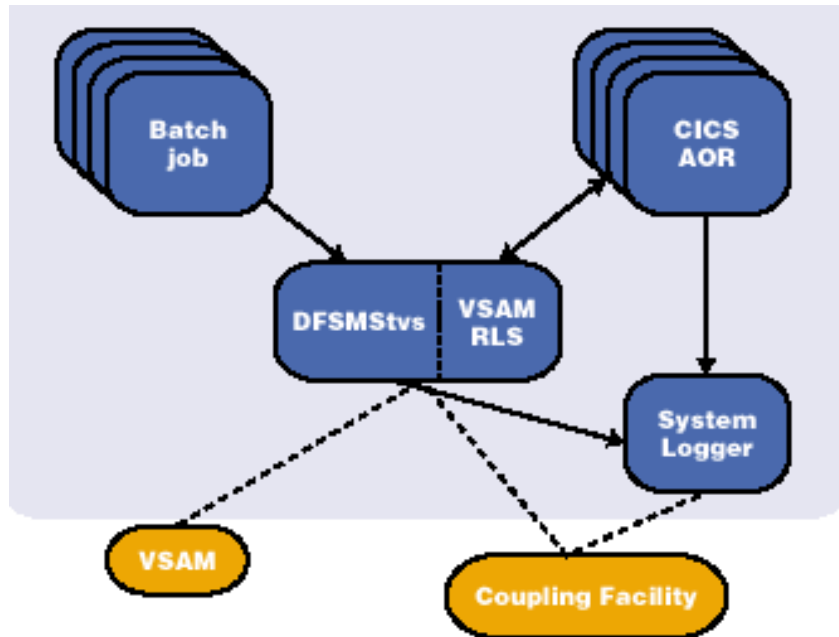


The Case for and Value of Transactional VSAM

A CICS/Batch File Sharing Enhancement



IBM zSeries Technical Conference
Session TSS03

Ruth Ferziger

IBM Design & Development

ruthf@us.ibm.com

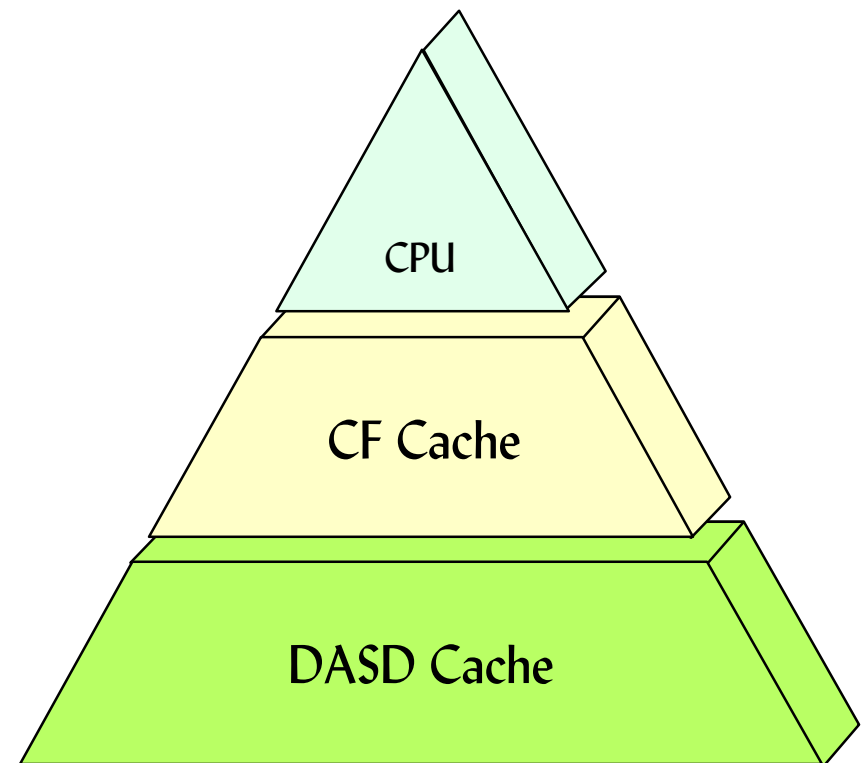
Agenda

- What is VSAM Record Level Sharing?
- The business value of Transactional VSAM
- What is Transactional VSAM?
- How Transactional VSAM fits into the system
- High level batch job considerations

What is VSAM RLS?

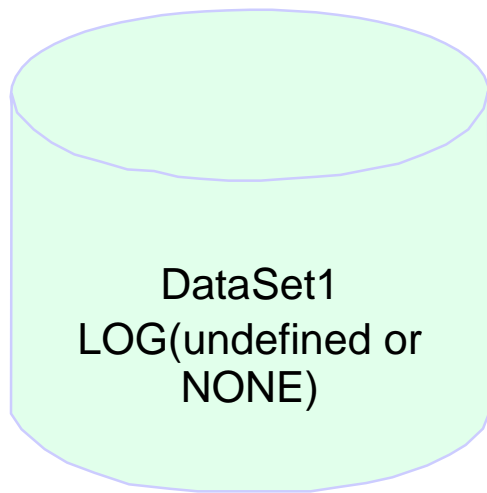
- Alternative to 'traditional' VSAM
- Enhances cross system data sharing - scope is sysplex
- Uses the same VSAM interfaces & data format
- KSDS, ESDS, RRDS & VRRDS
- **Must be SMS-managed!**
- One server per MVS image
- Access mode specified on ACB or on JCL
- Record level locking via CF

Storage Hierarchy



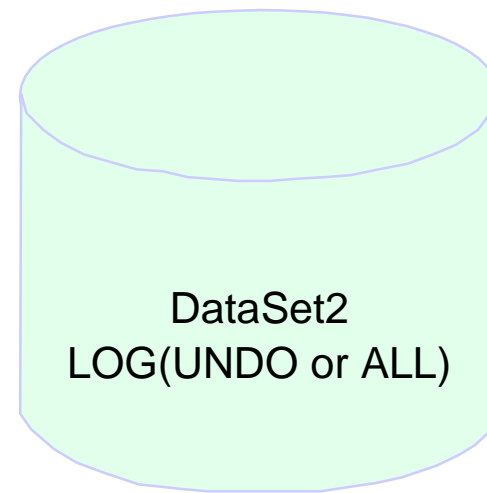
What is a recoverable data set?

Non-recoverable data set



Accessible for read &
write in RLS mode

Recoverable data set

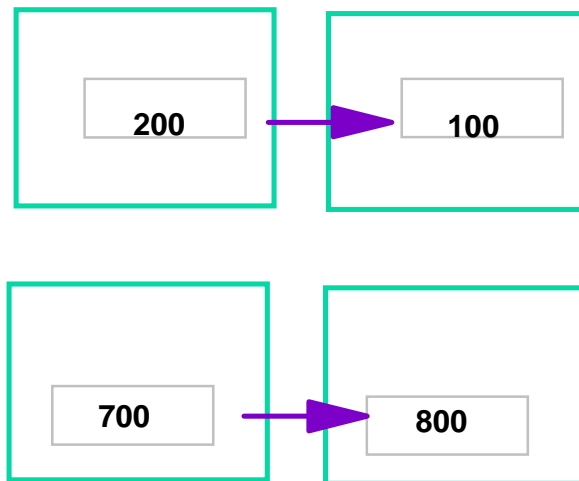


Accessible for **read-only**
in RLS mode

What is Transactional Recovery?

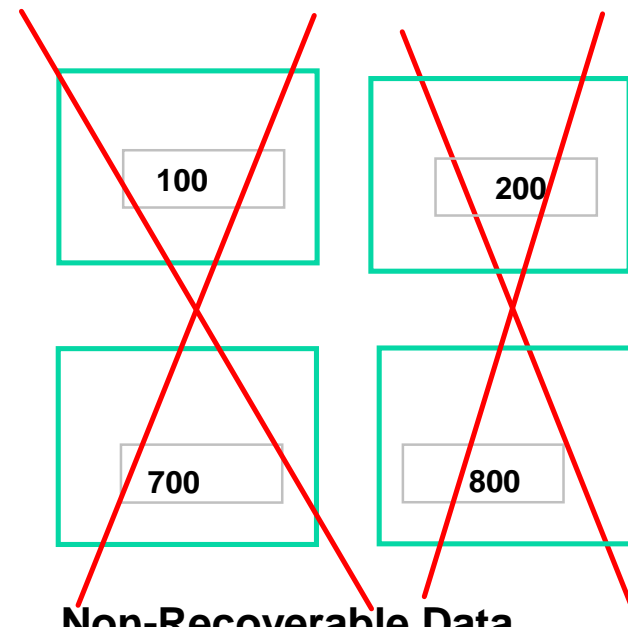
Atomic commit of **multiple** changes to recoverable resources

- ▶ All or nothing
- ▶ Exclusive locks with back out on failure



Recoverable Data

Atomic commit of both changes



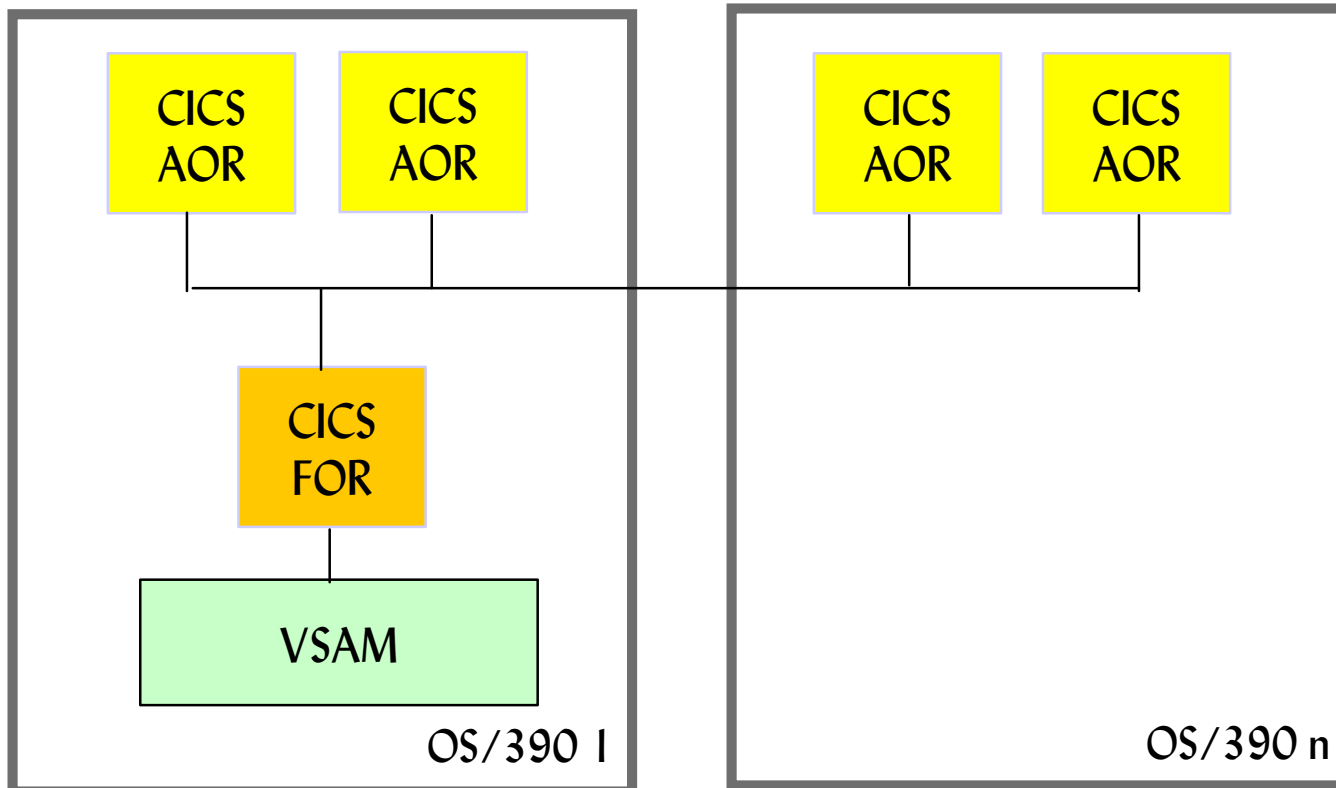
Non-Recoverable Data

failure may result in partial update (one change made, the other change NOT made)

CICS Function Shipping Before RLS

AOR = Application Owning Region

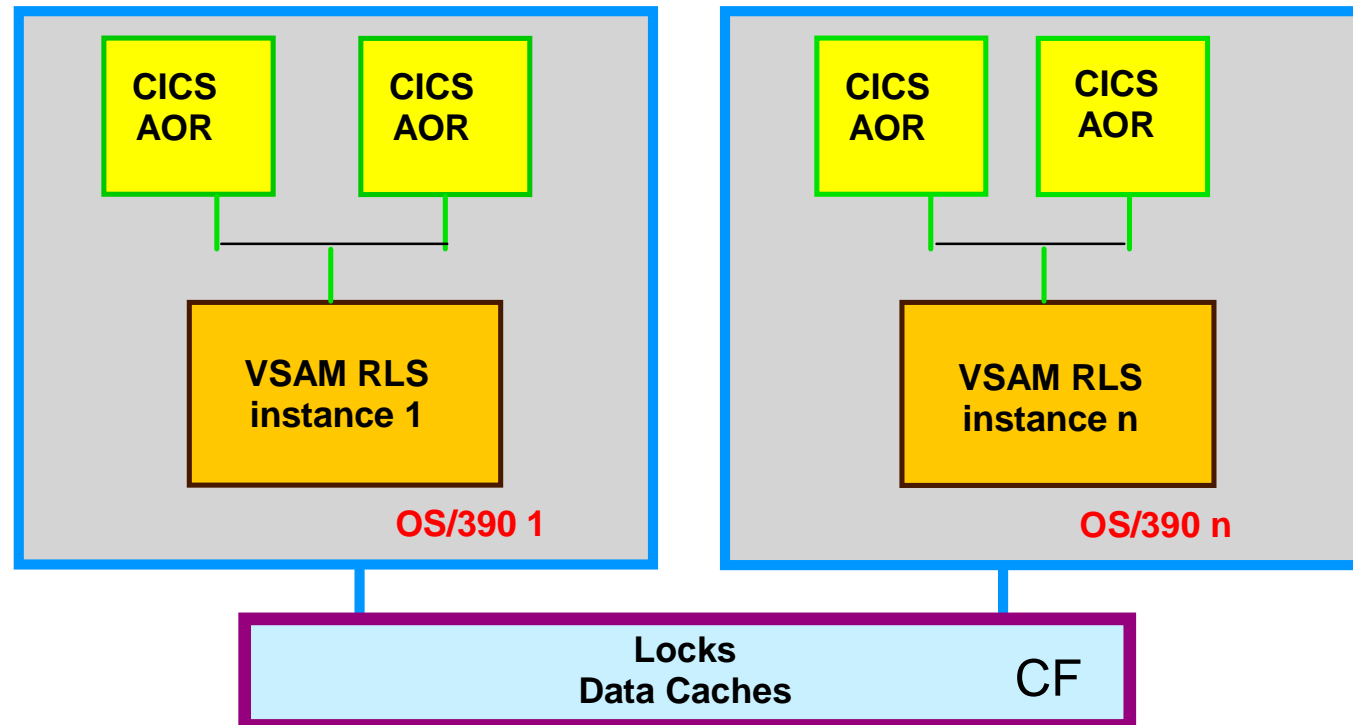
FOR = File Owning Region



Problems:

- ◆ CICS FOR is a single point of failure
- ◆ Multiple system performance is no acceptable (uses VTAM or XCF cross system)
- ◆ No exploitation of System/390 Parallel Sysplex

Parallel Sysplex CICS with VSAM RLS



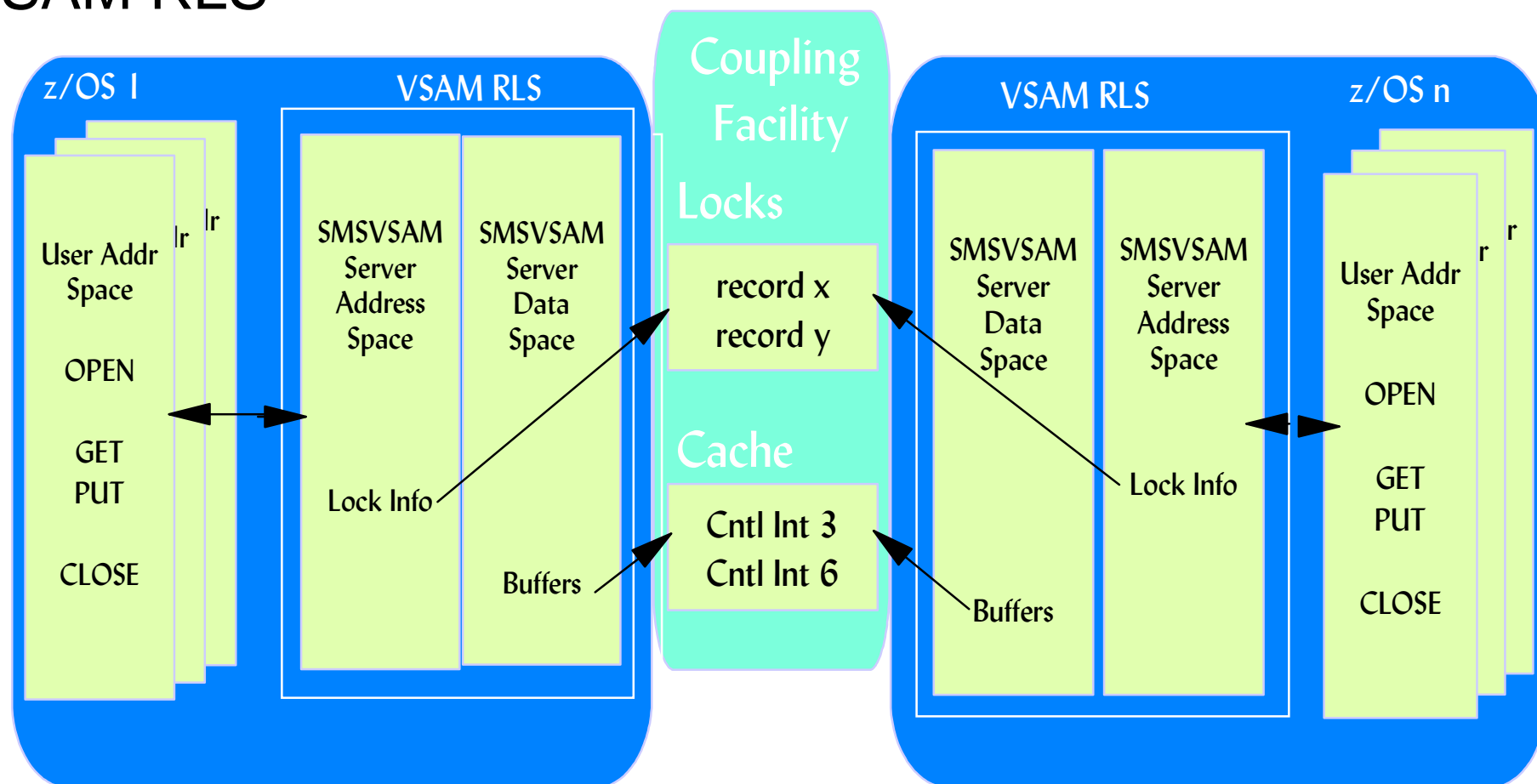
Customer value:

- ◆ Price/performance of CMOS z/OS
- ◆ Availability, applications remain available across CICS, z/OS, S/390 failures
- ◆ Capacity via granular growth (add S/390s to sysplex)

Using VSAM RLS

- A new data set access mode specified in the ACB or on the JCL
 - ▶ MACRF = NSR | LSR | GSR | RLS
 - ▶ JCL RLS=NRI or CR keyword
- Record level sharing across multiple z/OSs
 - ▶ Scope is a z/OS sysplex
 - ▶ Serialization is at the record level rather than at the CI level
- VSAM RLS Multisystem Server
 - ▶ One SMSVSAM address space in each z/OS
 - ▶ Performs record level locking via System/390 Coupling Technology: the *coupling facility (CF)*

VSAM RLS



Coupling Facility provides mechanism for:

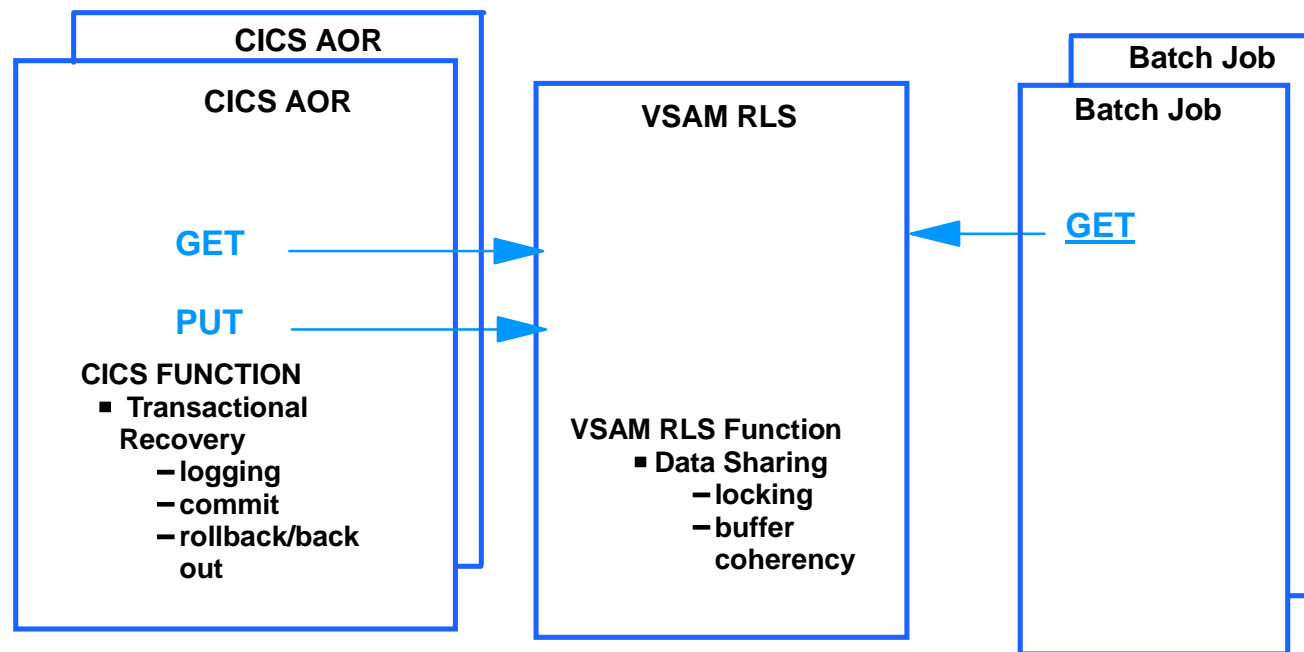
- ♦ record locking
- ♦ buffer coherency: cross invalidate/store thru cache

- ♦ VSAM RLS is an access mode specified in the ACB
- ♦ Requests are processed in the VSAM RLS server address space

RLS Read Integrity

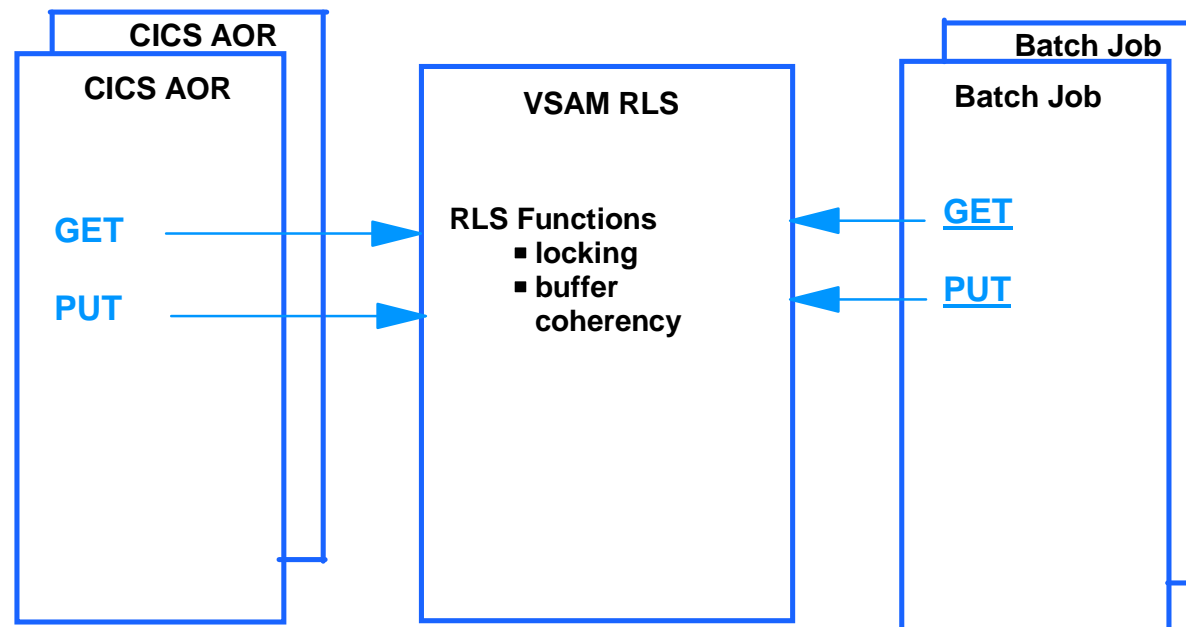
- **NRI - no read integrity**
 - ▶ Also known as "dirty read"
 - ▶ Does no locking
- **CR - consistent read**
 - ▶ Prevents reading of uncommitted data
 - ▶ Obtains a shared lock for the duration of the read
- **CRE - consistent read explicit**
 - ▶ Also known a "repeatable read"
 - ▶ Obtains a shared lock for the duration of the transaction
 - ▶ Locks out any updaters for the duration of the transaction
 - ▶ In a pure RLS world, only available to CICS

RLS Sharing of *Recoverable* Files



- READ/WRITE sharing across CICS AORs
 - CICS provides transactional recovery
- and *READ ONLY* sharing with/across Batch Jobs

RLS Sharing of *Non-Recoverable* Files



- NO TRANSACTIONAL RECOVERY (no logging, commit, back out)
- *Read/Write* sharing across CICS AORs and *Batch Jobs*
- Most transactions and Batch Jobs are NOT designed to use this form of data sharing

What is the Batch Window Problem?

- A period of time in which CICS access to **recoverable** data sets is quiesced so batch jobs can run
- Requires taking a backup of the data set
- Batch updates are then performed
- A forward recovery backup is taken, if needed
- When finished, CICS access to the data set is re-enabled

DFSMSStvs enhances the availability of CICS/VSAM applications by helping eliminate this problem

Transactional VSAM Background

- IBM Statement of Direction on Transactional VSAM Services (TVS) in May 2000
- TVS is a priced feature of z/OS 1.4
- TVS builds on the functionality provided by VSAM Record Level Sharing (VSAM RLS)
- TVS is a member of the DFSMS family of products
 - ▶ Formally known as "**DFSMS**tv**s**"
 - ▶ It is like other DFSMS family members such as DFSMSdss, DFSMSHsm, and DFSMSrmm

What is DFSMStvs?

- RLS allows batch sharing of recoverable data sets for *read* (but only with NSR and shareoption 2)
 - ▶ RLS provides locking and buffer integrity
 - ▶ CICS provides logging and two-phase commit protocols
- Transactional VSAM allows batch sharing of recoverable data sets for *update*
 - ▶ Logging provided using the System Logger
 - ▶ Two-phase commit and back out provided using z/OS Recoverable Resource Management Services (RRMS)
 - ▶ Commit and back out performed by Transactional VSAM
 - ▶ Allows CICS online applications and *multiple* batch applications to run concurrently

The Value of Transactional VSAM

- Eliminate the batch window
 - ▶ Concurrent shared **update** of VSAM recoverable files across CICS and batch
 - ▶ Ability to run batch jobs for **update concurrently** instead of serially
 - ▶ Allows backup-while-open using DFSMSdss and DFSMSHsm
- Data Integrity and recoverability
 - ▶ Utilizes z/OS Resource Recovery Services (RRS)
 - ▶ Provides locking, logging, commit and backout
 - ▶ Uses the same forward recovery logs as those used by CICS TS and supported by CICSVR
- Increases availability of your business applications
 - ▶ Enables 24x7 CICS Transaction Servers (TS) applications
 - ▶ Parallel sysplex data sharing across CICS and batch
- Allows programs to interact with multiple resource managers such as IMS and DB2

Extending the Availability of CICS Applications

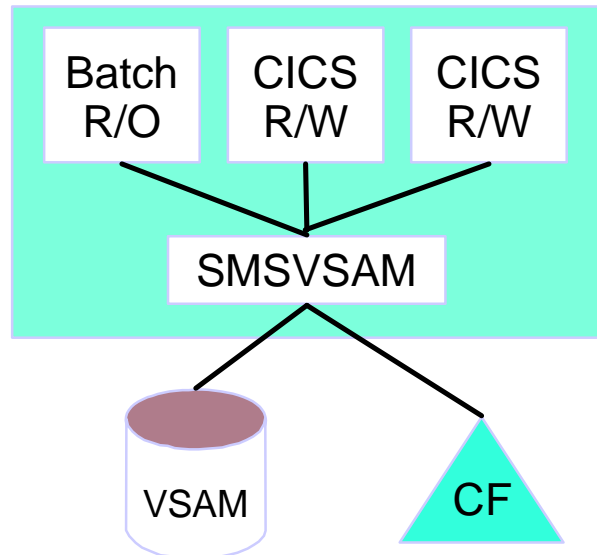
- Enhances Parallel Sysplex VSAM Data Sharing
 - ▶ DFSMStvs provides a sharing capability between online (CICS) and batch
 - ▶ Enables 24x7 operations when fully exploited
- "Transactionalizes" Batch Access to VSAM Data Sets
 - ▶ Provides two-phase commit (and backout) capability
 - ▶ VSAM becomes a data base like IMS DL/1 and DB2
- Exploitation
 - ▶ May require changes to applications to fully exploit
 - ▶ See DFSMStvs Application Migration Guide (SG24-6972)

TVS Customer Value

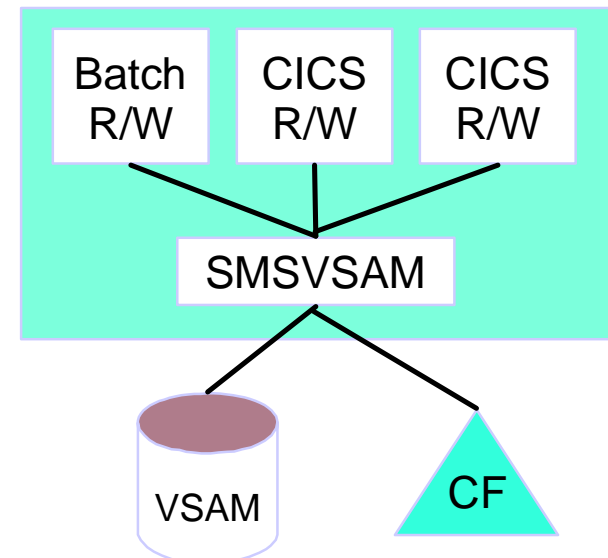
- Keep CICS online available for update while running batch updates
 - 24 x 7 application availability
 - multiple batch jobs can be run concurrently
- Use VSAM data without moving or converting the data
 - **Savings:** Customers **save tens of millions of dollars** by not converting VSAM data to DB2
 - An insurance company with a \$20M annual IT budget spent \$10M and 2 years to convert 40% of their VSAM data to windows platform and swear they will not convert the remaining 60%
 - DB2 Migration vendors charge \$1,000 per program (average) excluding testing to run DB2 transparency
 - Reduce complexity in application development

VSAM Data Sharing - RLS

Recoverable Files



Non-Recoverable Files

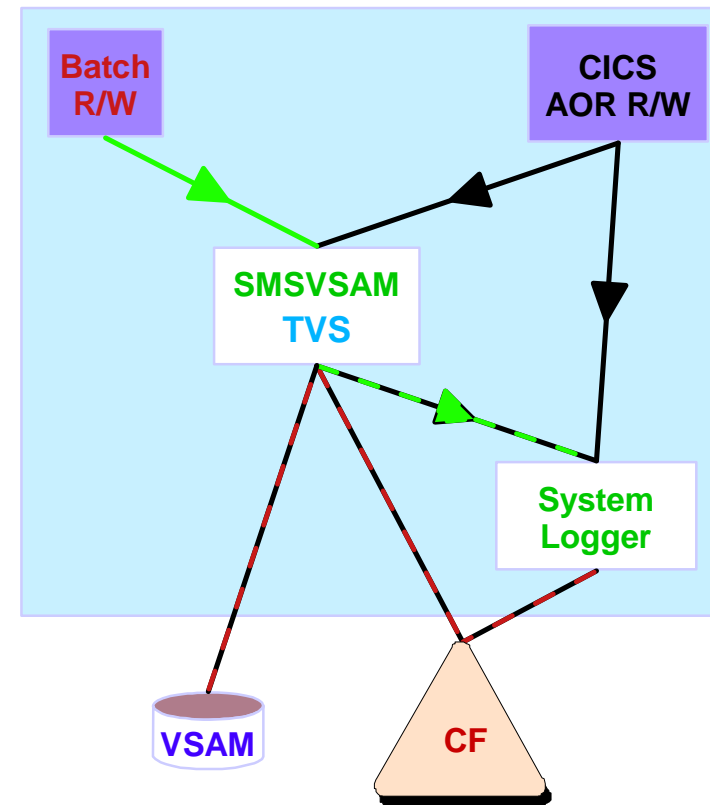


With RLS:

- ✓ Recoverable Files: Batch has Read-Only Access
- ✓ Nonrecoverable File: Batch has Read/Write Access

VSAM Data Sharing - Transactional VSAM

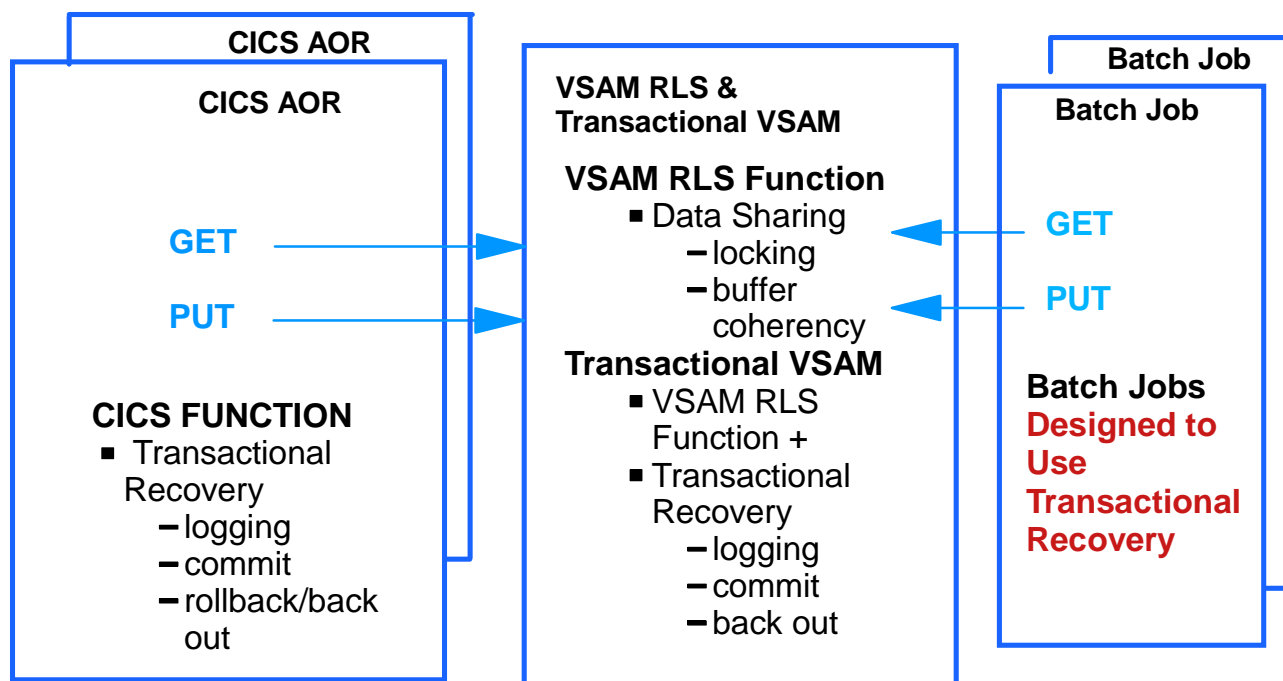
- **Allows batch update sharing concurrent with CICS on-line access to recoverable data**
- Allows multiple batch update programs to run concurrently anywhere in the sysplex against the same files
- Allows transactional support for non-CICS applications to share VSAM data with integrity
- Priced Feature of z/OS 1.4 or later
 - ▶ Announced May 2003
 - ▶ Available June 2003



Accessing a Data Set with Transactional VSAM

- Define VSAM data set as recoverable
 - ▶ LOG(UNDO)
 - ▶ LOG(ALL)
- Specify RLS access in the ACB
 - ▶ Open for input with read integrity option CRE *or*
 - ▶ Open for output
- Interface with the RRS component of RRMS to define sync points (commit or back out)

Using Transactional VSAM



Concurrent Read/Write Sharing of **RECOVERABLE FILES**
Across CICS and Batch Jobs

What is a Transaction?

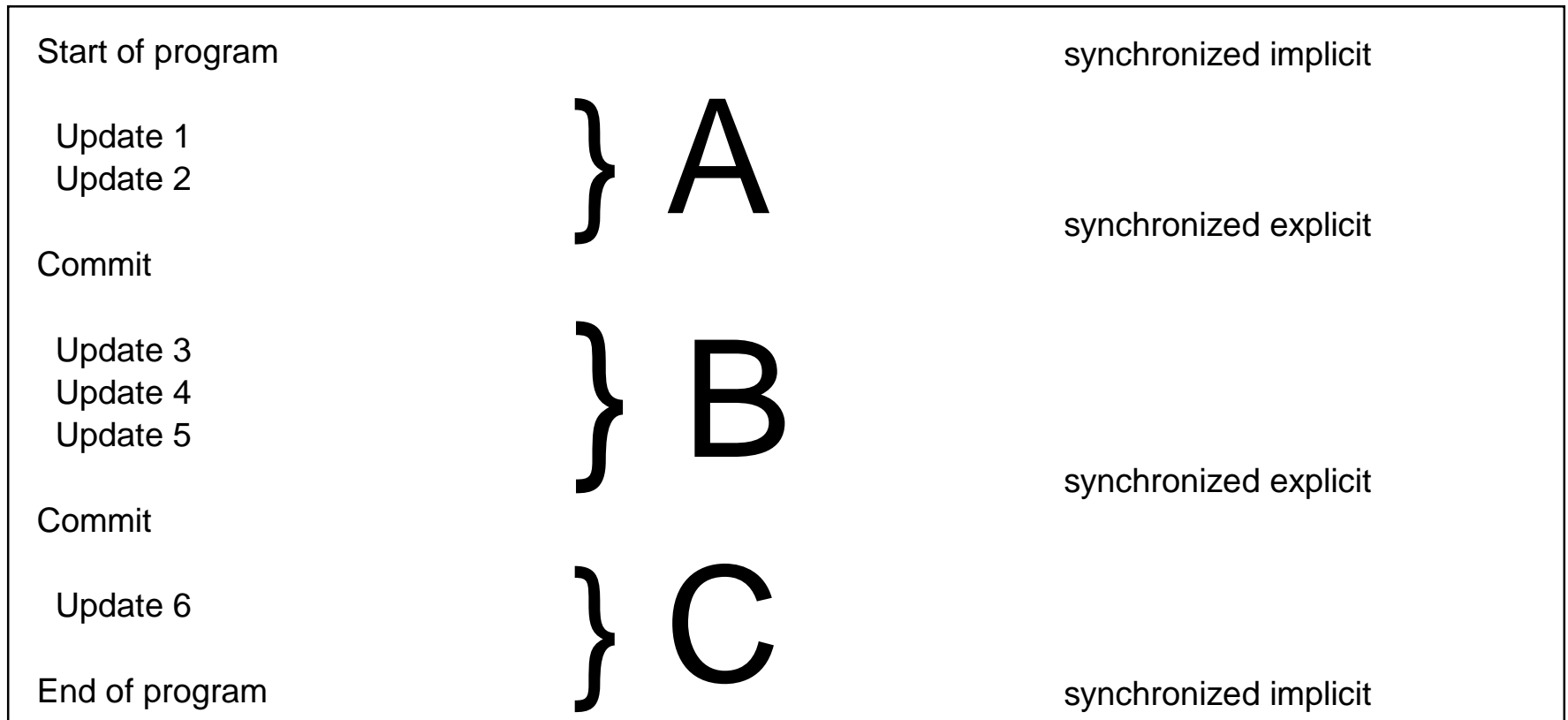
Application

Result

Start of Transaction -----	No locks held
GET UPD record 1-----	Obtain an exclusive lock on record 1
PUT UPD record 1 -----	Lock on record 1 remains held
GET repeatable read record n-----	Obtain a shared lock on record n
PUT ADD record n+1-----	Obtain an exclusive lock on record n+1
GET UPD record 2 -----	Obtain an exclusive lock on record 2
GET UPD record 2 -----	Lock on record 2 remains held
Call SRRCMIT -----	Commit changes, all locks released

Unit of Recovery

- Set of changes processed by a single commit of backout is call a *unit of recovery*



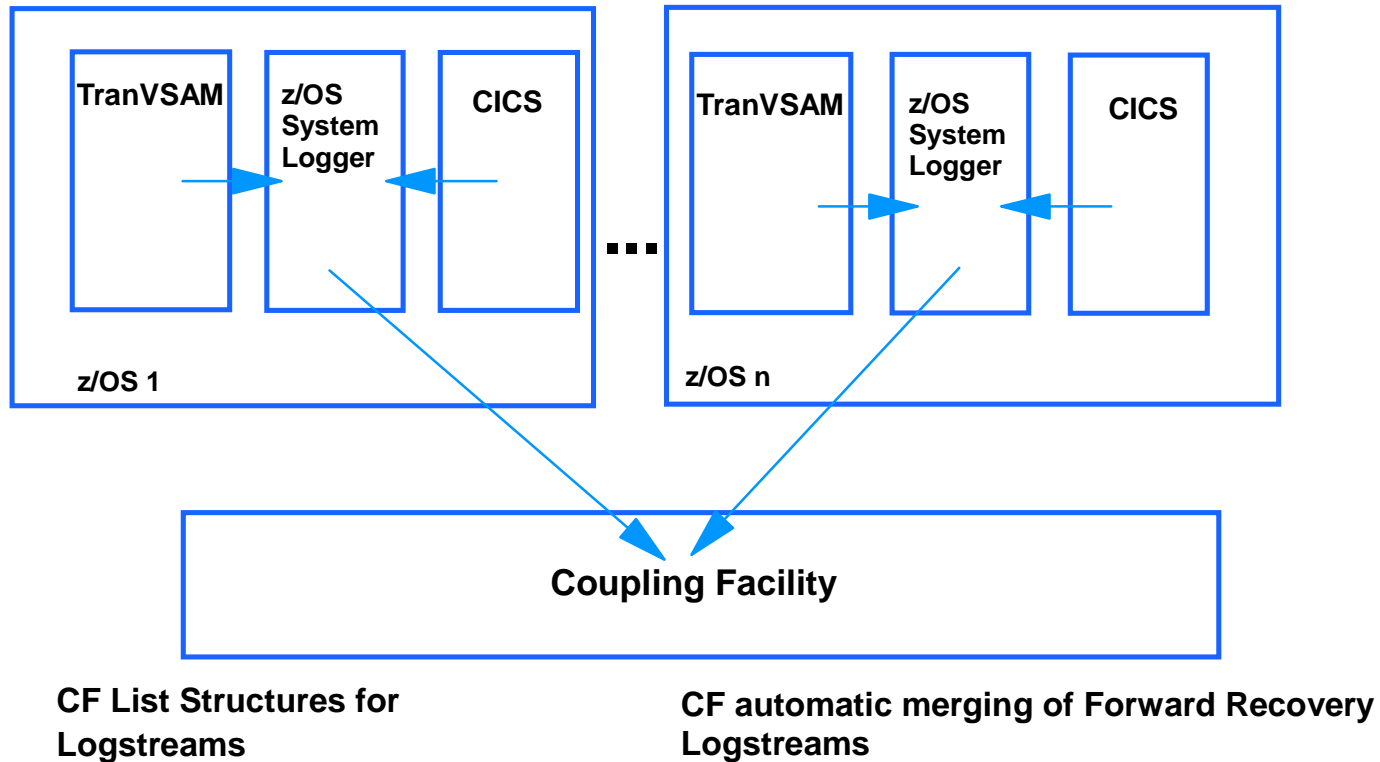
Transactional VSAM Log Streams

- Each z/OS image has two System Log Streams
 - ▶ Short-lived transactions (primary system log)
 - ▶ Long-lived transactions (secondary system log, or shunt log)
- Forward Recovery Log Streams
- Log of Logs (can be same as CICS)
- All log streams must be predefined before Transactional VSAM is started

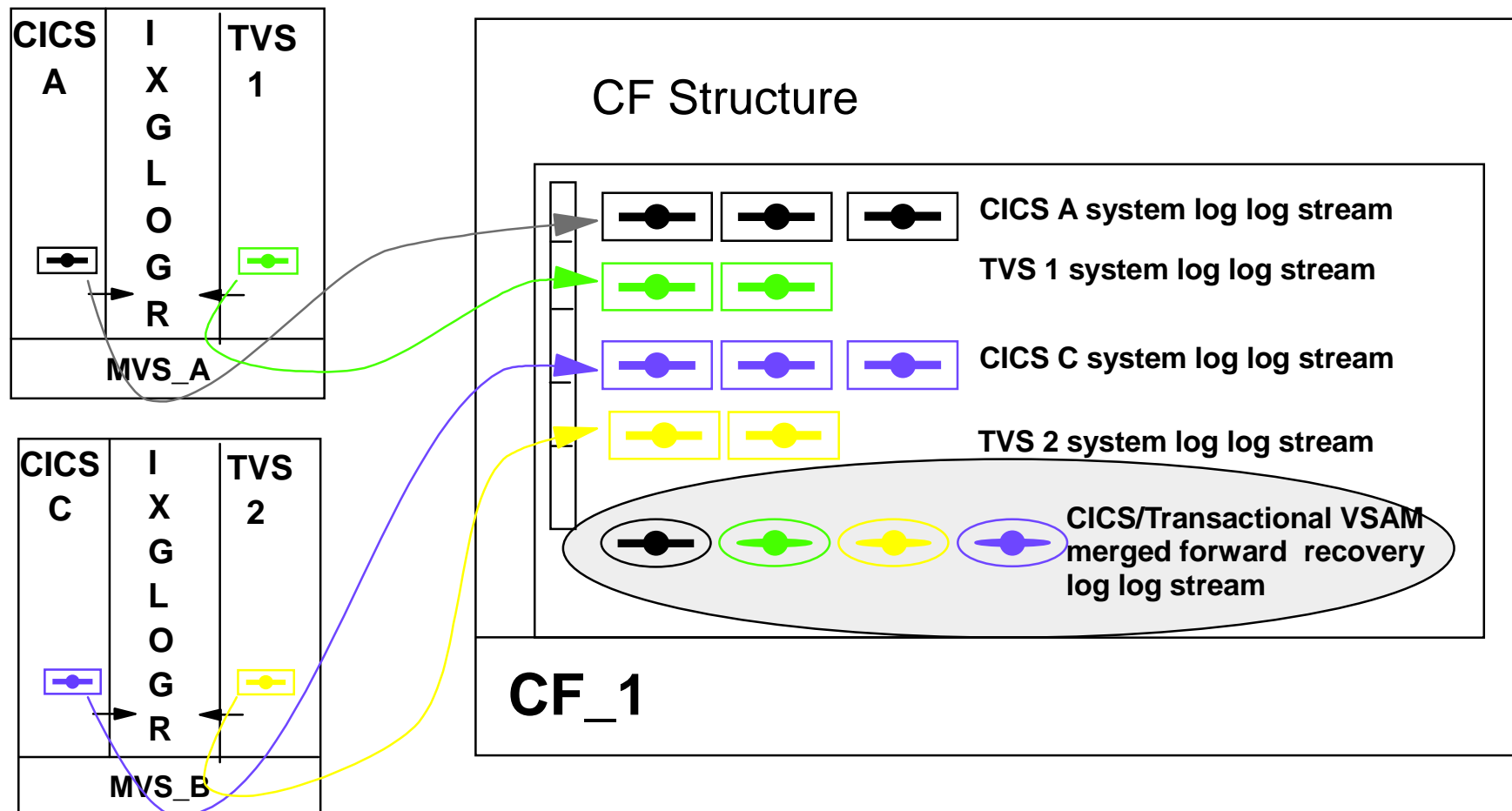
Forward Recovery Logging

- Provided only for data sets defined with LOG(ALL)
- The forward recovery log is an attribute of the data set specified via LOGSTREAMID
- Transactional VSAM and CICS use the System Logger to log changed data
 - ▶ They both write to the same forward recovery log(s) and logs of logs
 - ▶ CICSVR does the forward recovery process

Transactional VSAM Logging



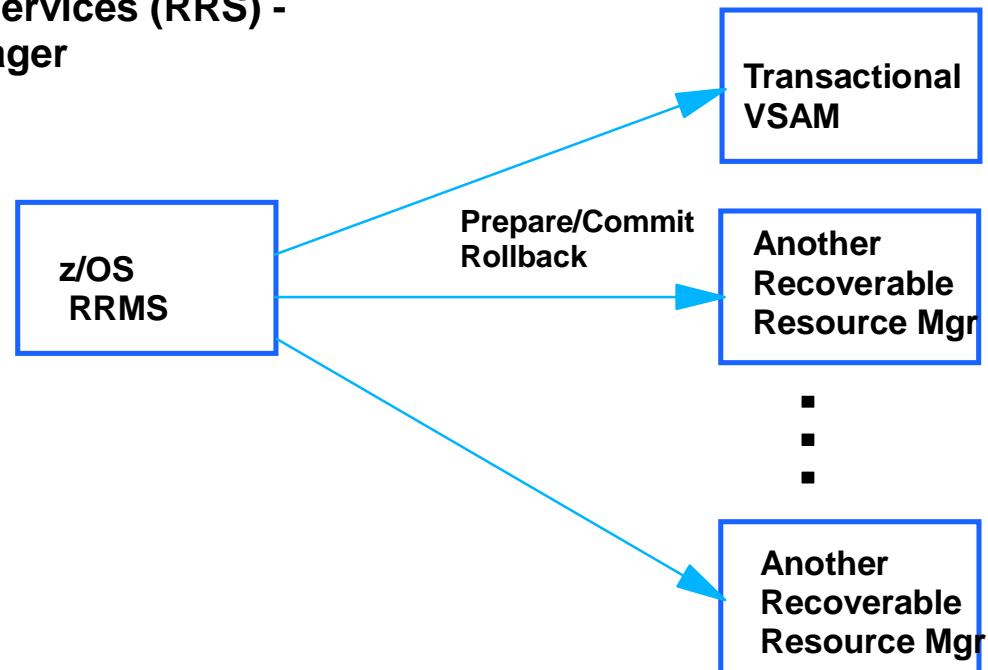
CICS/Transactional VSAM Logstreams



Transactional VSAM & z/OS RRS

z/OS RRMS:

- Registration Services
- Context Services
- Resource Recovery Services (RRS) -
z/OS sync point manager



SYS1.PARMLIB Changes

SMS	ACDS(acds) INTERVAL(nnn 15) REVERIFY(YES NO) SYSTEMS(8 32) SIZE(nnnnnK M) JOBNAME(jobname *) SELECT(event,event....) DSNTYPE(LIBRARY PDS) RLSINIT(NO YES) SMF_TIME(NO YES) BMFTIME(nnn 3600) DEADLOCK_DETECTION(iii 15,kkk 4) SYSNAME(sys1,sys2....) TV_START_TYPE(WARM COLD,WARM COLD...) LOG_OF_LOGS(logstream) MAXLOCKS(max 0,incr 0)	COMMDS(commnds) DINTERVAL(nnn 150) ACSDEFAULTS(YES NO) TRACE(OFF ON) TYPE(ALL ERROR) ASID(asid *) DESELECT(event,event....) DSSTIMEOUT(nnn 0) RLSMAXCFFEATURELEVEL(A Z) RLS_MAX_POOL_SIZE(nnn 100) CF_TIME(nnn 3600) CACHETIME(nnn 3600) RLSTMOUT(nnn 0) TVSNAME(nnn1,nnn2....) AKP(nnn 1000,nnn 1000) QTIMEOUT(nnn 300)
------------	---	---

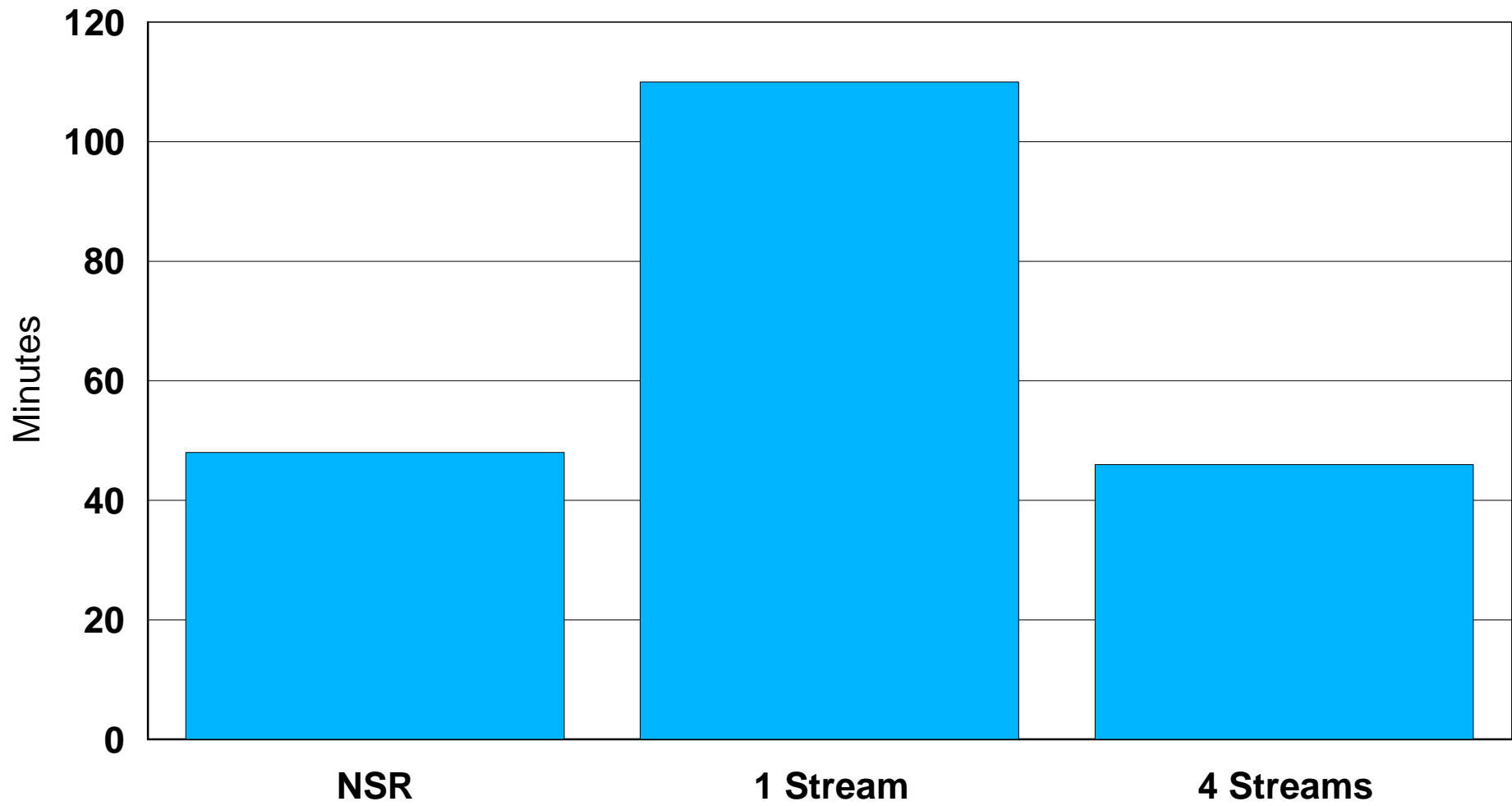
Supported Languages

- **PLI**
- **C & C++**
- **COBOL**
- **Assembler**

Performance

- Extra Overhead TranVSAM vs NSR or LSR
 - ▶ Cross-Address Space access to server
 - ▶ Loss of NSR chained sequential I/O
 - ▶ Loss of LSR deferred write
 - ▶ New overhead of record locking
 - ▶ New overhead of CF cache access
 - ▶ New overhead of logging
- Expect Transactional VSAM overhead to be similar to VSAM file access via CICS RLS

Performance -- Parallelizing the Workload



Next Steps - Planning for Transactional VSAM

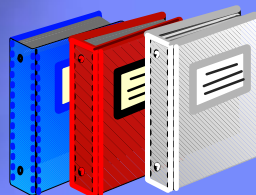
- z/OS 1.4 required
 - ▶ OS/390 2.10 End of Service --- September, 2004
 - ▶ www.ibm.com/zseries/zos/migration
- Coupling Facility Structures
- Processing Capacity
 - ▶ More CPU cycles consumed
- Reevaluate Job Scheduling
- Defining Log Streams
- Restart and VSAM Recovery Procedures

Transactional VSAM Summary

- Transactional VSAM is a general purpose VSAM recoverable file manager for z/OS Transactional VSAM addresses a long-standing CICS batch window requirement
 - ▶ Read/write sharing across CICS and batch jobs
 - ▶ Read/write sharing across multiple batch jobs
 - ▶ Major steps towards 24x7 CICS on-line availability
- Use of RRS as sync point manager enables commit across Transactional VSAM and other recoverable resource managers
 - ▶ DB2
 - ▶ MQ

Appendix A:

References



Information about TVS

Information about DFSMS and TVS

- www.storage.ibm.com/software/sms/index.html
- www.storage.ibm.com/software/sms/tvs/index.html

Additional Information

- www.redbooks.ibm.com
 - Transactional VSAM Presentation Guide SG24-6973
 - Transactional VSAM Overview and Planning Guide SG24-6971
 - Transactional VSAM Application Migration Guide SG24-6972
 - VSAM Demystified SG24-6105

References

- ***z/OS MVS Setting Up a Sysplex - SA22-7625***
- ***z/OS MVS Programming: Resource Recovery - SA22-7616***
- ***z/OS MVS Diagnosis: Tools and Service Aids - GA22-7589***
- ***z/OS MVS Programming: MVS Assembler Services Guide - SA22-7605***
- ***z/OS MVS Programming: Authorized Assembler Services Reference, Volume 2 - SA22-7610***
- ***OS/390 Parallel Sysplex Configuration Cook Book, Volumes 1-3 - SG24-2075, SG24-2076, SG24-2077***
- ***CICS Recovery and Restart Guide, SC33-1698***

Notices & Disclaimers

Copyright © 2003 by International Business Machines Corporation.

No part of this document may be reproduced or transmitted in any form without written permission from IBM Corporation.

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 product(s) and/or program(s) 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 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.

Trademarks

DFSMSdfp, DFSMStvs, 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, Seascope, 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.