

SRDF/A Best Practices



Kevin Batty

Agenda

- Acknowledgements and attributions
- SRDF/A Overview
 - Basic SRDF/A
 - Understanding SRDF/A Consistency
 - SFDF/A – MSC
- SRDF/A Best Practices
 - Planning
 - Performance Considerations
 - General Operation
 - Usage in z/OS Environments
 - Usage in Open Systems Environments
 - Troubleshooting

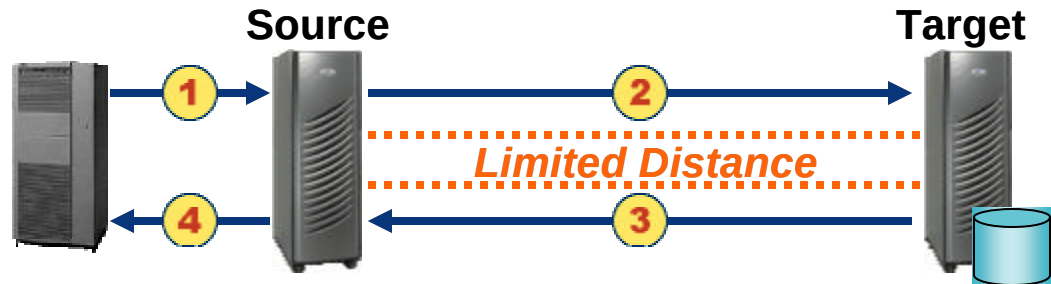
SRDF/A Overview



The Most Complete Remote Replication Family of Products

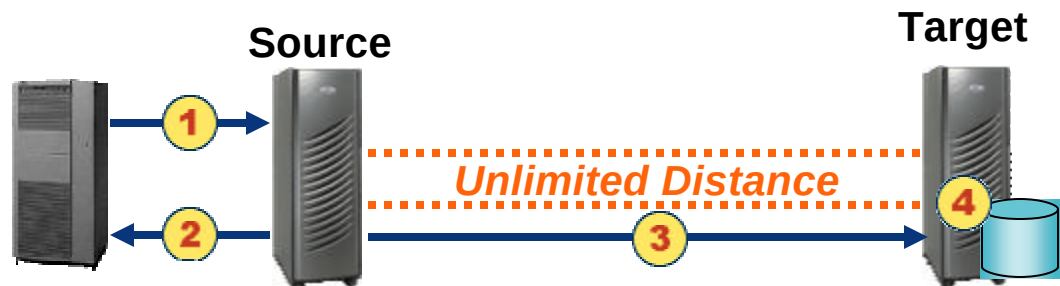
SRDF

- No data exposure
- Some performance impact
- Limited distance



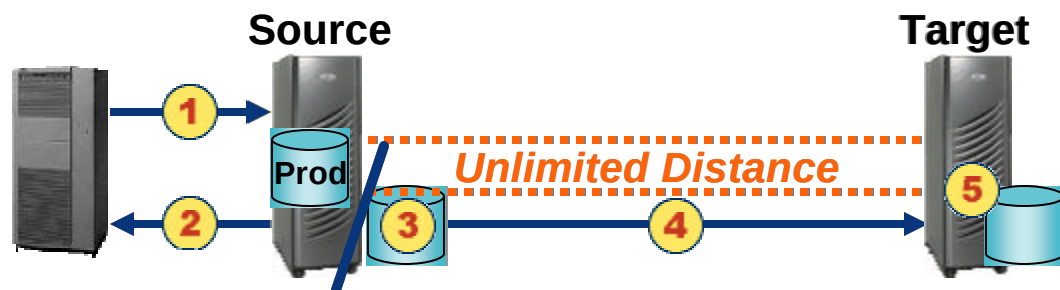
SRDF/A

- Seconds of data exposure
- No performance impact
- Unlimited distance

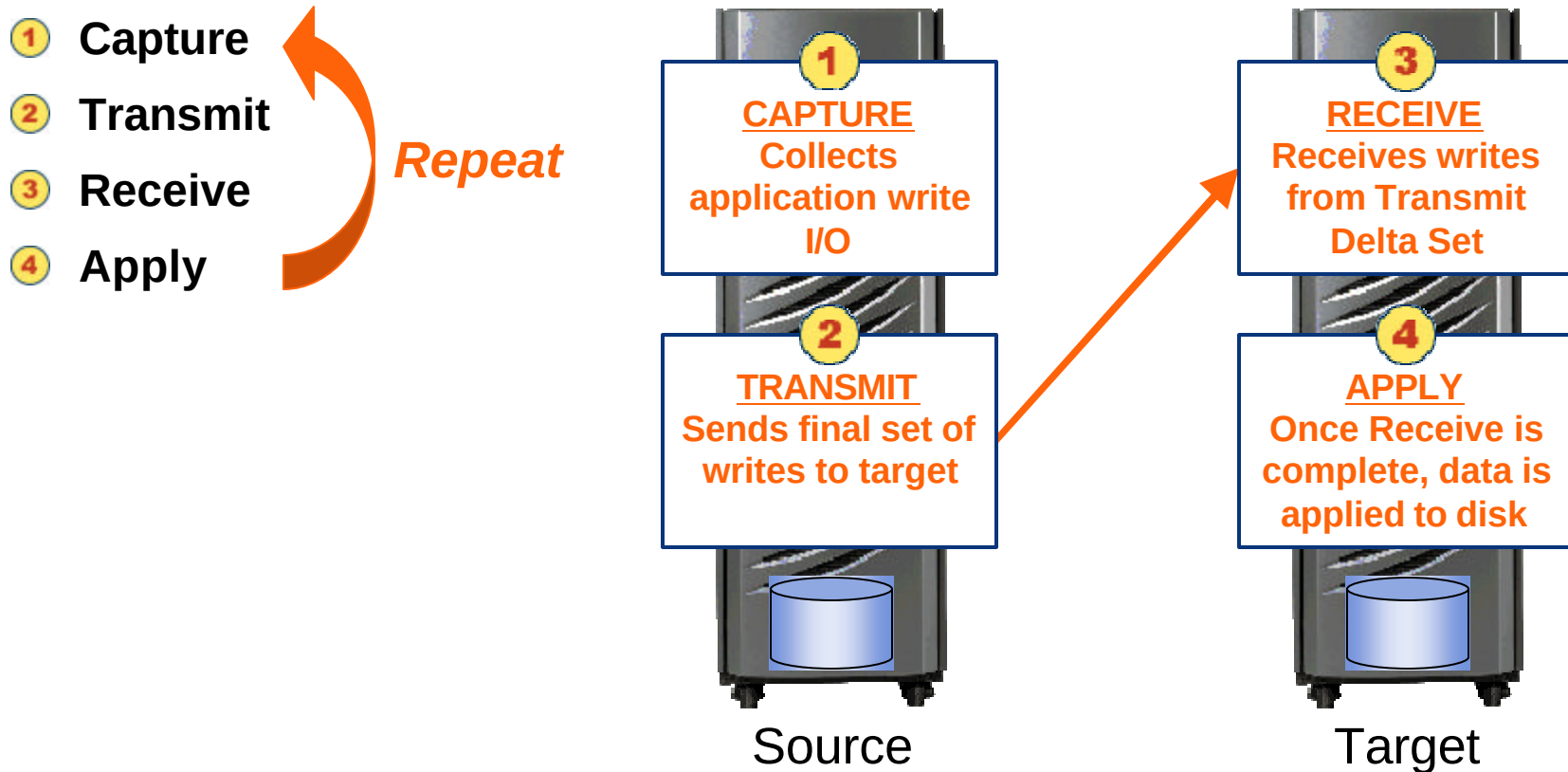


SRDF/AR

- Hours of data exposure
- No performance impact
- Unlimited distance
- Requires BCVs



SRDF/Asynchronous Operation



SRDF/A performs “Write Folding”—only sends Transmits of the very last writes from the Capture Delta Set

Understanding SRDF/A Consistency

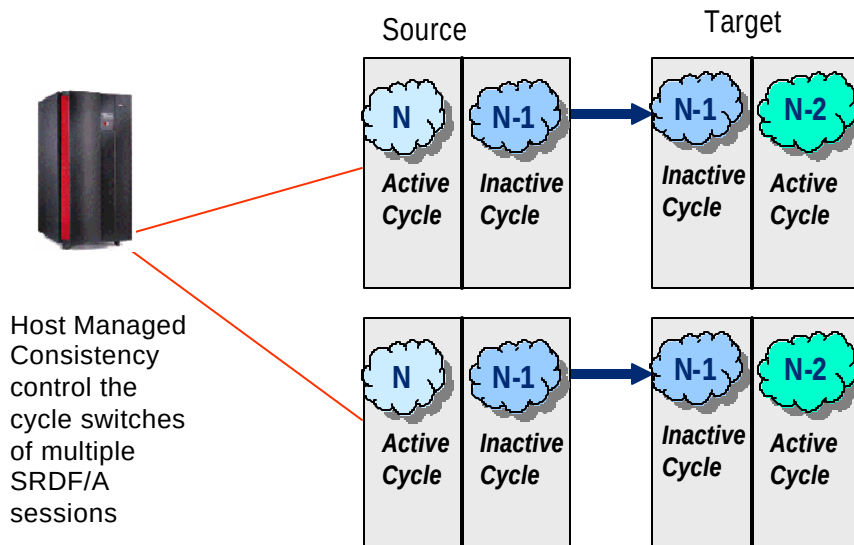
The Dependent Write I/O Principle

- The logical dependency between write I/Os; that is embedded by the logic of an application, operating system, or database management system
 - **A write I/O that will not be issued by an application until a prior related write I/O has completed**
 - A **logical** dependency, not a **time** dependency
 - Inherent in all database management systems (DBMS)
 - Page (data) write is **dependent write I/O** based on a successful log write
 - Applications can also use this technology
 - Power failures create a **dependent write consistent** image
 - Restart transforms **dependent write consistent** to **transactionally consistent**
- Ensuring “dependent write consistency” is the basic principle behind all of EMC Consistency Technology solutions
 - Enterprise SRDF/S Consistency Groups
 - Enterprise TimeFinder Consistent Split
 - EMC Consistent TF/Snap, TF/Mirror (clones) and Open Replicator
 - SRDF/A
 - MirrorView/A and MirrorView/S with Consistency Groups

A Word about SRDF/A Cycle Switching

- SRDF/A cycle switches cannot occur until the following events have occurred:
 - Minimum cycle time has elapsed
 - Transmit process is complete
 - Apply (Restore) process is complete

What is SRDF/A Multi Session Consistency (MSC)?

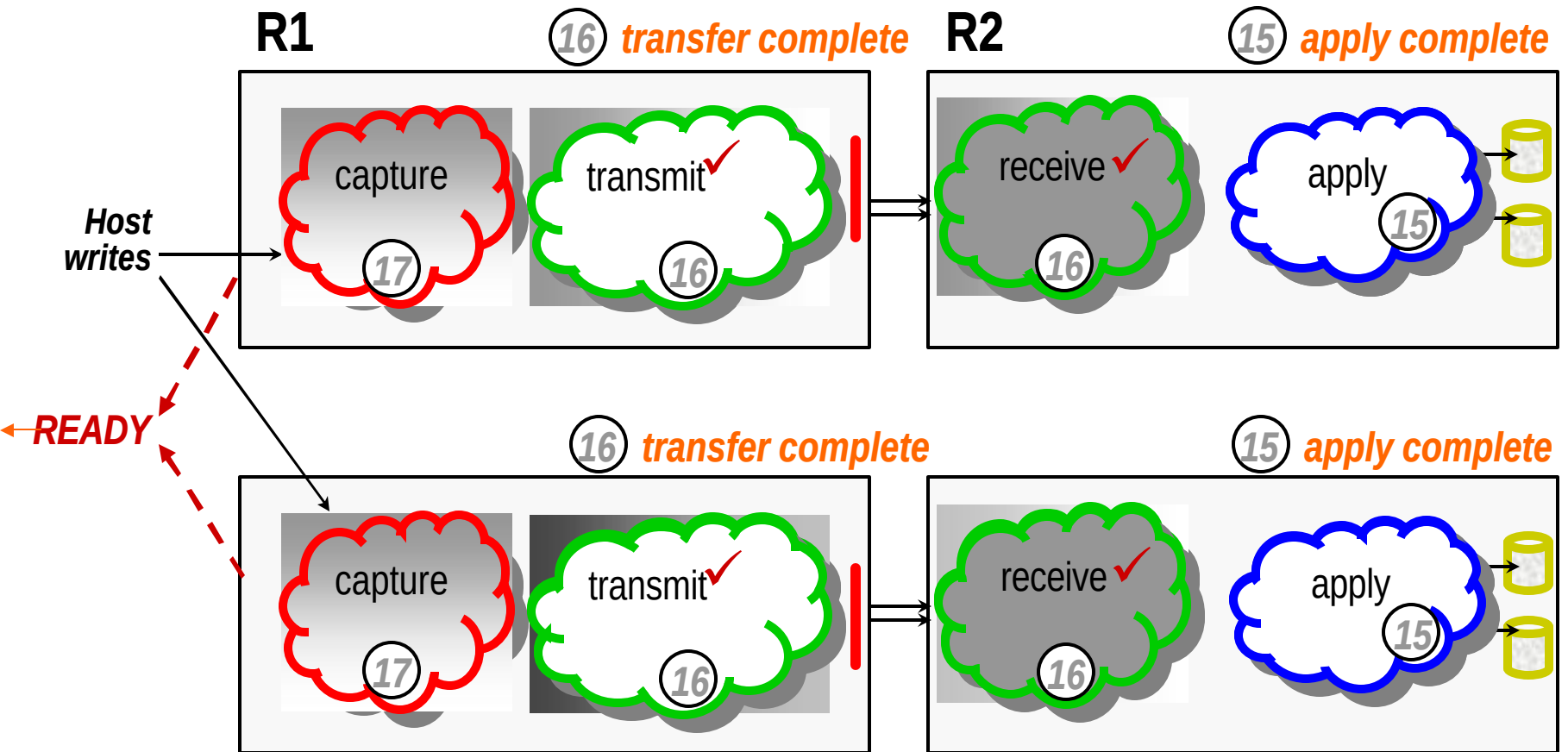


- Enhancement to SRDF/A that allows host software to ensure remote R2 consistency across multiple Symmetrix systems running SRDF/A.
 - Ideally involves an equal number of Symmetrix systems at primary and secondary sites
 - Usually there is one SRDF/A session per Symmetrix, (i.e., one SRDF/A group per Symmetrix)
 - Can be controlled from either open systems or mainframe
 - Enterprise support is available

Cycle Switching in MSC Environments – Ready Condition

- Each Symmetrix pair is ready for cycle switch, when the following two conditions are met:
 - Target (R2) frame has finished its apply (restore) cycles
 - Target (R2) frame has received the entire transmit cycle from the source (R1) frame
 - Implies that there are no outstanding host writes owed to the transmit cycle on the source side

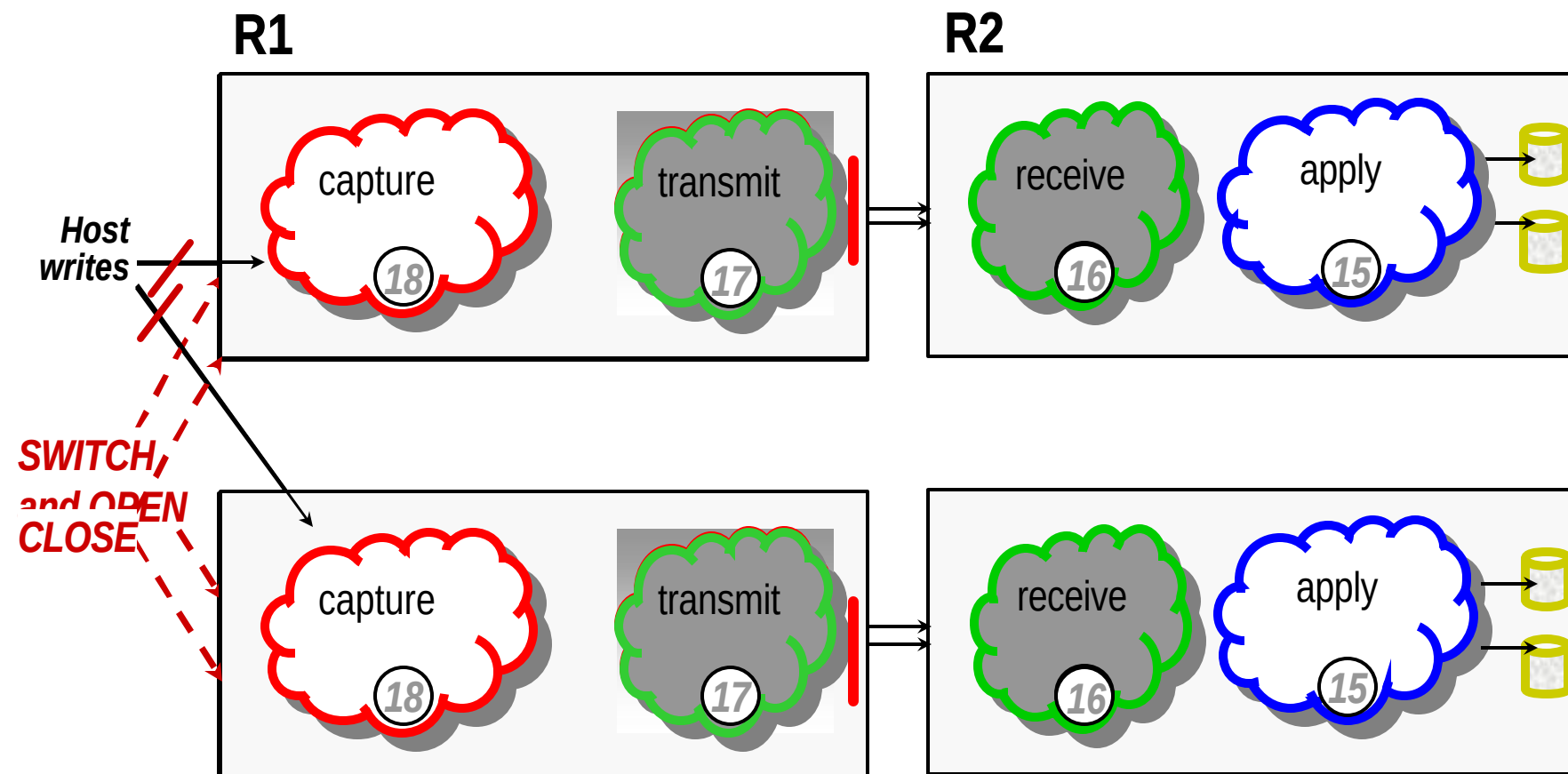
Cycle Switch Process – Ready Condition



Cycle Switch Process – Host Software

- The SRDF/A Multi Session Consistency feature (MSC) polls all source Symmetrix units to check their readiness for cycle switch
- When all source Symms are ready and internal conditions are satisfied, MSC performs the following steps:
 - Send switch and open command to all source Symms
 - Once all Symms executed switch and open command, close SRDF/A window on all boxes
 - Normal write I/O behavior returns (no longer disconnect on write)

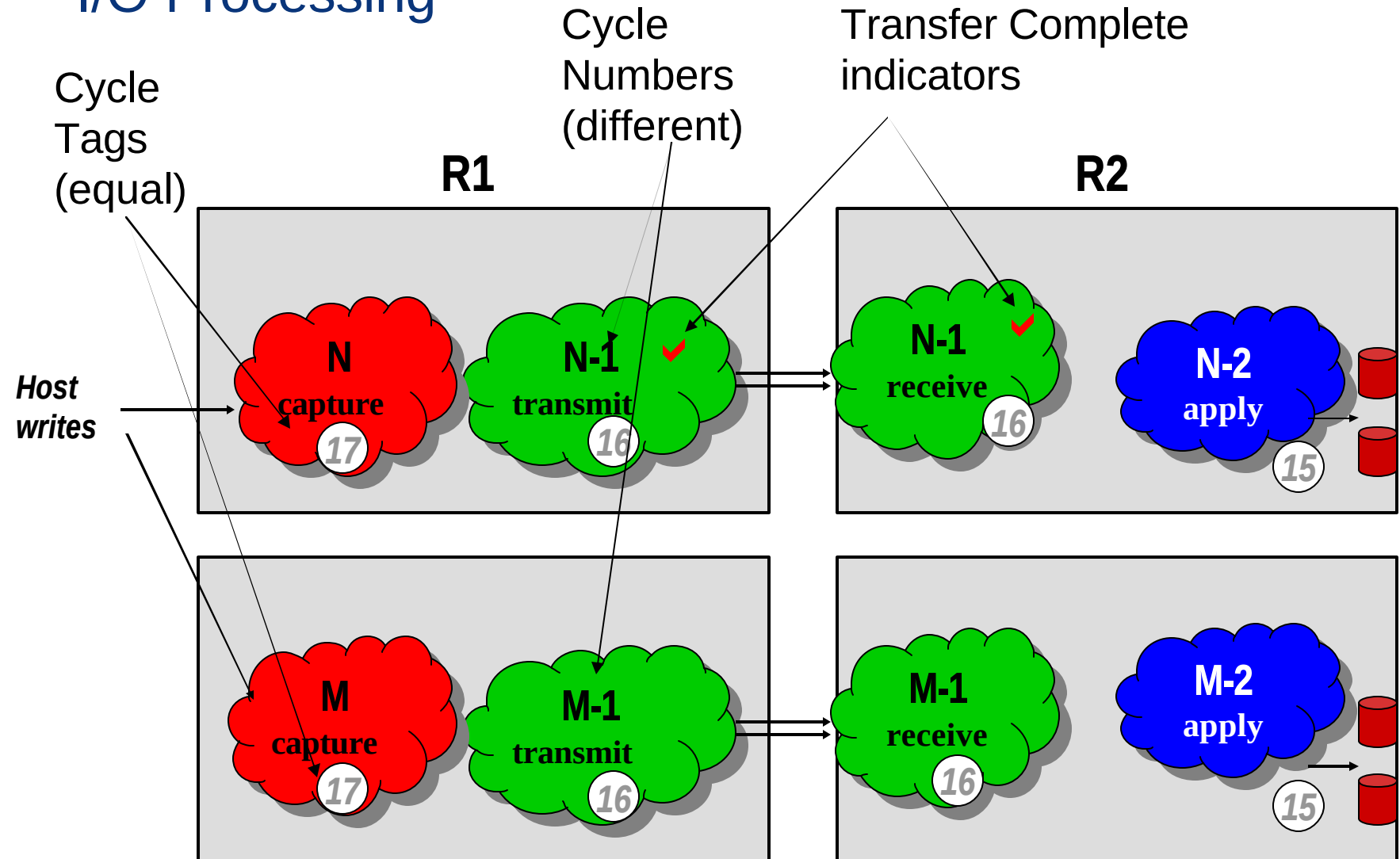
Cycle Switching



Cycle Numbers and Tag IDs

- A cycle number is a sequential number created by the microcode and associated to a cycle.
- The cycle numbers are going to be different in the multiple Symmetrix pairs.
- When MSC is active, a TAG ID is created by the MSC task and associated to each cycle, the Host Cycle.
- The Tag ID's are used to verify that the multiple SRDF/A sessions in the MSC group are coordinated.

I/O Processing



The SRDF/A Window

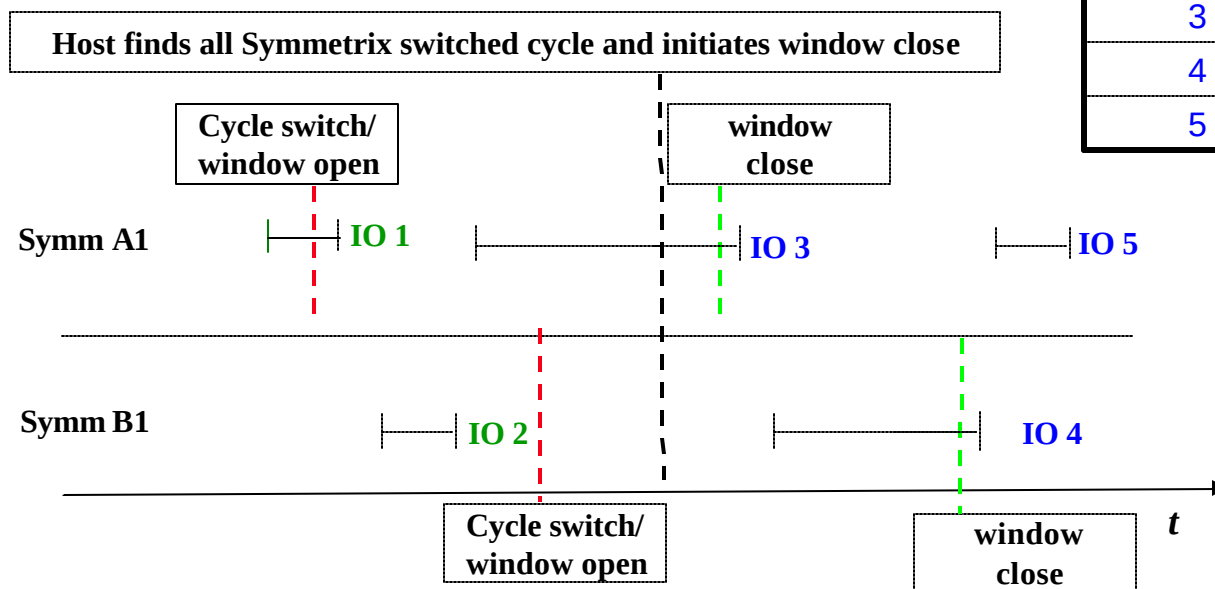
- The Window is automatically opened during cycle switch via a single “switch and open” Syscall
 - Command issued in parallel to all Symmetrix systems (performance)
 - New cycle tag assigned to CAPTURE cycle
 - Once opened on all systems, CLOSE Syscall is issued
- Very low impact
 - A single bit in the SRDF/A state table in global memory
 - No additional overhead to SRDF/A vs. single session mode

How SRDF/A MSC Ensures Consistency

- When SRDF/A window is open, Symmetrix Host Adapter (front end) disconnects on write
- Prevents dependent writes to SRDF/A devices by not allowing the previous write to complete
 - Allow reads (until the first write)
- Failsafe: Window automatically closed after 12 seconds (“timeout”)
 - SRDF/A is dropped if the window is closed by timeout. MSC behavior governed by MSC_DROP_Policy

Ensuring Data Consistency

- Dependent write/cycle switch/window interaction



I/O	Cycle
1	transmit
2	transmit
3	capture
4	capture
5	capture

Best Practices



Planning for SRDF/A

- SRDF/A requires planning: guess work will not do!
 - Evaluate the write workload (ECC recommended)
 - Logical volume with the most data to transmit will determine cycle time
 - Understand the characteristics of the workload being replicated
 - Determine the minimum required cache and bandwidth
 - EMC Field personnel have tools available to do this
 - Choose periods of peak workload for modeling intervals and try to allow for future growth
 - Understand your requirements for RPO and RTO
 - Allow for the impact of synchronous remote replication in the measured environment
- Never guarantee “No more than 60 sec “secondary delay”...”
 - SRDF/A cycle times are dynamic and depend on the workload from the host
- Make sure proposed Symmetrix configurations will sustain the load
- Keep it simple

Best Practice – Consider using an EMC Services Engagement!

SRDF/A Performance Considerations

- Writes are buffered in cache
- Balanced configuration - equal performance on both sites
- Locality of reference - reduces link and cache requirements



Best Practice – SRDF/A link bandwidth provided has to be equal or greater **on average** to the write input bandwidth

The Importance of Building a “Balanced Configuration”

- Source and Target environments need to be configured so that the Target has at least equal performance to the Source on a logical volume to logical volume basis – e.g.
 - Source RAID 5 to Target RAID 5 – OK
 - Source RAID 10 to Target RAID 5 – not OK
 - Source 2048 logical volumes to Target 2048 logical volumes – OK
 - Source 2048 logical volumes to Target 2048 logical volumes PLUS 2048 BCV – possibly not OK
 - Source 96 GB cache with 2048 logical volumes to Target 128 GB cache with 2048 logical volumes – probably OK
 - Source 96 GB cache with 2048 logical volumes to Target 96 GB cache with 2048 logical volumes PLUS 2048 BCVs – probably not OK

Best Practice – To achieve robustness, always ensure that the target environment can perform as well or better than the source

SRDF/A General Operation

- Large synchronization operations with SRDF-A active are *not recommended*

Best Practices –

Use adaptive copy modes for synchronizations, with SRDF/A inactive

Use SRDF QoS as necessary to minimize host impacts

- After an SRDF/A Drop, the microcode cleanup function runs on the SRDF directors as a background distributed task
 - During this cleanup, syscalls and inline utilities which manipulate valid/invalid status of tracks (and other potentially dangerous operations, e.g., removal of Dynamic Mirrors) are blocked in the microcode
 - In MSC mode, the cleanup running indication will be presented until host cleanup (intervention) is performed.

Best Practice –

Any commands that either you or your automation routines issue to change the environment or influence data movement during the microcode cleanup process will fail – don't be surprised!

SRDF/A General Operation

- Consider that unusual circumstances may cause a requirement for more hardware resources than recommended by the sizing tools – e.g.
 - Drive Replacements
 - Network Issues
 - Board Replacements
 - NDU code loads
 - TimeFinder/BCV
 - TimeFinder/Clone
 - Others

Best Practice – If continuous availability is a priority, over configure the environment to provide resiliency

SRDF/A General Operation

- SRDF has two modes of operation
 - Primary Mode – can be Synchronous or Semi-Synchronous
 - Secondary Mode
 - Adaptive Copy Write Pending
 - Adaptive Copy Disk Mode
- SRDF/A will drop into the SRDF Primary Mode after normal or abnormal terminations.

Best Practice –

Do not forget to change volumes to Adaptive Copy Disk Mode or Adaptive Copy Write Pending mode prior to resuming SRDF!

SRDF/A General Operation

Best Practices –

- Manually test all possible Dynamic Configuration changes and intermediate states before building automated changes
- Request a qualified and trained EMC hardware rep to be made available to help with troubleshooting
- Use the Escalation Process
 - If you run into anything that is outside of the “NORM” then open a case with EMC Software Support Center for review and tracking

SRDF/A in z/OS Environments

Best Practices –

- DO NOT replicate the following volume types
 - PAGE
 - TEMP
 - Couple DS (except LOGR)
- Make sure that SCF is started with “SUB=MSTR” so as to avoid timeout issues
- Use dedicated MSC Gatekeepers
 - Offline to all hosts
 - Not first volume in a Control Unit
 - Not used by any other Product

SRDF/A in z/OS Environments

Best Practices –

- Software Maintenance
 - Stay current on all host software products
 - Check for new software maintenance levels at least monthly
- MSC Cleanup
 - Run after every SRDF/A Termination
 - Ideally ensure that SFDF links are up prior to running
- MSC Group Name
 - Simple is generally best
 - Use name that is 8 characters or less
 - Remember that you will need to embed spaces in the cleanup job JCL if the MSC Group Name is less than 8 characters

Monitoring SRDF/A in z/OS Environments

Best Practices -

- Use ASY Monitor component of SCF for “real time” tracking of SRDF/A secondary delay
- Cut SMF user records for post processing

Open Systems SRDF/A and SRDF/A MSC Best Practice Parameter Settings

- Enable **gns_remote_mirror** for RDF2 type group propagation to remote side simplifies device group definition
 - DGs mirror BCV/Clone
 - CGs do not mirror BCV/Clone

Best Practice - Use Group Name Services!!!!

- Dynamic SRDF/A configuration parameters
 - Note : The Symmetrix level settings Max_Host_Throttle_Time and Max_Cache_Usage are mutually exclusive

Best Practice –

Do **NOT** use Maximum Host Throttle Time (**rdfa_host_throttle_time**)

- RDF group (aka SRDF/A session) level settings
 - Use to set cycle time and drop priority

Best Practice –

Remember Cycle Time setting is a **MINIMUM**, and that it can and will grow!!!

Open Systems SRDF/A and SRDF/A MSC Restart After Failover

Best Practice -

- GNS Enabled with Remote Mirroring
 - Automatically Propagates dg/cg to target host running GNS
 - Composite groups do not support BCV information
- Cleanup command
 - Perform at CG level if GNS used
 - Perform at RDF group if CG not available
- Consistent Split BCV/Clone to capture dependent write consistent point in time image
 - Consistent split invokes ECA
 - Consistent split performs sanity check
- R2s available

Open Systems SRDF/A and SRDF/A MSC Restoration Process – Driven From R2 Host

Best Practice -

With the applications still up on secondary side servers.....

- Change SRDF mode to Adaptive copy
- Perform R1 update until to a low number of tracks
- Depends on customer environment

Bring down applications

- Consistent Split BCV/Clone to capture dependent write consistent point-in-time image prior to resynchronization
- If performed on R2 side, consistent split is not necessary
- Begin synchronization from R1 to R2 in async mode
- When complete, enable consistency

Bring up applications on primary side servers

Configuration Changes in SRDF/A & SRDF/A MSC Env.

Best Practice -

- No configuration changes allowed while consistency enabled
- Consistent Split BCV/Clone to capture dependent write consistent point-in-time image prior to resynchronization
 - Consistent split invokes ECA/performs on a cycle boundary
 - Consistent split performs sanity check
- Disable consistency
- Switch mode to Adaptive Copy
- Perform configuration work
- Resynchronize R1 to R2
- At designation invalid track count
 - Switch mode to SRDF/A
 - Enable consistency

SRDF/A and SRDF/A MSC

Understanding Mainframe/Open Systems Variances

- Mainframe Tolerance Mode Off/On represents itself through the Enable/Disable of Consistency Enabling
 - SRDF/A and SRDF/A MSC both REQUIRE consistency enabled to create a dependent write consistent point in time copy!!!
- Open Systems SymCLI uses “Pure ready state”
- Open Systems SymCLI uses composite commands not making transition in and out of SRDF/A easier
- Commands (Pend Drop = split/suspend; Drop = split/suspend –immediate)

Monitoring SRDF/A in Open Systems Environments

- Symstat –type requests
 - Host Write Workload versus RDF Throughput
- Symstat –type cache
 - Cache utilization
 - SRDF/A max cache usage parameter
 - Session priority can be set to drop group with highest cache usage
- Symstat –type cycle
 - Verify cycle switching
 - Look at SRDF/A cycle sizes
 - Network issue
 - Slow R2 issue

Sampling SRDF/A Metrics

- Polling intervals are typically greater than cycle switch times
 - Data is not collected from every cycle
- Regular polling intervals do not collect data at same point in every cycle
- Not collecting peak data points
- Use at least 30 second interval times (-i 30)

Troubleshooting

Best Practice – Understand the technical meaning of SRDF/A error codes, and what they mean from a business perspective

- CACA.1x
 - Source system max write pending limit reached
- CACA.20
 - An SRDF/A target device has become Not Ready on the link
- CACA.30
 - Target SRDF/A device made not ready on the link at the R1 side
- CACA.40
 - All links to target frame were lost for a period of time greater than that set in the link limbo parameter in the binfile (default is 10sec)
- CACA.50
 - SRDF/A MSC mode cycle switch window did not close within Enginuity's required time (12sec)

Troubleshooting Tools Required

- ASCII text viewer for examining downloaded logs from hosts, switches, and routers
- ISPF for looking directly at mainframe syslogs and/or SCF and Host Component started task logs
- ECC Workload Analyzer
- Have full documentation available from implementation/last re-configuration, in a format which is shareable by all necessary parties, and which includes;
 - Hardware lists, including vendor and hardware/ucode levels
 - Software lists, including vendor, release levels, maint levels
 - Detailed cabling diagrams
 - Detailed host partitioning information
 - Make sure Symmetrix is at current Enginuity revision levels
 - Make sure Symmetrix remote dial in capability is working
 - For all systems involved
- EMC reps will also use:
 - Symmetrix Log Viewer
 - STP Navigator
 - SymmMerge

Troubleshooting Data Required

- Clarify case #
- Info on timestamp differences between hosts, Symms, and network devices involved
- Host syslog
- SCF and Host Component started task job logs for mainframe environments
- SymCLI symstat data for open systems environments
- Switch and router logs to diagnose network related issues
- WLA or STP data from all frames on both the source and target sides of the link, ideally at 2min intervals
- Symmetrix logall's from all frames involved (EMC use only)
- Binfiles from all frames involved (for SymmMerge analysis) (EMC use only)

Where To Go For Help....

Best Practices -

- In all circumstances open a Clarify case, and for problems during implementation escalate as necessary through your local EMC Technology Solutions and Customer Support Representatives.
- Ask you local EMC reps to engage your local EMC “Speed Guru” on performance related issues
- Host software issues need to be escalated via the SSC
 - Make sure you specify open or mainframe to ensure you get the right people working the case....
- Symm hardware and network issues need to be escalated via the PSE lab

Thank you



EMC²
where information lives[®]