DS8000 Replication Performance Considerations

Lisa Gundy

DFSMS Copy Services Architect

IBM Systems Division

#SHAREorg

SHARE is an independent volunteer-run information technology association that provides **education**, professional **networking** and industry **influence**.

SHARE in Seattle 2015

Agenda

Actions | Filter

- Replication Review
- Multiple Incremental FlashCopy
- Multi-Target PPRC Performance
- PPRC Synchronization
- Global Copy Collision Enhancement
- zHyperWrite ©
- Workload Based z/OS Global Mirror (XRC) Write Pacing
- Easy Tier Heat Map Transfer

DS8000 Replication Review

FlashCopy

Point in Time Copy

Within the same Storage System

Metro Mirror

Synchronous Mirroring

Primary Site A

Metro distance Site B

Global Mirror
z/OS Global Mirror

Asynchronous Mirroring

Primary Site A

Out of Region Site B

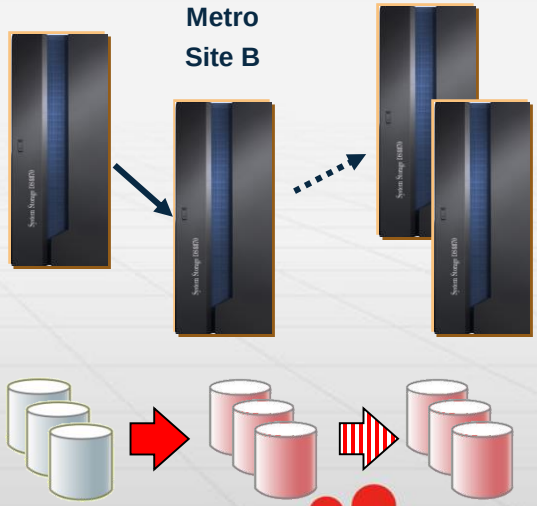
Metro Global Mirror
Metro z/OS Global Mirror

Three site and Four Site Synchronous & Asynchronous Mirroring

Primary Site A

Metro Site B

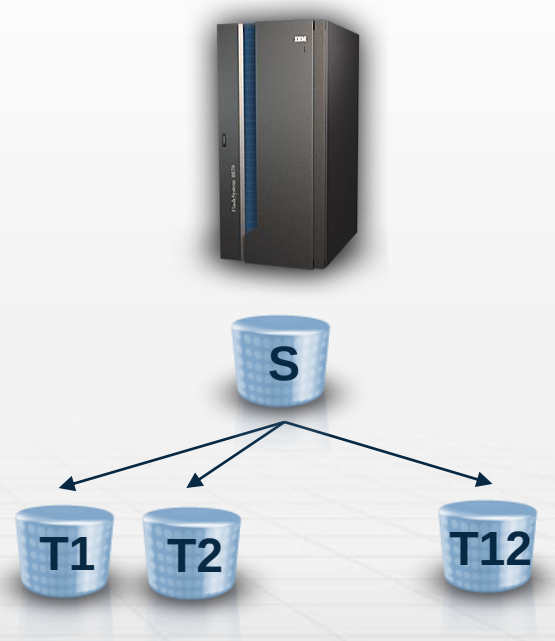
Out of Region Site C/D



Multiple Incremental FlashCopy

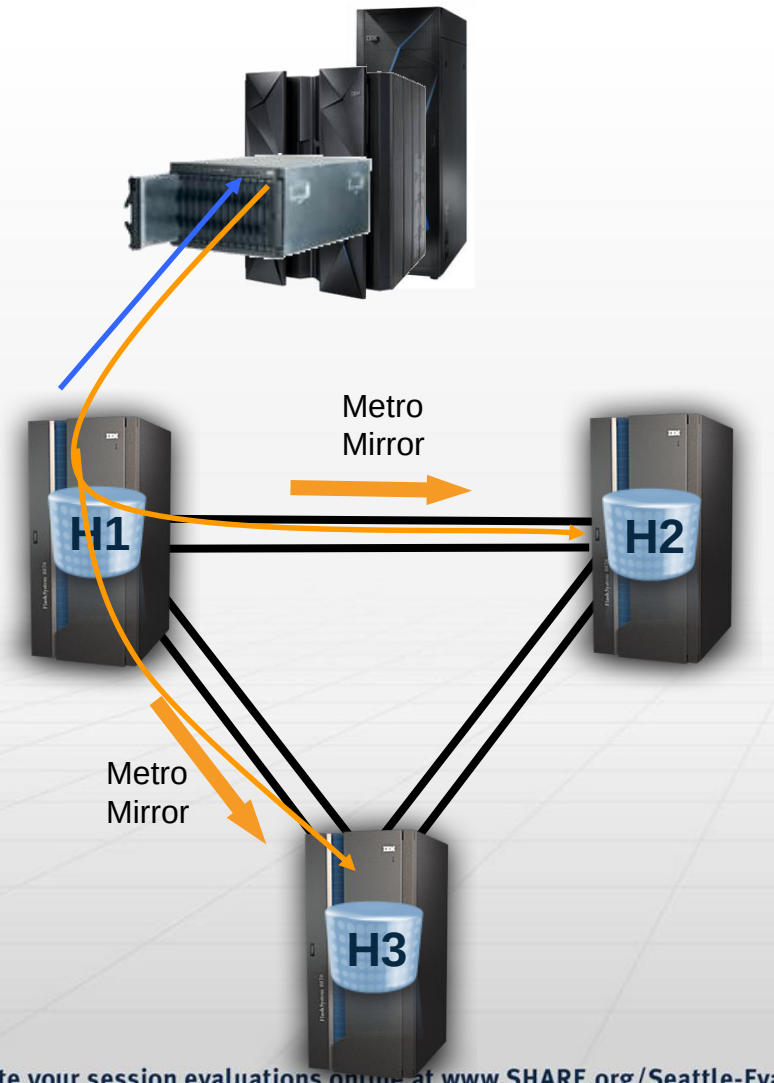
Multiple Incremental FlashCopy

- Previously only a single incremental FlashCopy was allowed for any individual volume
- This provides the capability for up to 12 incremental FlashCopies for any volume
- A significant number of clients take two (or more) FlashCopies per day for database backup both of which can now be incremental
- The Global Mirror journal FlashCopy also counts as an incremental FlashCopy so the testing copy can now also be incremental
- The functionality is also available as an RPQ from R7.1.5



MultiTarget Metro Mirror Performance

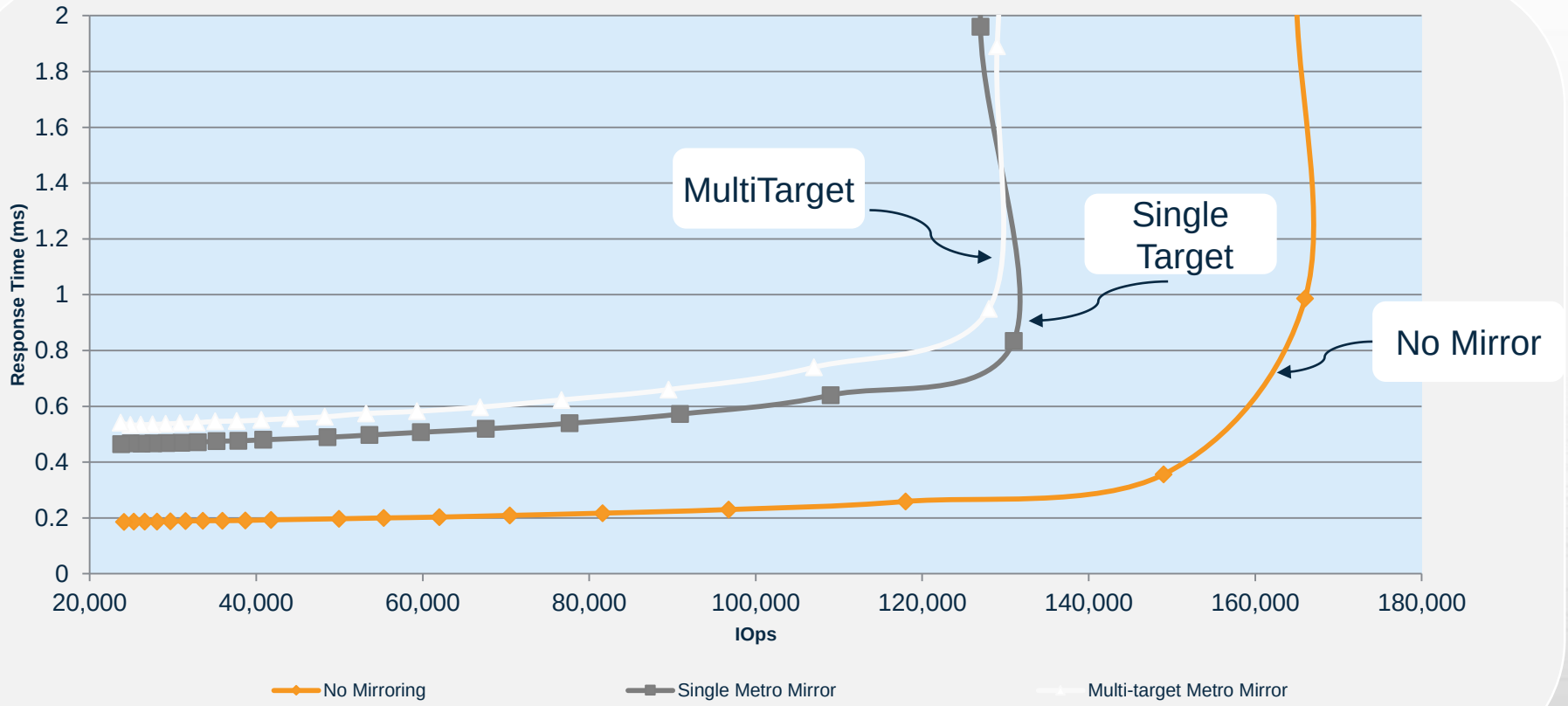
Multi-Target Metro Mirror



- Allow a single volumes to be the source for more than one PPRC relationship
- Provide incremental resynchronization functionality between target devices
- Use cases include
  - Synchronous replication within a datacentre combined with another metro distance synchronous relationship
  - Add another synchronous replication for migration without interrupting existing replication
  - Allow multi-target Metro Global Mirror as well as cascading for greater flexibility and simplified operational scenarios
  - Combine with cascading relationships for 4-site topologies and migration scenarios

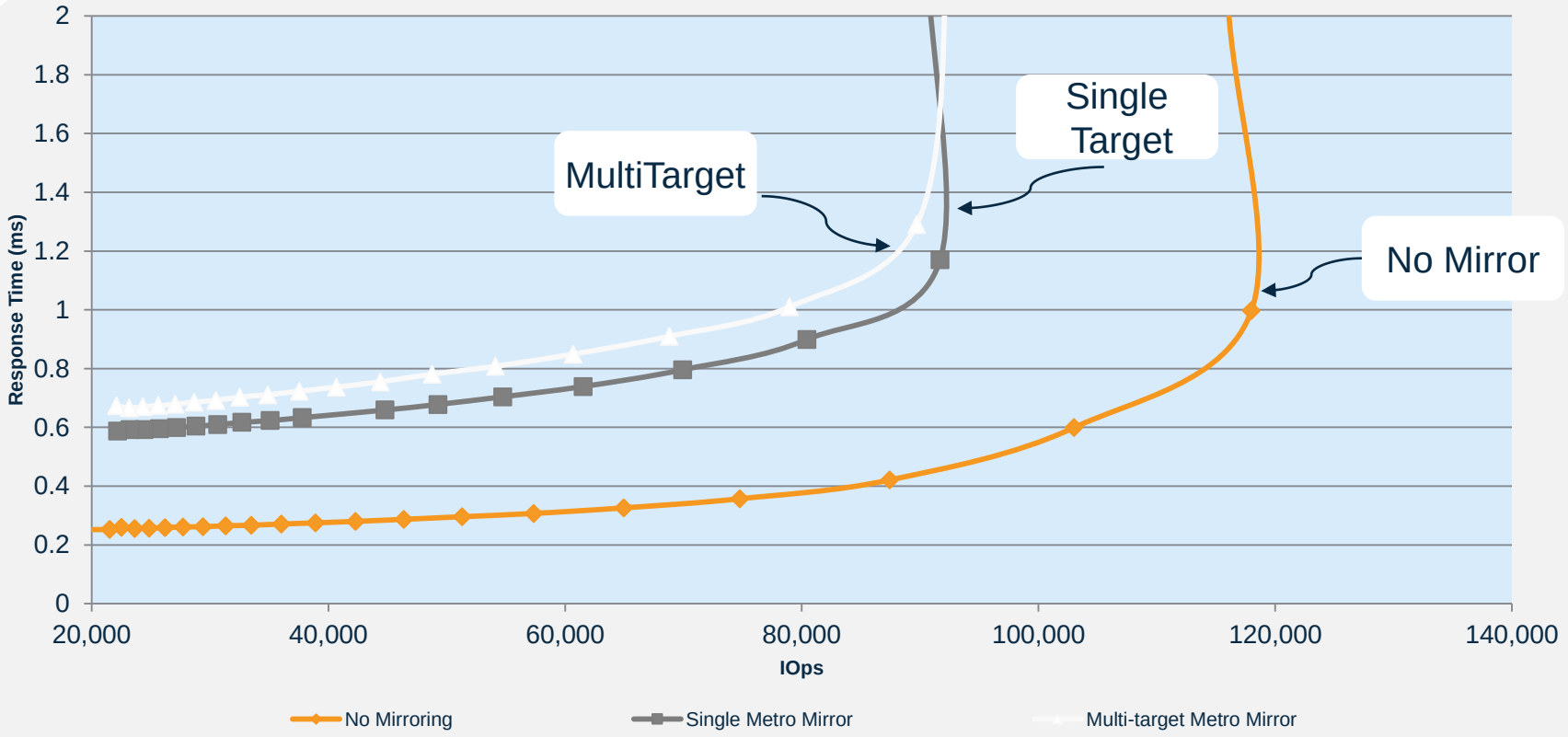
MultiTarget Metro Mirror Performance

4KB Writes



MultiTarget Metro Mirror Performance

27KB Writes

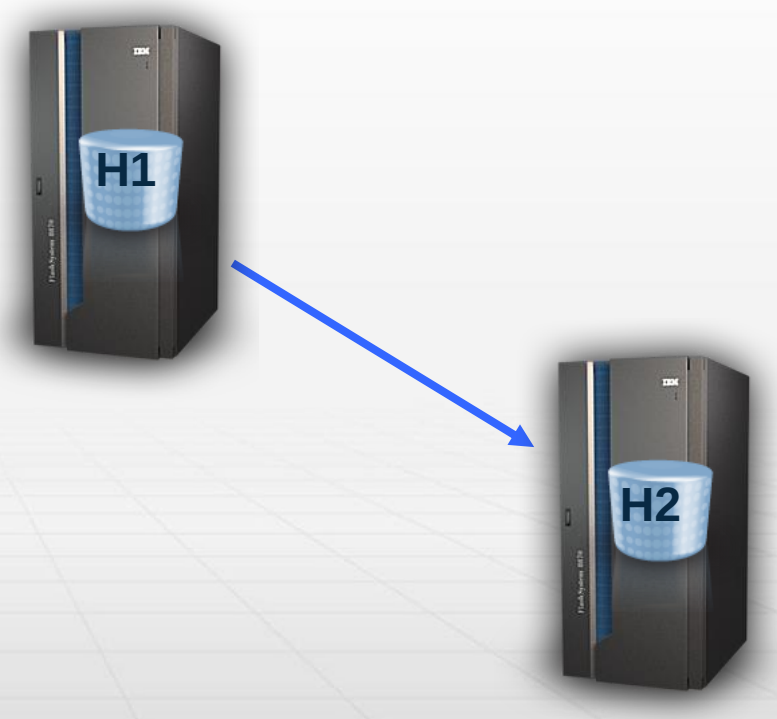


PPRC Synchronization

PPRC Synchronization

- The asynchronous copying of data from a PPRC primary to a secondary.

- Copies data that is out-of-sync between primary and secondary
  - Initial copy when a pair is established or resumed
  - Global Copy / Global Mirror to asynchronously transfer updated data

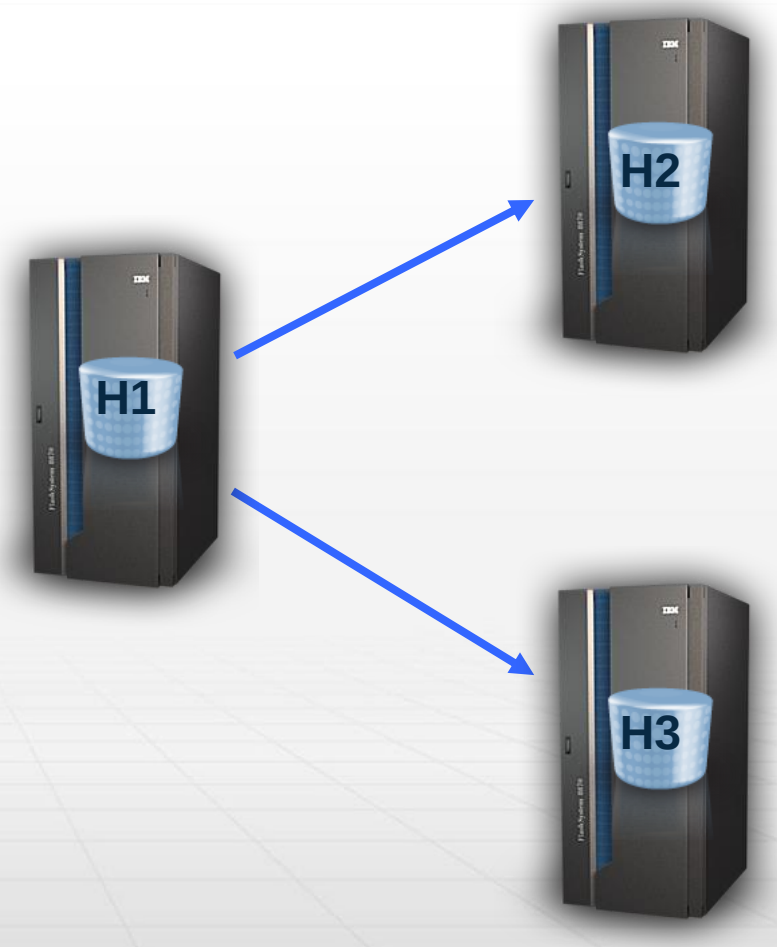


Pre-7.4 Design

- Volume based
  - When a volume spans ranks, only the part on one rank copied at a time
- Did not scale with volume size
  - Resources allocated per volume, regardless of size
- No priority mechanism
- Unable to handle multiple relationships on a volume for MultiTarget PPRC

Objectives

- Support MultiTarget PPRC
- Finish the copy as quickly as possible
  - Fully utilize the PPRC links
- Minimize the impact on other work
  - Do not overdrive the ranks on the primary
  - Minimize impact on host I/O
- Do the most important work first
  - Priority scheme



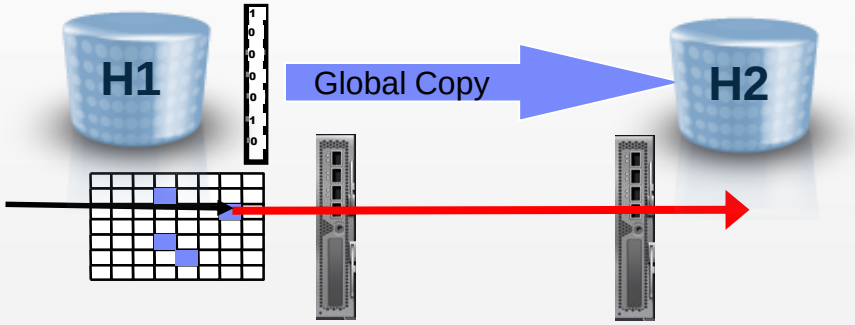
New Design

- Balances workload across:
  - PPRC Ports
  - Extent Pools
  - Device Adapters
  - Ranks
- Assigns priorities
  - For example, forming GM consistency groups > Resynchronization
- Unit of work is an extent
  - Scales with volume size

Global Copy Collision Avoidance

Global Copy Collision

- Collision definition:
  - Track is locked for Global Copy to transfer it to the secondary
  - Host write occurs for same track.
- Result:
  - Host write must wait for Global Copy transfer to complete
  - Impact to application

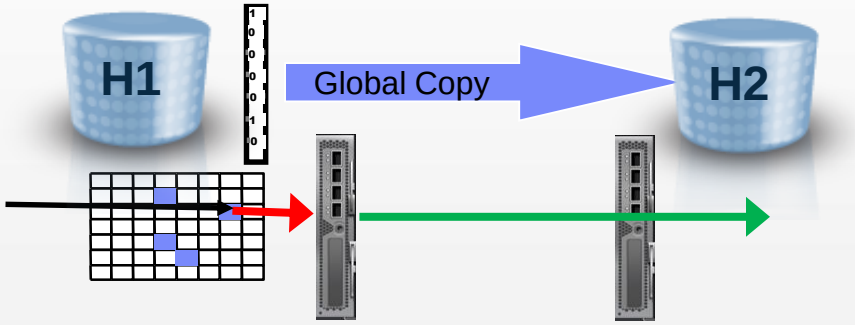


Track in the process of being sent is locked to prevent writes from occurring

- Not usually a problem except for situations with
  - Have unstable networks
  - Have high latency / long distance networks
  - Have workloads with a high rate of data re-reference (e.g. logging)
  - Have very latency sensitive applications

Global Copy Collision Avoidance

- Global Copy releases track lock after transfer of data to local host adapter
- Allows Host Write to access track immediately without waiting for Global Copy transfer to complete
- Global Copy detects when track has been modified by another host write
- Available with R7.4 and as RPQ on R7.2 and R6.3

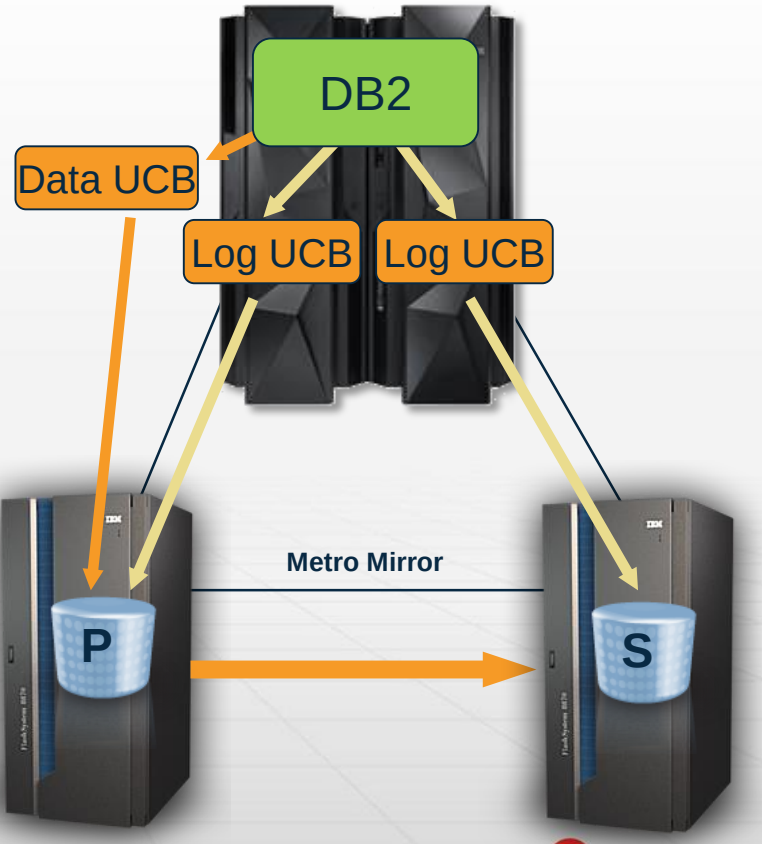


IBM zHyperWrite

zHyperWrite

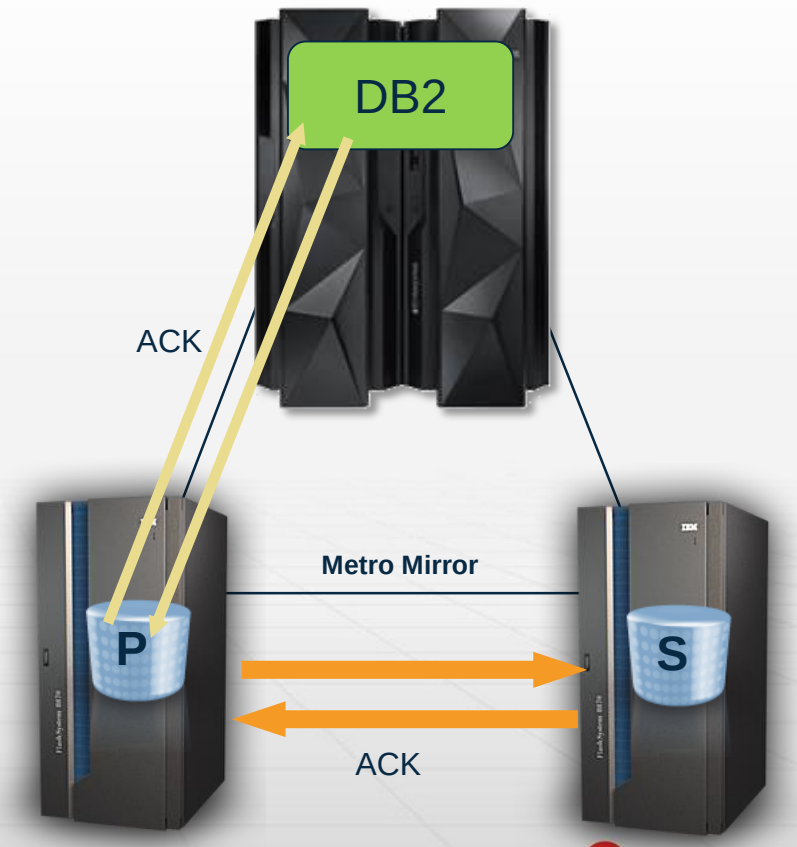
- Improved DB2 Log Write Performance with DS8870 Metro Mirror
  - Reduces latency overhead compared to normal storage based synchronous mirroring

- Reduced write latency and improved log throughput



DB2 Log Write with Metro Mirror

1. DB2 Log Write to Metro Mirror Primary
2. Write Mirrored to Secondary
3. Write Acknowledged to Primary
4. Write Acknowledged to DB2

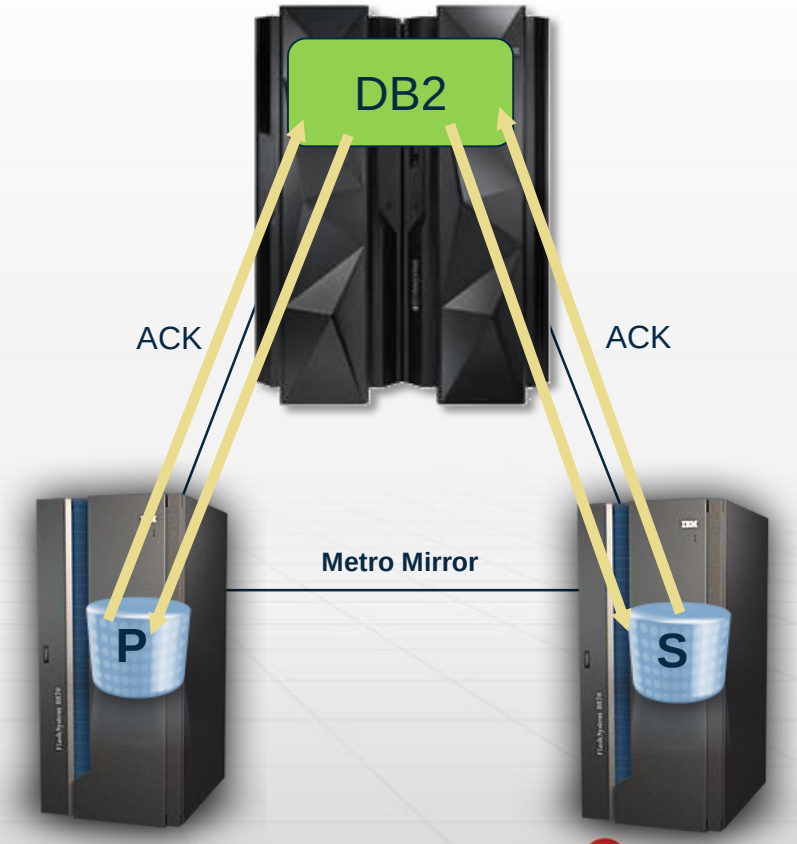


Write with zHyperWrite

1. DB2 Log Write to Metro Mirror Primary and Secondary in parallel

2. Writes Acknowledged to DB2

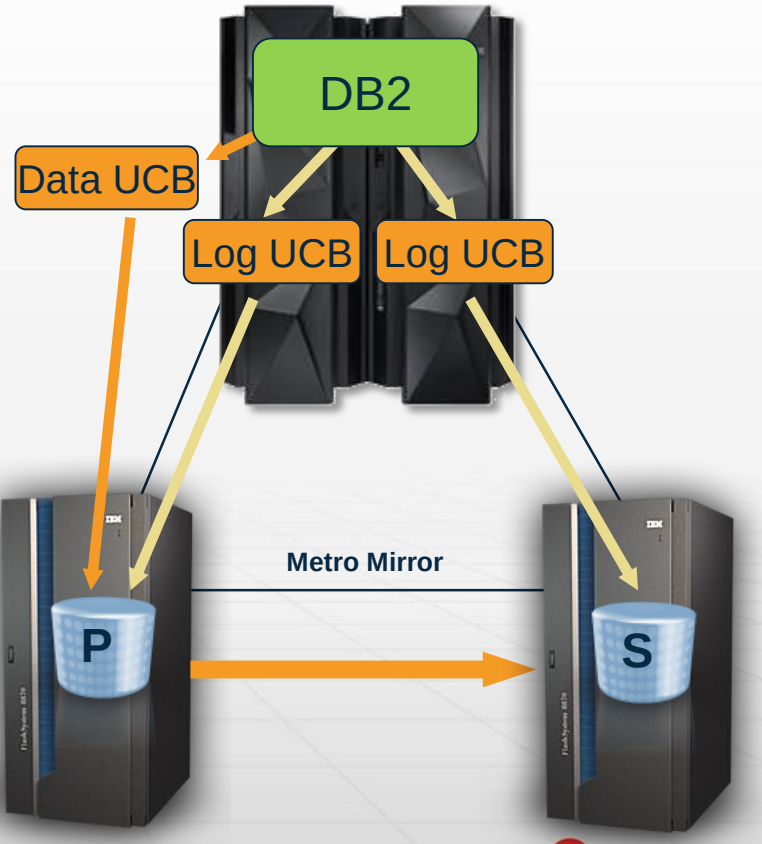
3. Metro Mirror does not mirror the data.



IBM zHyperWrite

- Supports HyperSwap with TPC-R or GDPS

- Enabled through
  - SYS1.PARMLIB(IECIOSxx)
  - SETIOS command
  - DS8870 R7.4, IOS, DFSMS PTF's

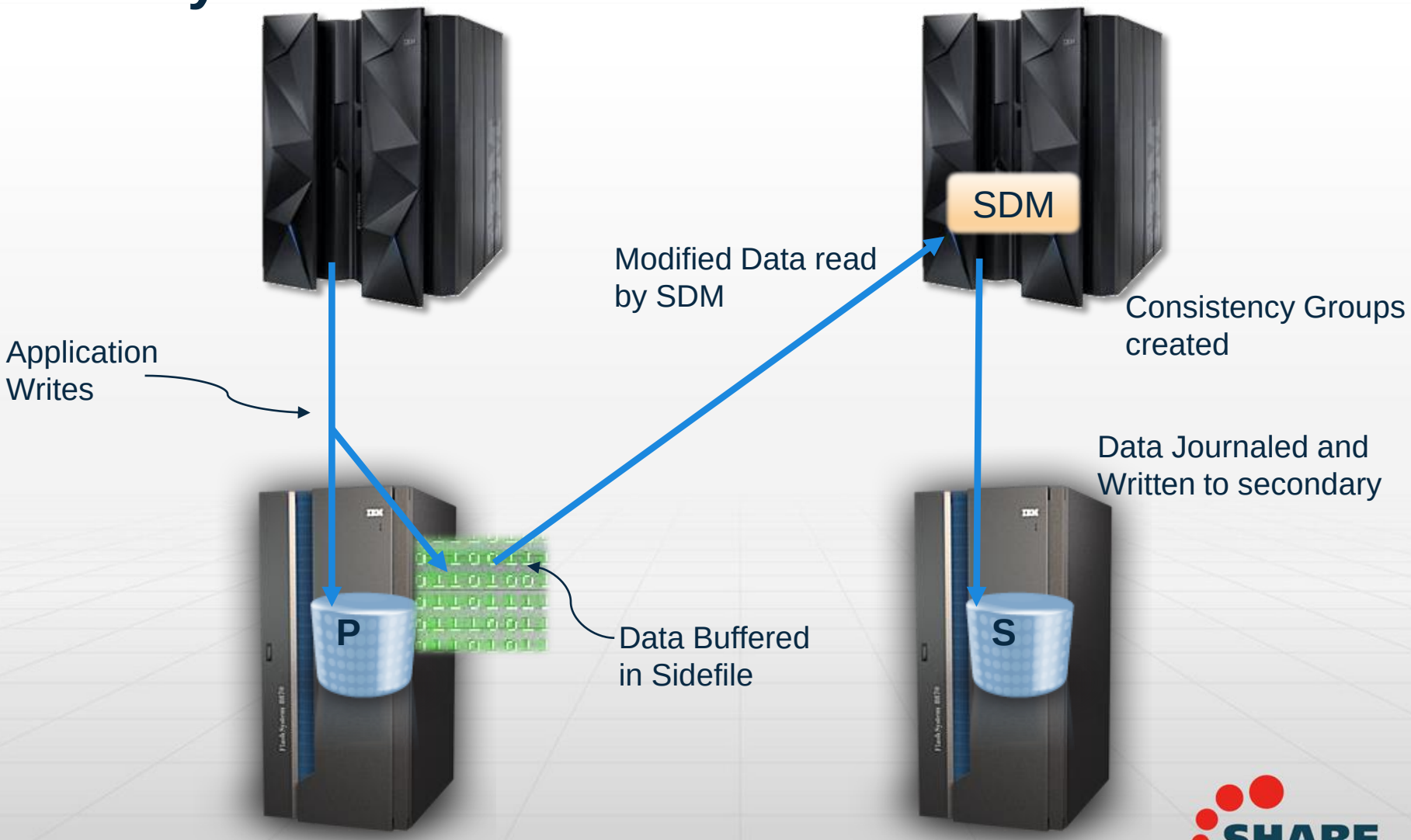


z/OS (XRC) Global Mirror Workload Based Write Pacing

z/GM (XRC) Workload Based Write Pacing

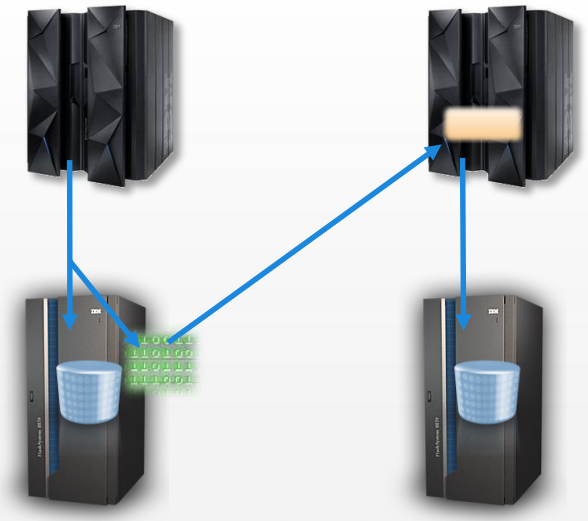
- Need for Write Pacing
- Current Write Pacing
- Limitations of Current Write Pacing
- Requirements
- Use of Workload Manager (WLM)
- Example
- Implementation Requirements

z/GM System



Need for Write Pacing

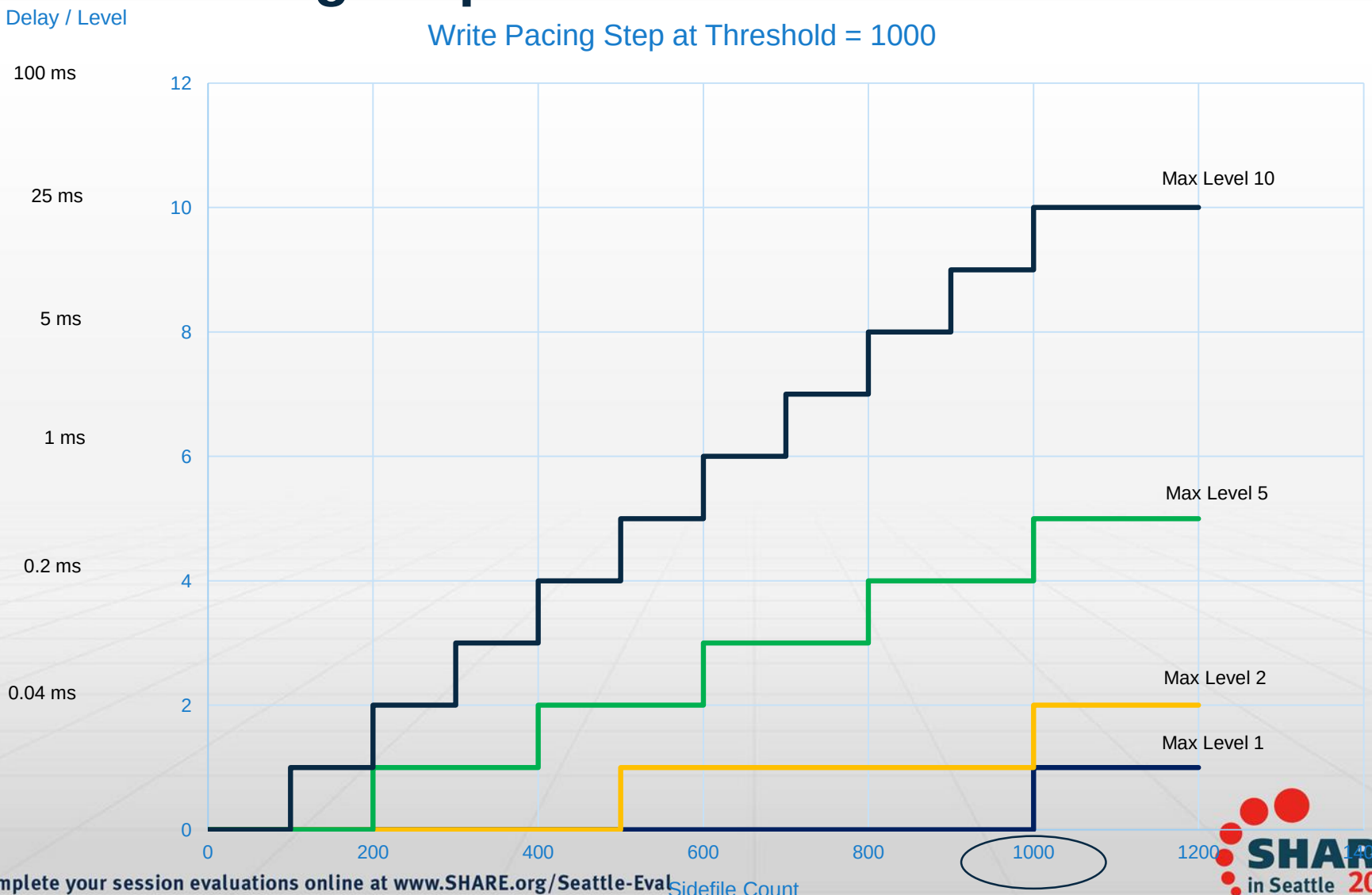
- Write data is buffered in the DS8000 sidefiles
  - Maximum sidefile size is finite
- Burst write rates can exceed capacity to offload data
  - Sidefiles grow
  - RPO increases
  - Possible suspension if persists
- Write Pacing monitors sidefile size and injects delays to flatten out peaks of the write rate



Previous XRC Write Pacing

- Volume based
  - Sidefile count monitored for each volume
- Thresholds and Maximum Delay are specified for each volume
  - Different volumes may have different values
- If the sidefile count for a volume grows:
  - Delays injected for writes to that volume
  - Delay starts very small
    - Delay increased if sidefile count increases, up to maximum allowed
    - Delay reduced if sidefile count decreases

Write Pacing Step Function



Limitations to Previous Write Pacing

- Different applications have different response time requirements
- These requirements are currently met by:
  - Assigning different pacing threshold and limits to different volumes
  - Placing data on volumes with the appropriate pacing levels
- Requires significant planning for data placement
- If requirements change, data must be moved to different volume

Write Pacing Requirements

- Meet application response time and performance objectives
- Maintain disaster recovery capability within desired Recovery Point Objective (RPO)
- Minimize the amount of manual planning and intervention
- Automatically adapt to changing application needs

Workload Manager

- z/OS Workload Manager (WLM) provides ability to set performance goals
- Applications with similar goals are grouped into Service Classes
- WLM assigns resources to maximize goal achievement
- One part of the resource management is that I/O has an ***importance*** value
  - Six importance values:
    - 1 = Highest
    - 5 = Lowest
    - 6 = Discretionary (or default, when not part of a service class)

Workload Based z/GM Write Pacing

- Takes into account the I/O's importance value from WLM when determining the amount of pacing
- Each importance level is mapped to a Maximum Pacing Level
- Pacing levels are set so that higher importance I/O is paced less then lower importance I/O

Example with WL Based Pacing

- Given:
  - Threshold level = 1000
  - Sidefile count = 500
  - Volume Pacing level = 8

Importance Level	Pacing Level	Workload Pacing Delay	Volume Pacing Delay
1 (high)	4	0.04ms	0.2ms
3 (med)	8	0.2ms	0.2ms
5 (low)	12	1.0ms	0.2ms

- Delay varies based on I/O's importance

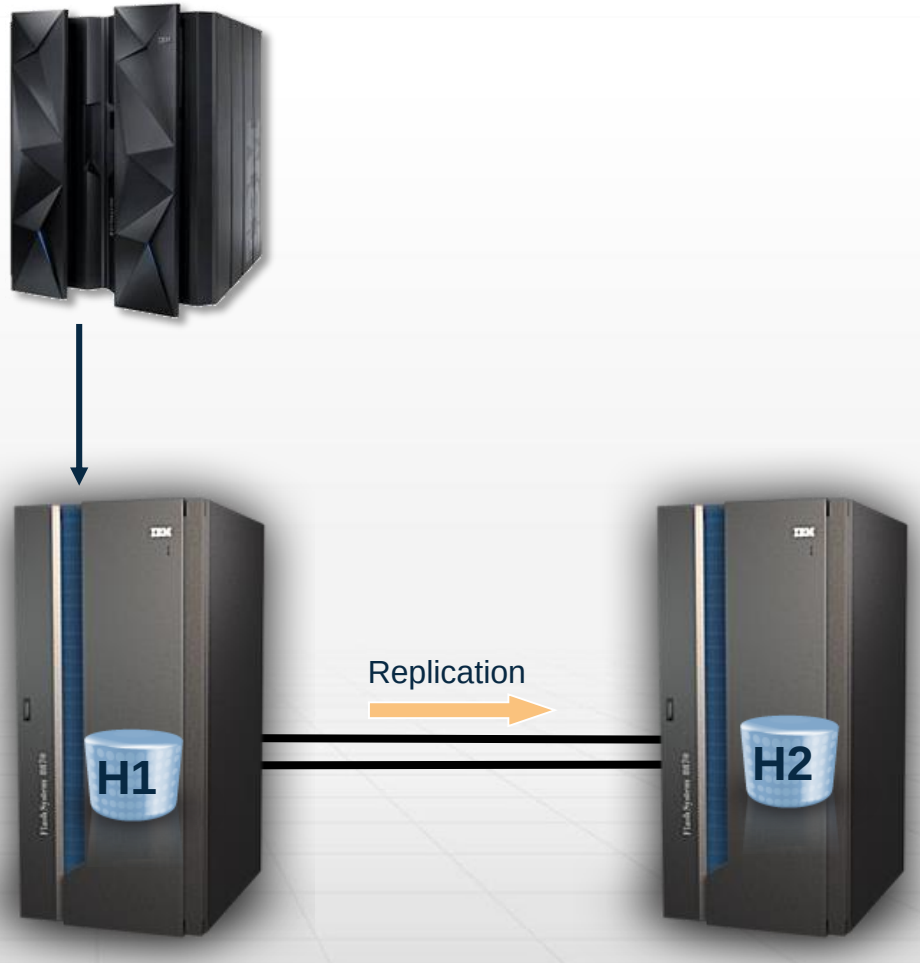
Implementation Requirements

- Configure WLM
- Define Workload Classes
- Enable IO Priority Management
- Determine maximum delay for each workload class
- Specify these values in the XRC PARMLIB

Easy Tier Heat Map Transfer

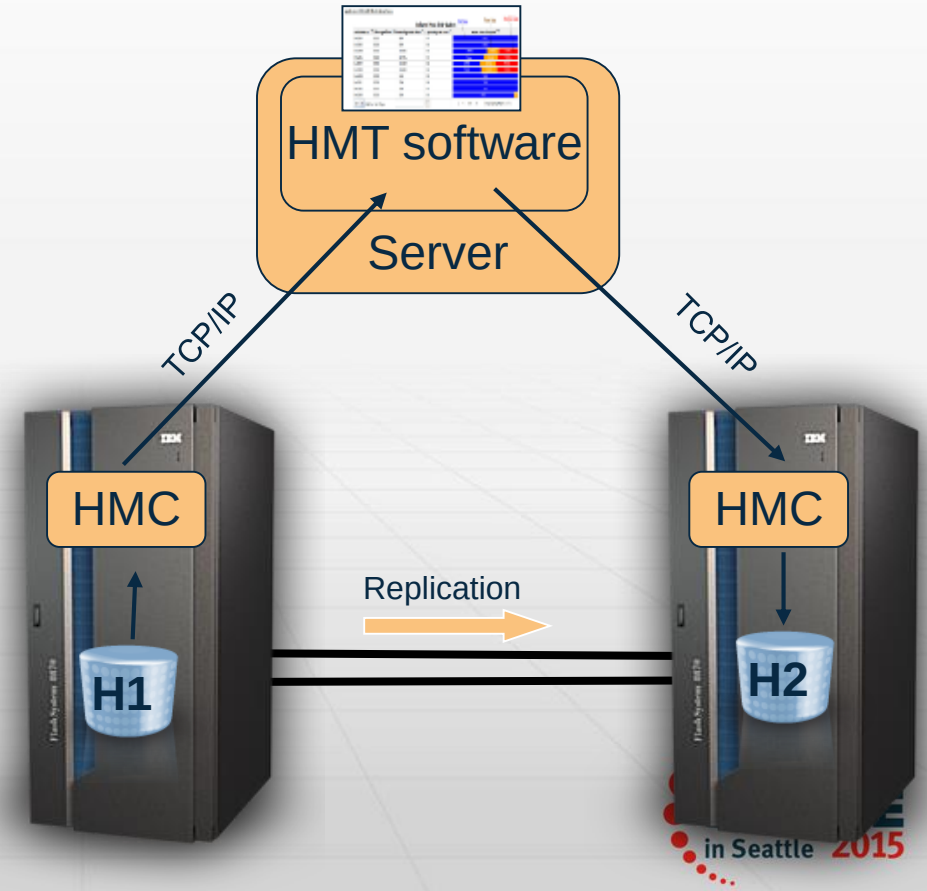
Easy Tier Heat Map – With PPRC

- Heat Map maintained at both the primary and the secondary
- But… I/O at the secondary is different from that at the primary

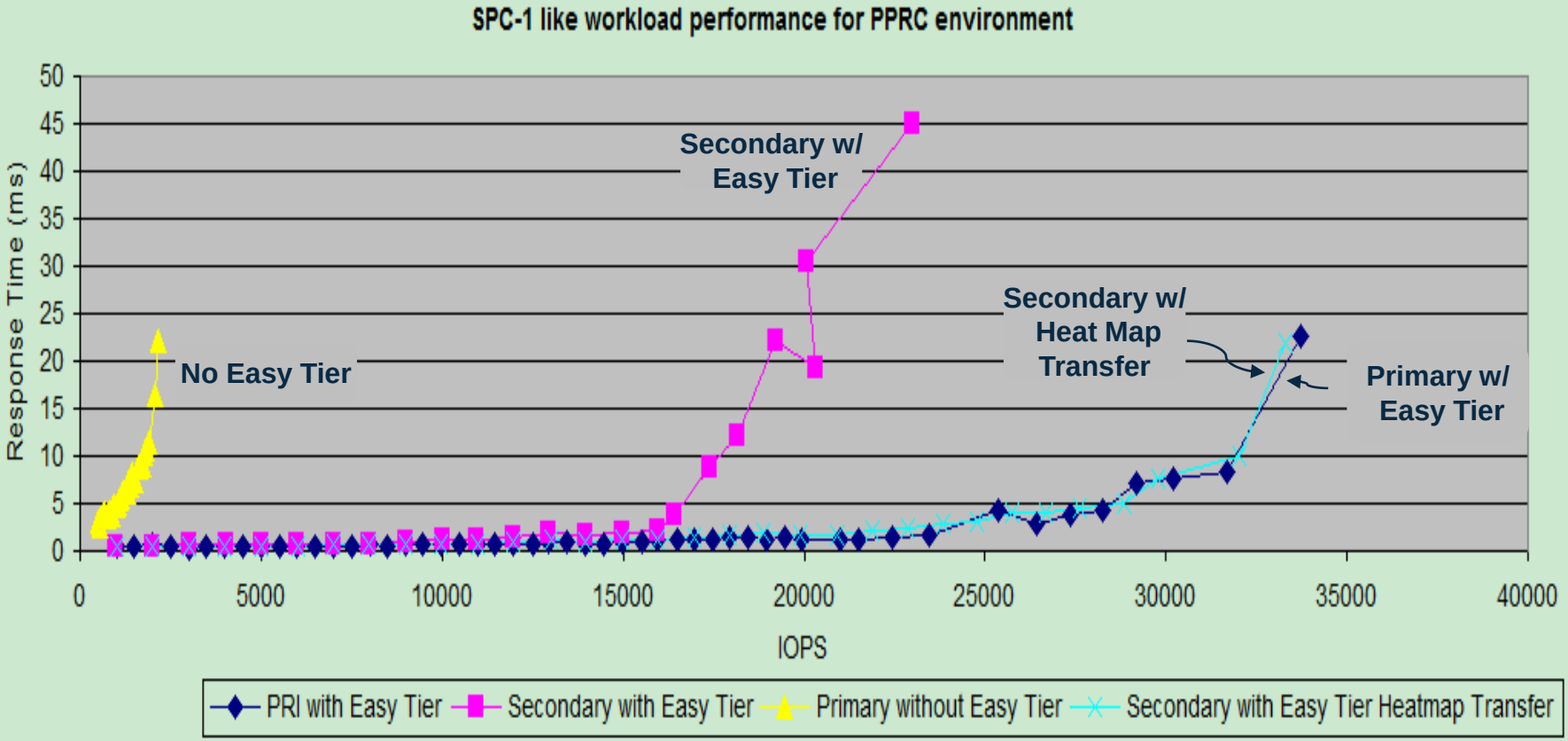


Easy Tier Heat Map Transfer

- Transfers Easy Tier Heat Map information for a volume
- Out of band software implementation
- TPC-R and GDPS support as well as standalone utility

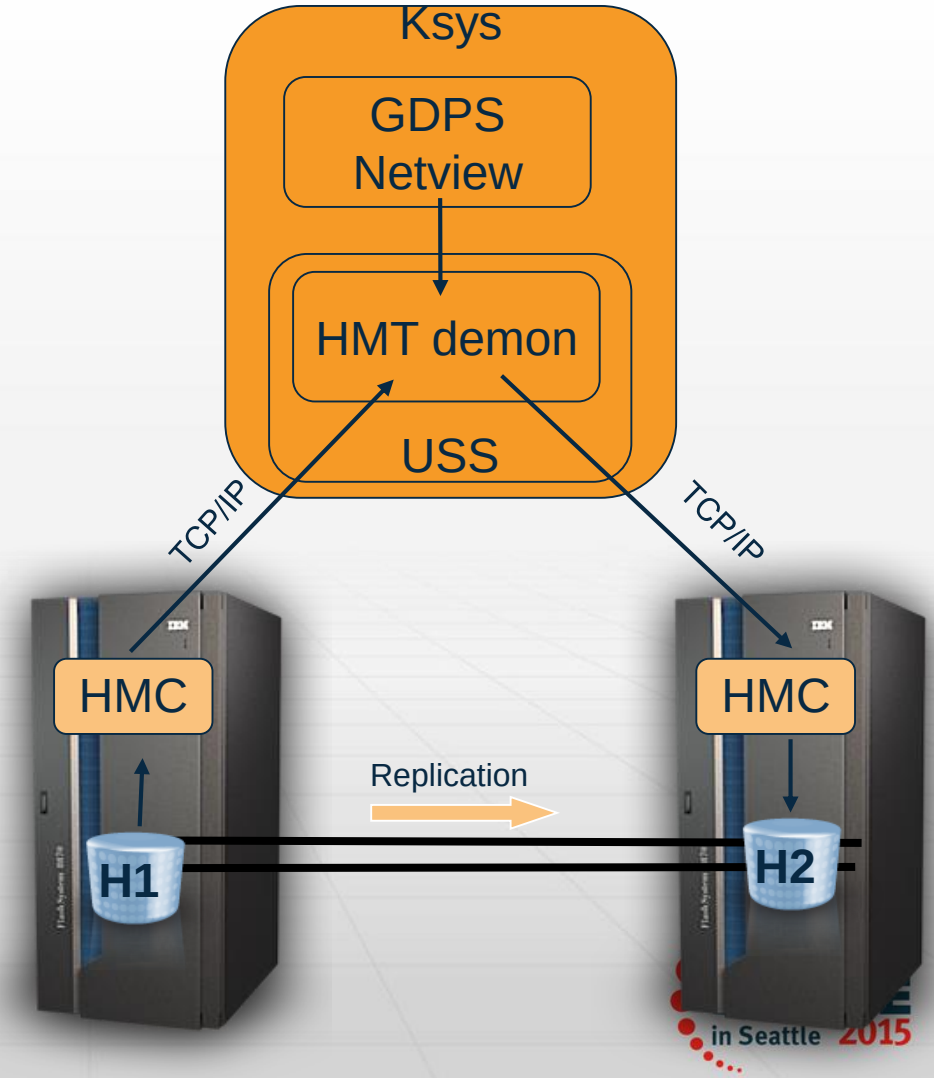


Heat Map Transfer Measurement



Easy Tier Heat Map Transfer

- GDPS/PPRC support available in an SPE with GDPS 3.10 and GDPS/GM support available with GDPS 3.11
- GDPS/XRC support is planned to be released next
- 3 and 4 site support planned by combining the different functions



Session Summary

- Replication Overview
- Multiple Incremental FlashCopy
- MultiTarget PPRC Performance
- PPRC Synchronization
- Global Copy Collision Enhancement
- zHyperWrite
- Workload Based z/OS Global Mirror Write Pacing
- Easy Tier Heat Map Transfer