

Sysplex 101: Setting up a Sysplex

Session 17914 12 August 2015

*Mark A Brooks
mabrook@us.ibm.com
IBM Poughkeepsie*



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

Copyright (c) 2015 by SHARE Inc.  Except where otherwise noted, this work is licensed under <http://creativecommons.org/licenses/by-nc-sa/3.0/>



Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

IBM®	MQSeries®	S/390®	z9®
ibm.com®	MVS™	Service Request Manager®	z10™
CICS®	OS/390®	Sysplex Timer®	z/Architecture®
CICSplex®	Parallel Sysplex®	System z®	zEnterprise™
DB2®	Processor Resource/Systems Manager™	System z9®	z/OS®
eServer™	PR/SM™	System z10®	z/VM®
ESCON®	RACF®	System/390®	z/VSE®
FICON®	Redbooks®	Tivoli®	zSeries®
HyperSwap®	Resource Measurement Facility™	VTAM®	
IMS™	RETAIN®	WebSphere®	
IMS/ESA®	GDP\$®		
	Geographically Dispersed Parallel Sysplex™		

The following are trademarks or registered trademarks of other companies.

IBM, z/OS, Predictive Failure Analysis, DB2, Parallel Sysplex, Tivoli, RACF, System z, WebSphere, Language Environment, zSeries, CICS, System x, AIX, BladeCenter and PartnerWorld are registered trademarks of IBM Corporation in the United States, other countries, or both.
 DFSMSHsm, z9, DFSMSrmm, DFSMSdtp, DFSMSdss, DFSMS, DFS, DFSORT, IMS, and RMF are trademarks of IBM Corporation in the United States, other countries, or both.
 Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
 Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
 InfiniBand is a trademark and service mark of the InfiniBand Trade Association.
 UNIX is a registered trademark of The Open Group in the United States and other countries.
 Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the 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.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Abstract

Your management has heard all these great stories about how a sysplex will provide the high availability characteristics that your business requires. Congratulations! You've been chosen to build them one of those. So you grab a copy of *z/OS MVS Setting Up A Sysplex* and start reading. But with over 500 pages to digest, it would sure be nice to get some guidance as to how to approach the task. In this session, we'll explain what a sysplex is and what needs to happen to deliver those highly available business services that your boss expects. We'll look at the various components of a sysplex and the things that must be done to build a multisystem sysplex. We'll keep it at a high level since there is so much to cover. But by the end, you'll understand all the sysplex terminology and you'll have the understanding you need to navigate *Setting Up A Sysplex* so you can successfully create the sysplex that will meet the needs of your business.

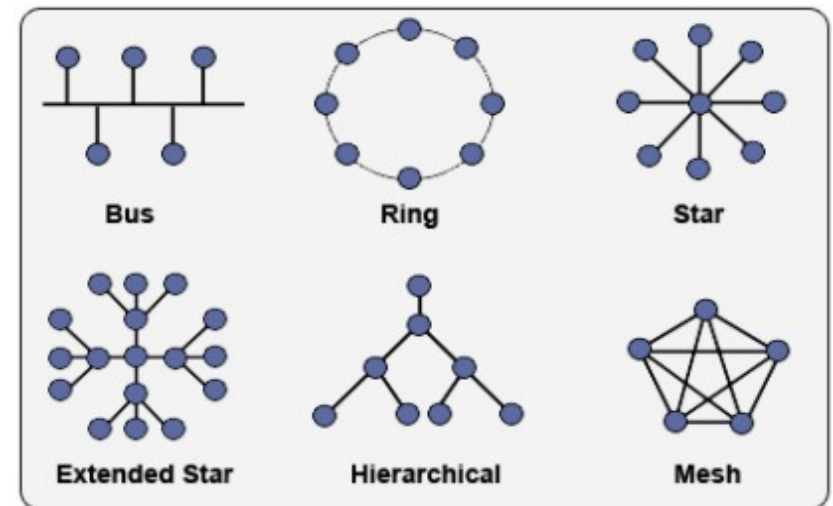
Please note:

There are far more slides in this handout than I will present during the session. In an hour, I only have time to talk about the highlights. I include the additional slides as reference material in order to:

- Provide details that I may fail to verbalize during the session
- Leave you with a narrative that will (hopefully) stand on its own when you review the material in a few weeks and have long forgotten my verbal explanation
- Despite the extra slides, there is even more that could be said. It is my hope that this presentation will give you the foundation for further exploration of those necessary details.

Sysplex is a cluster of z/OS images

- Foundation for high availability, resiliency, horizontal scaling
 - Incremental growth with near linear scalability
 - Near continuous availability
 - Automatic, dynamic workload management
 - High performance access to shared data with integrity
 - Flexibility
 - Technology, price points, performance, capacity
- Issues common to all clusters
 - Node management
 - Communication
 - Data access and data integrity
 - Workload distribution
 - Failures



*wide variety of possible cluster topologies
(sysplex is a mesh, any to any connectivity)*

Why sysplex ?

- Really must start with your business objectives
 - Availability?
 - Growth?
 - Consolidation?
 - Operations?
 - Workload balancing?
 - Cost?
- The business must:
 - Determine level of acceptable risk
 - Make appropriate investments in infrastructure, people, and processes
- The technologist must understand those objectives and provide:
 - An appropriate architecture
 - Well implemented technology
 - Excellent systems management and operations
- Both must work together to achieve the business objectives
 - There are no guarantees; iterative improvement over time
 - Understand and agree to discrepancies between goals and reality
 - Appropriate risk mitigation

Single system vs Multisystem Sysplex

■ XCF-Local Mode

- No couple data sets

COUPLE=**

COUPLE=COUPLE00 (as supplied by IBM)

■ Monoplex

- Has couple data sets
- Determined by PLEXCFG=MONOPLEX

- First system into sysplex updates the sysplex couple data set to indicate that no other system is permitted to join the sysplex.
- Must re-IPL the system to change

■ Multisystem

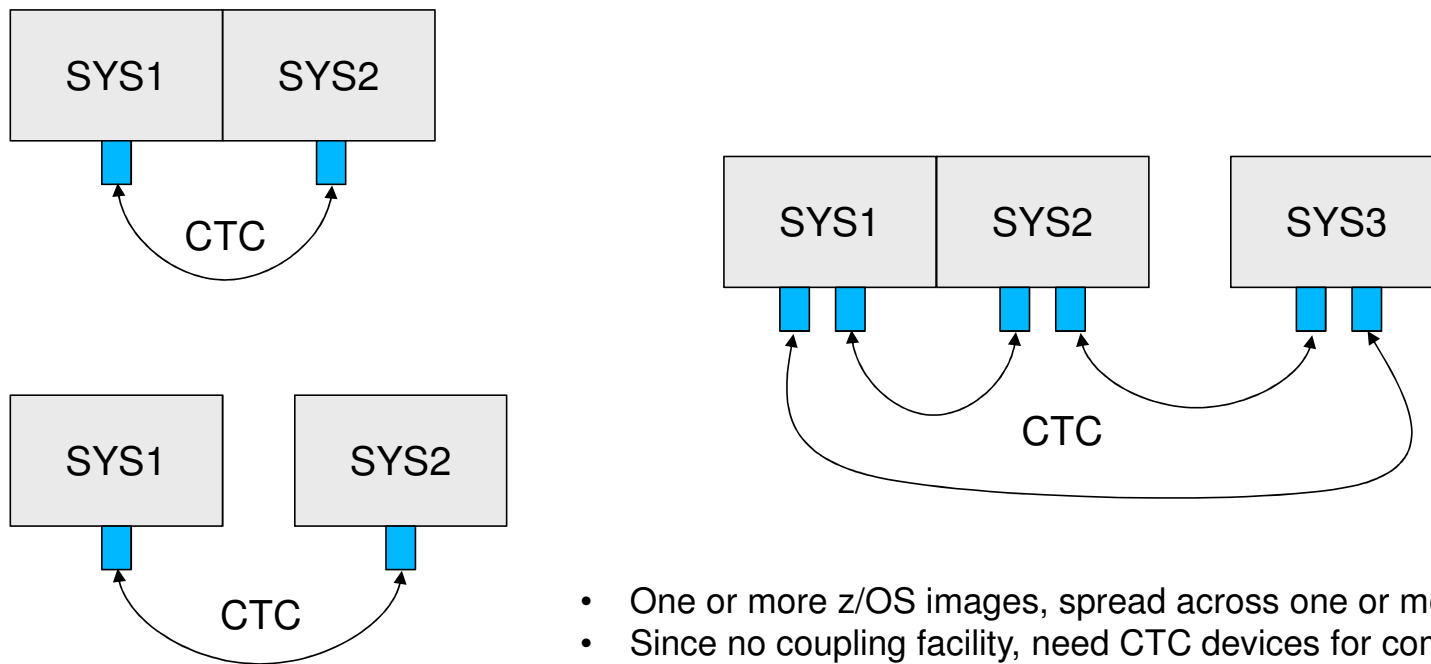
- Has couple data sets
- A system can join an existing sysplex if it can:
 - Use the same sysplex couple data sets as the rest of the sysplex
 - Establish signal connectivity with every system in the sysplex
 - Use the same common time reference as the rest of the sysplex

All must use the same sysplex name

Types of sysplex

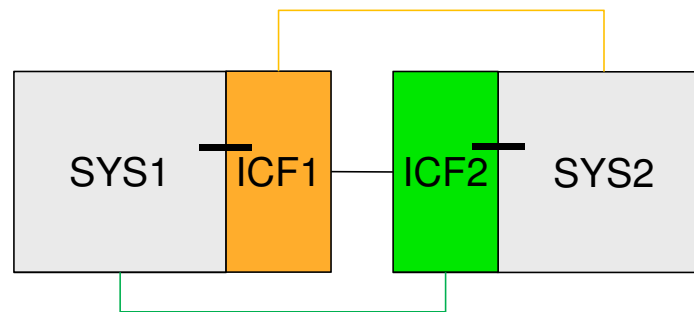
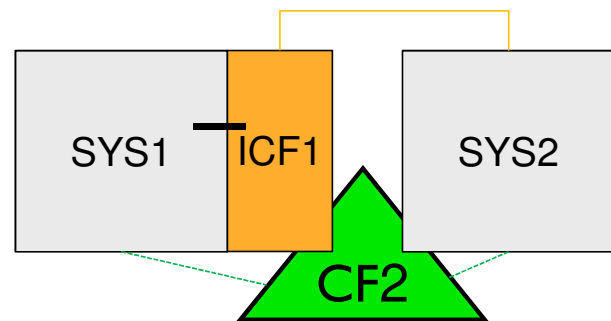
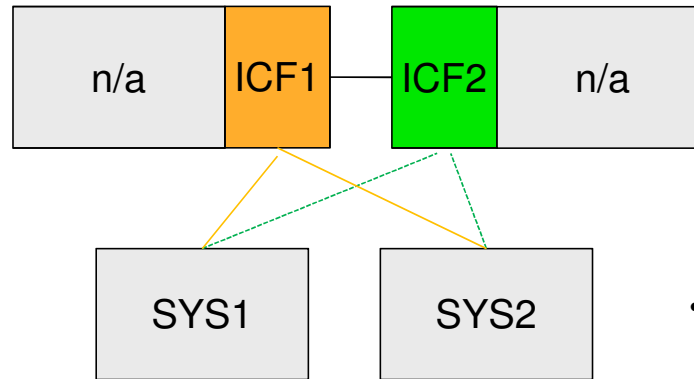
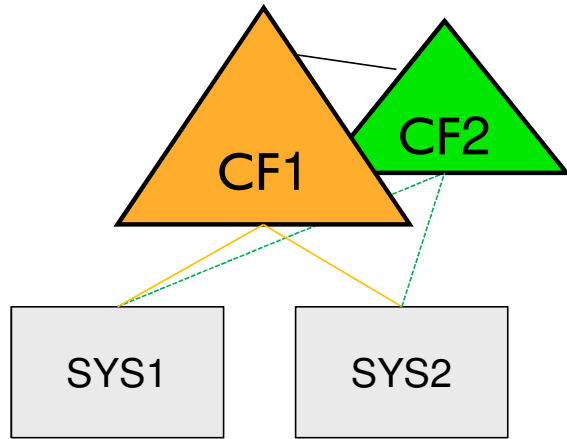
- Base Sysplex
 - Does not have a coupling facility
 - Typically used for resource sharing
- Parallel Sysplex
 - Has a coupling facility
 - Used for resource sharing, data sharing, or both

Variety of configurations: Base Sysplex



- One or more z/OS images, spread across one or more CECs.
- Since no coupling facility, need CTC devices for communication.
- As number of nodes increases, CTC configuration (which is $O(n^2)$) tends to become unmanageable. GRS ENQ performance (ring mode) might also be an issue.

Variety of configurations: Parallel Sysplex



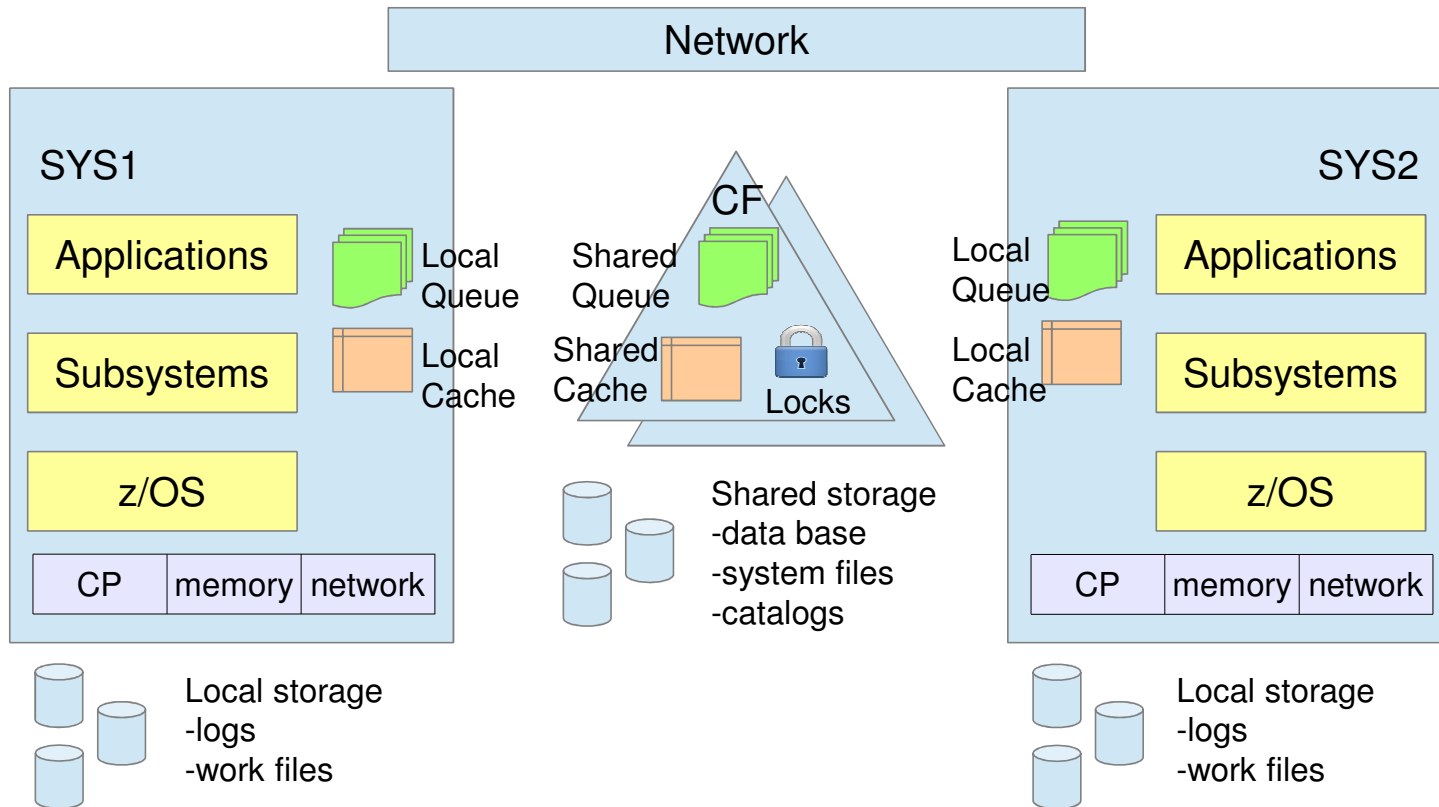
- CF placement relative to z/OS images critical for failure recovery.
- Could have CTC devices for communication as well

- Standalone CF
- Internal CF
- External CF ?

Build it. What can I do with it? vs What do I need? Build it.



Resources used?
Single points of failure?
Affinities?

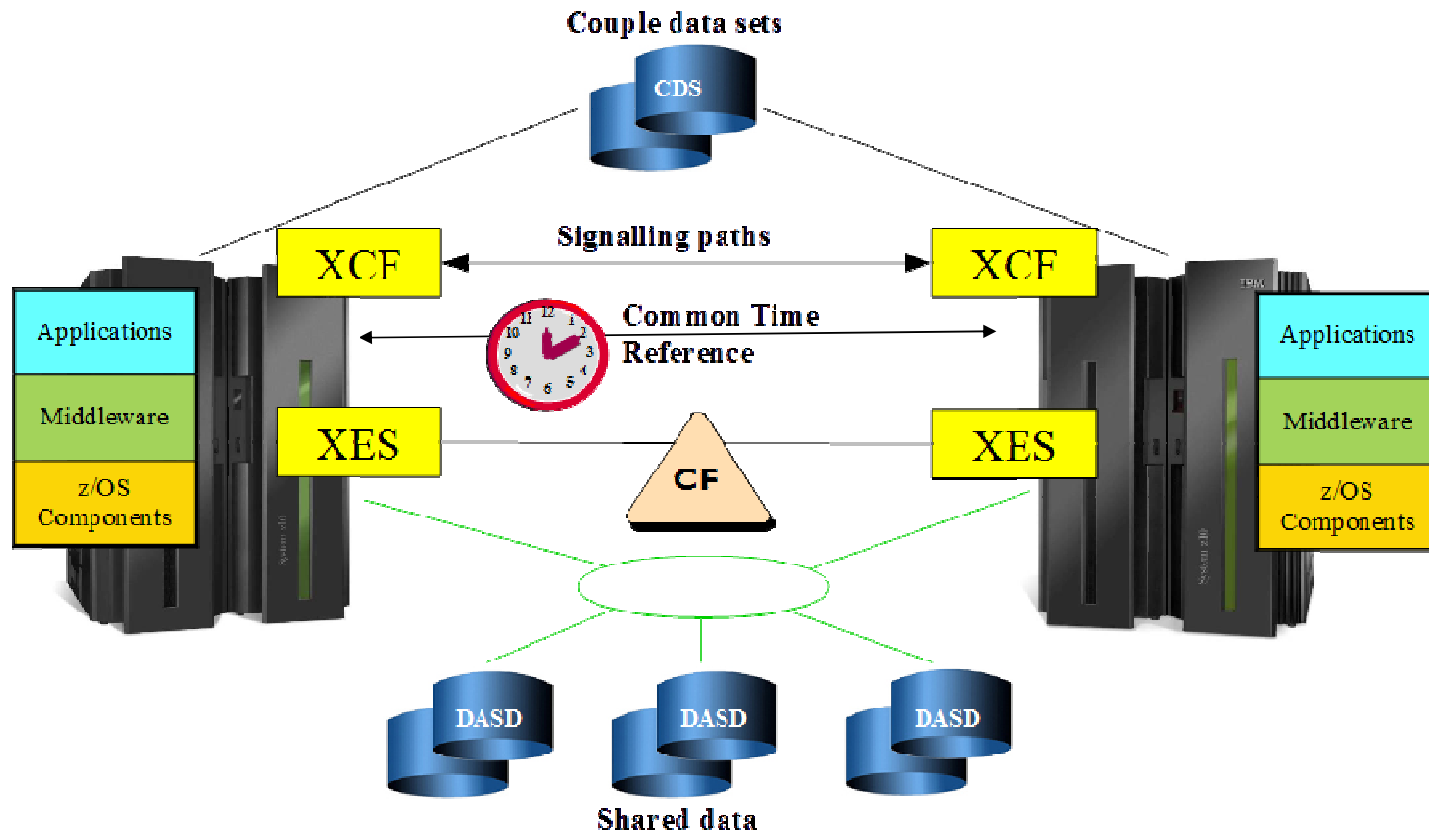


Some starting points

Redbooks

- **z/OS Parallel Sysplex Configuration Overview (SG24-6485)**
 - Published back in 2006
 - So won't have the latest, but much of it still applies
- **System/390 MVS Parallel Sysplex Migration Paths (SG24-2502)**
 - Published back in 1996 when the sysplex world was much simpler
 - All the machine references are obsolete, but the concepts still apply

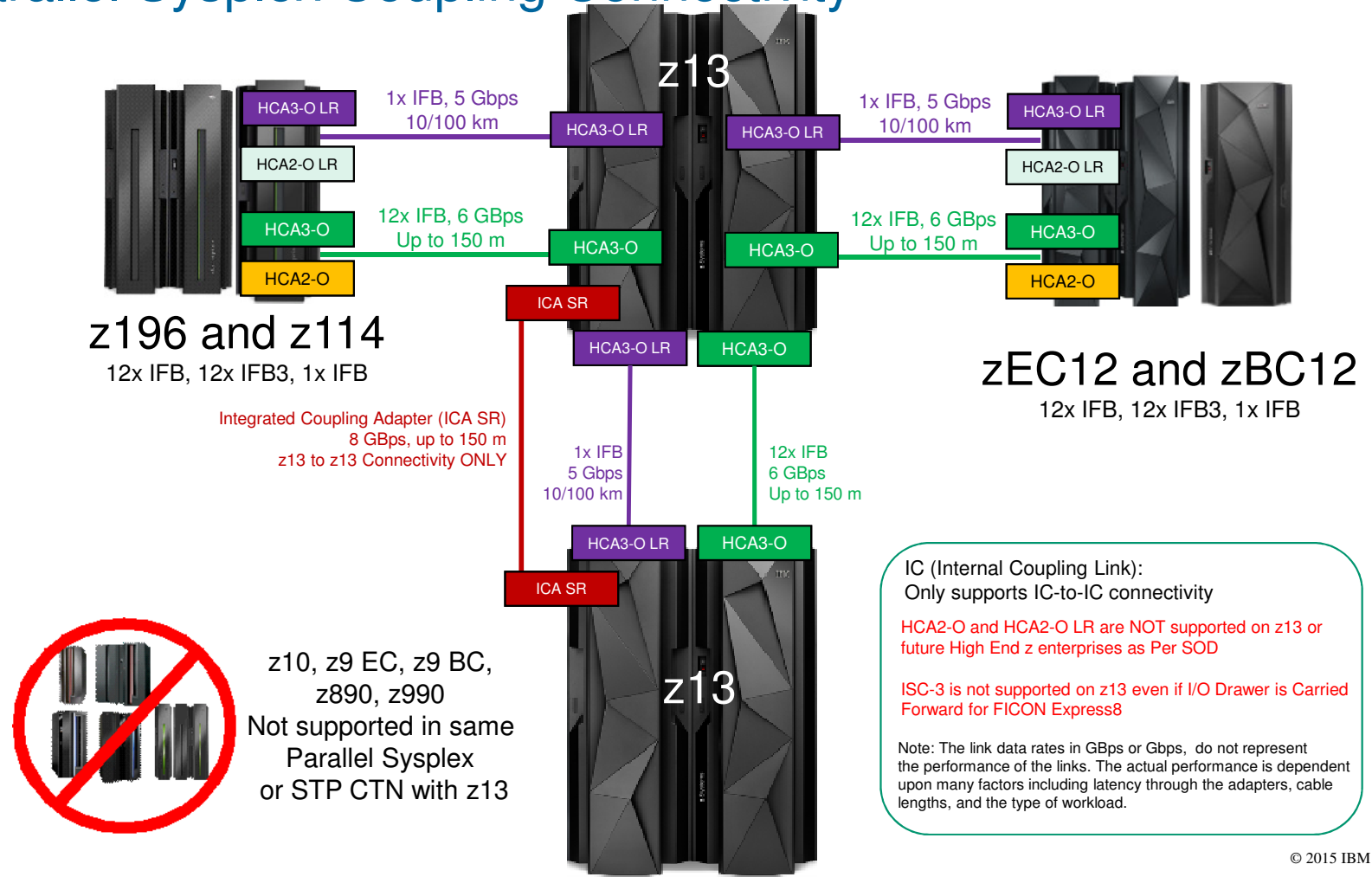
Parallel Sysplex



Building a sysplex

- Hardware Infrastructure
 - One or more z/OS images on one or more CECs
 - Coupling links
- Common Time Reference
- Couple Data Sets (CDS)
 - Sysplex CDS
 - Function CDS
- XCF Signal Paths
- Coupling Facility

z13 Parallel Sysplex Coupling Connectivity

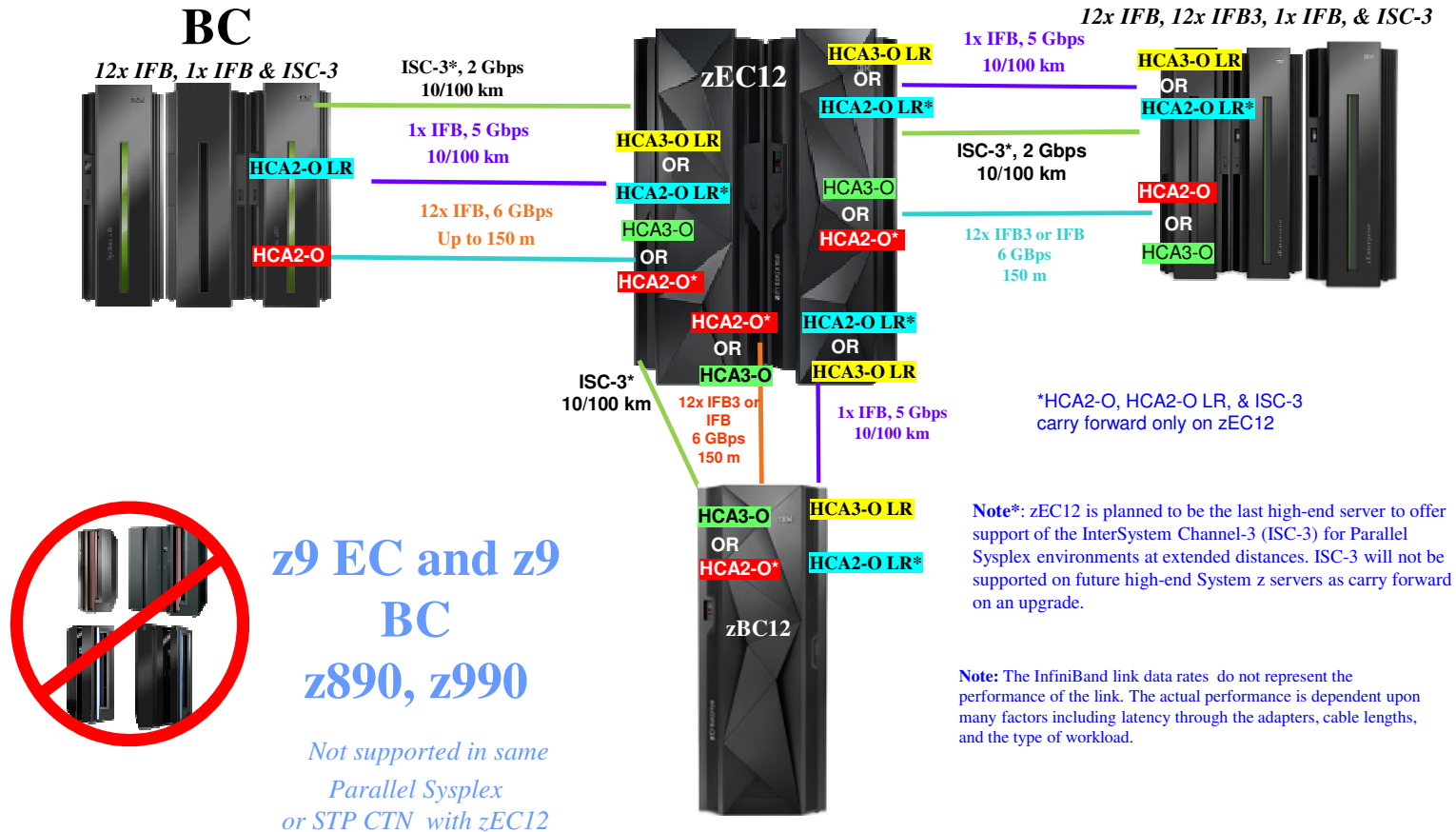


z10, z9 EC, z9 BC,
z890, z990
Not supported in same
Parallel Sysplex
or STP CTN with z13

zEC12/zBC12 Parallel Sysplex Coupling Connectivity

z10 EC and z10

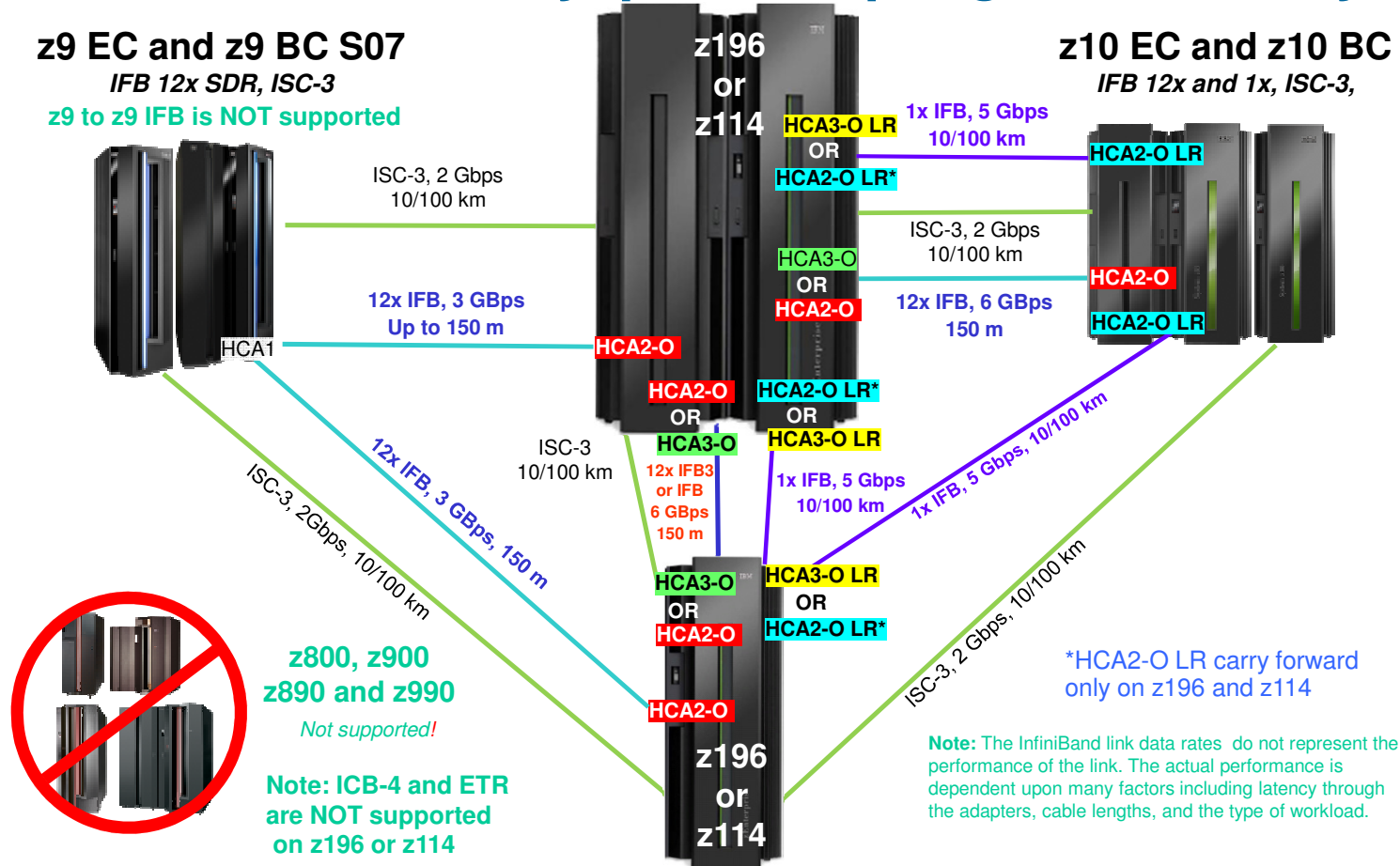
z196 and z114



z9 EC and z9 BC
z890, z990

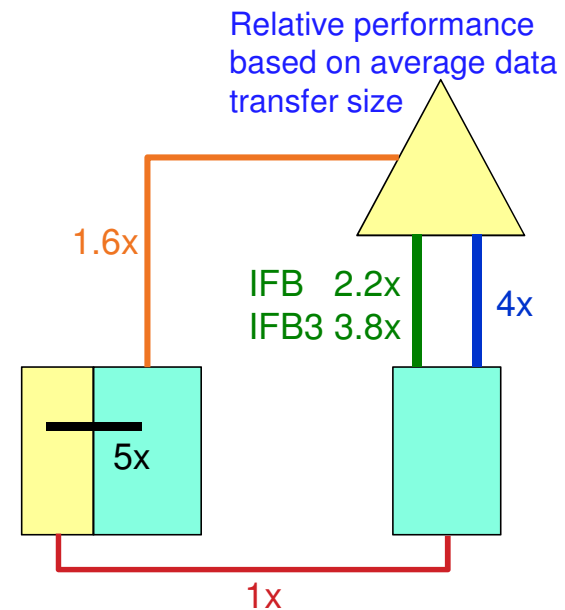
Not supported in same Parallel Sysplex or STP CTN with zEC12

z114 and z196 GA2 Parallel Sysplex Coupling Connectivity



Coupling Link Choices - Overview

- **ISC (Inter-System Channel / HCA2-C / CPC-to-IO Fanout)**
 - Fiber optics
 - I/O Adapter card
 - 10km, 20km support with RPQ 8P2197 as carry forward only, and longer distances with qualified DWDM solutions
- **PSIFB (1x IFB / HCA2-O LR or HCA3-O LR / CPC-to-CPC Fanout)**
 - Fiber optics – uses same cabling as ISC
 - 10km and longer distances with qualified WDM solutions
 - Supports multiple CHPIDs per physical link
 - Multiple CF partitions can share physical link
- **PSIFB (12x IFB / HCA2-O or HCA3-O / CPC-to-CPC Fanout)**
 - 150 meter max distance optical cabling
 - Supports multiple CHPIDs per physical link
 - Multiple CF partitions can share physical link
- **ICA SR (Integrated Coupling Adapter)**
 - Connects to PCIe fanout
 - 150 meter max distance
 - Supports up to 4 CHPIDs per physical link
- **IC (Internal Coupling Channel / Internal CPC)**
 - Microcode, no external connection
 - Only between partitions on same processor



For more information

- “IBM z Systems Connectivity Handbook” (SG24-5444)
 - Updated April 2015

- “Implementing and Managing InfiniBand Coupling Links on System z” (SG24-7539)
 - Available at www.redbooks.ibm.com

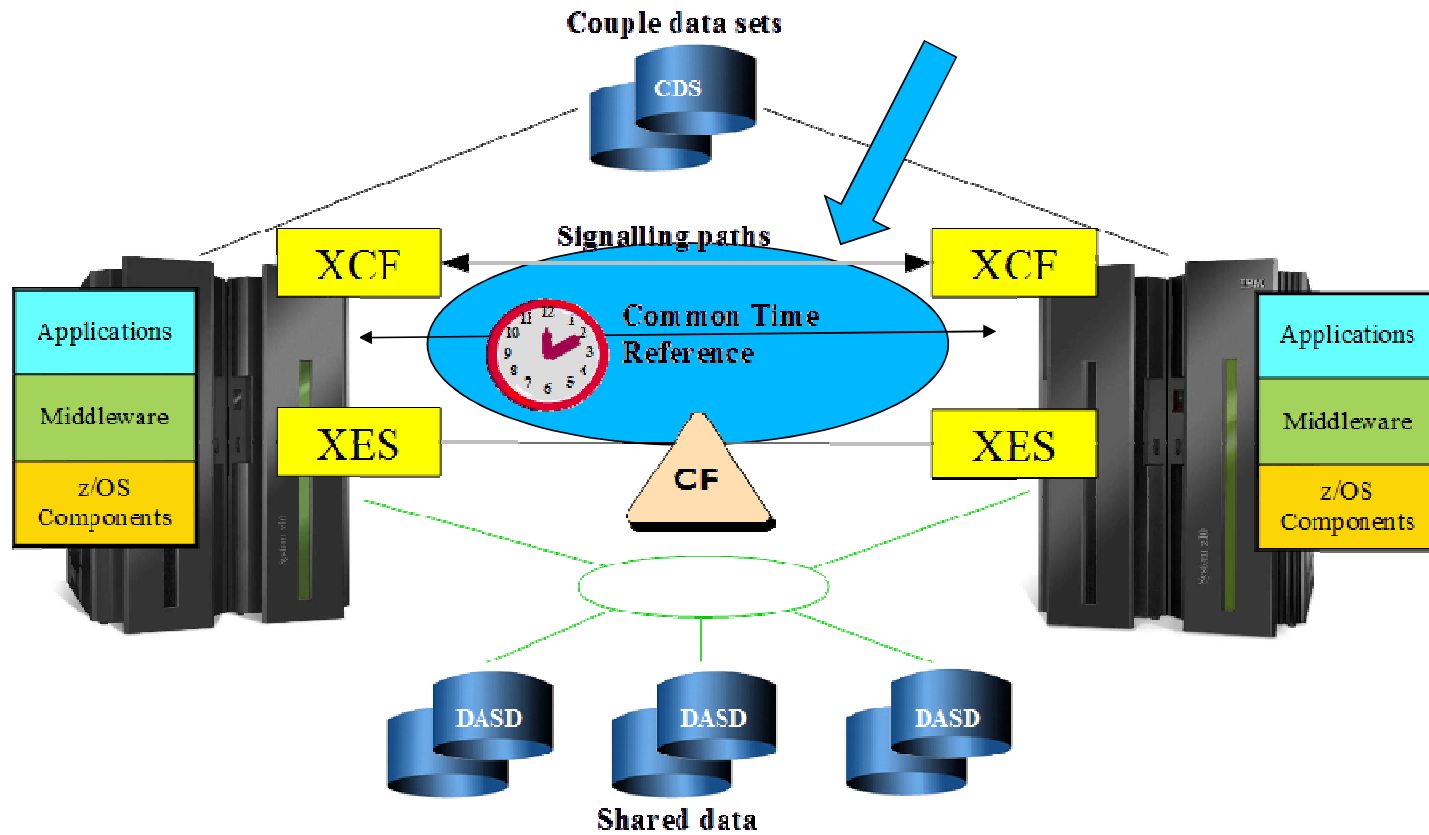
Other considerations

- Generally want the fastest links
- Distance matters
- Redundancy

- How many z/OS images?
- Spread across which CECs?

- “Cloning”

Sysplex: Common Time Reference



Time Synchronization

- z/Architecture ensures that two successive time stamps are unique within a CEC
 - Critical if you want to have unambiguous ordering of events
- Through a “common time reference”, this guarantee can be extended to time stamps taken on different CECs
 - Thus there can be an unambiguous ordering of events within the sysplex
 - A tremendous simplification for applications and people
- The “common time reference” is now implemented via Server Time Protocol (STP)

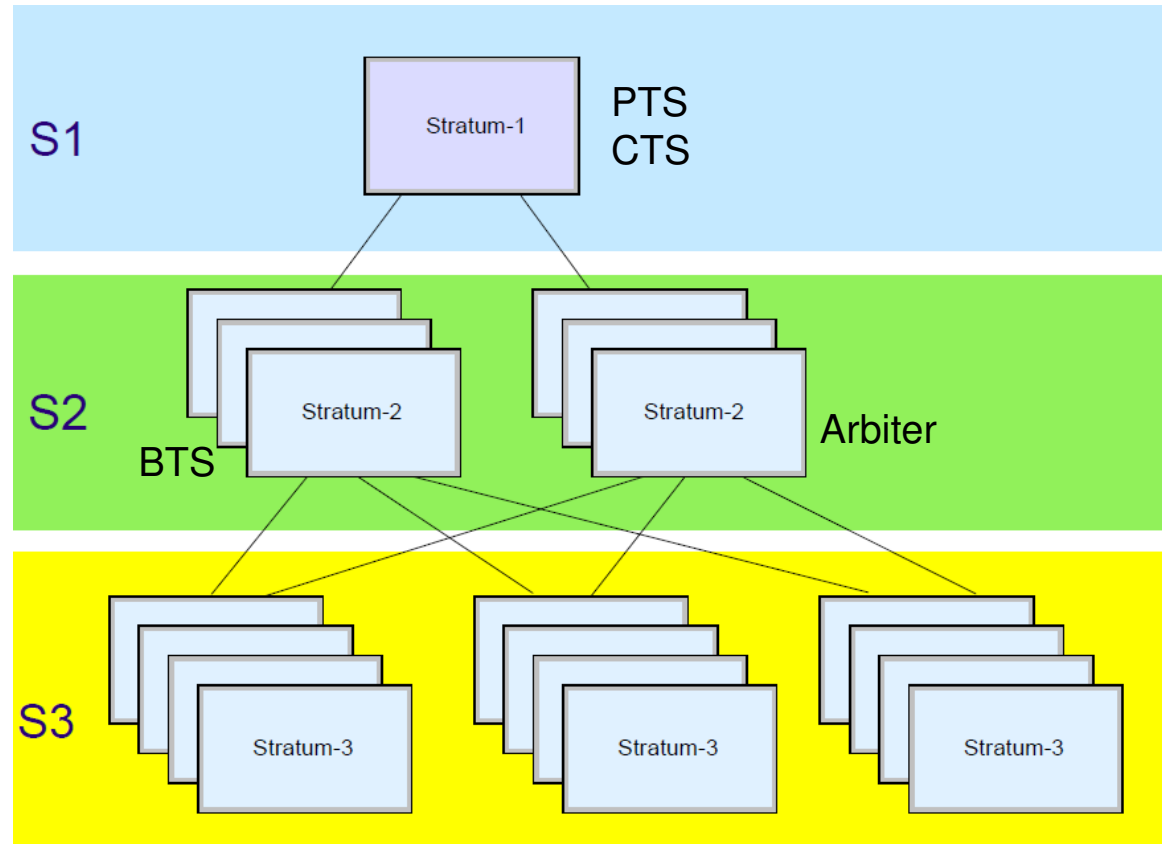
Server Time Protocol (STP)

- Licensed Internal Code (LIC) within the CEC
- Presents a single view of time to PR/SM™
- STP enabled servers can maintain time synchronization with each other
- STP is a message based protocol in which timing data is sent over coupling links that connect servers participating in a Coordinated Timing Network (CTN)

- STP can be configured to synchronize with an external time source (ETS)
 - Helpful if you want to have the same time reference across heterogeneous platforms in your enterprise
 - Exploits Network Time Protocol (NTP)
 - Accuracy to within a range of 1 to 10 milliseconds of UTC (Universal Time, Coordinated)
 - NTP is too imprecise for a sysplex common time reference, hence STP

Server Time Protocol

- Timer signals distributed in layers called “stratums”
- Servers use these timer signals to synchronize their TOD clocks to Coordinated Server Time (CST)
- Servers synchronized to CST participate in a common Coordinated Time Network (CTN) identified by a unique CTN ID
- Designated servers with the CTN are assigned STP “roles”



Coordinated Timing Network (CTN)

- CTN is a collection of servers that are time synchronized
- The Coordinated Server Time (CST) is the time to which they are synchronized
 - Can think of the CST as being “the” TOD clock for the sysplex
 - Each server uses the timer signals to determine what CST is
- Servers in the same CTN all have the same CTN ID
 - CTN ID is customer specified value that identifies a given CTN
 - Only servers with the same CTN ID can participate in a given CTN
- All servers in the CTN have identical set of time control parameters

STP Server Roles in a CTN

- Preferred Time Server (PTS)
 - The server you want to provide the CST
 - Should have connectivity to the BTS, the arbiter, and all servers that are to be stratum 2
- Backup Time Server (BTS) (if you have 2nd CEC)
 - The server that is to take over as CTS if the PTS fails
 - Should have connectivity to the PTS, the arbiter, and all stratum 2 servers connected to PTS
- Current Time Server (CTS)
 - Either the PTS or the BTS
 - Normally PTS, but role can be re-assigned as part of planned reconfiguration or failure
- Arbiter (if you have 3^d CEC)
 - Stratum 2 server that helps the BTS determine if it should take over CTS role from the PTS
 - Should have connectivity to PTS and BTS

Glossary for System z Server Time Protocol (STP)

Acronym	Full name	Comments
Arbiter	Arbiter	Server assigned by the customer to provide additional means for the Backup Time Server to determine whether it should take over as the Current Time Server.
BTS	Backup Time Server	Server assigned by the customer to take over as the Current Time Server (stratum 1 server) because of a planned or unplanned reconfiguration.
CST	Coordinated Server Time	The Coordinated Server Time in a CTN represents the time for the CTN. CST is determined at each server in the CTN.
CTN	Coordinated Timing Network	A network that contains a collection of servers that are time synchronized to CST.
CTN ID	Coordinated Timing Network Identifier	Identifier that is used to indicate whether the server has been configured to be part of a CTN and, if so, identifies that CTN.
CTS	Current Time Server	A server that is currently the clock source for an STP-only CTN.
GAS	Going Away Signal	A reliable unambiguous signal to indicate that the CPC is about to enter a check stopped state.
PTS	Preferred Time Server	The server assigned by the customer to be the preferred stratum 1 server in an STP-only CTN.

Configuring STP

- Use HMC for all time related functions
 - Initialization of processor TOD clock
 - Time zone offsets
 - Leap seconds offset
 - Daylight saving time offset
 - Scheduling changes to any of these offsets
 - Set or change CTN ID
 - Assign server roles
- z/OS CLOCKxx parmlib member
 - IBM defaults imply use of STP if configured
 - But see if you need something unique for your configuration

z/OS, Sysplex, and Time adjustments

- Set processor TOD clock to UTC
- Use relevant interfaces to make adjustments (local time, daylight savings)
 - **Use HMC to adjust time parameters for STP**
 - TIMEZONE within CLOCKxx parmlib member
 - SET CLOCK, SET TIMEZONE commands
- Never change the processor TOD clock directly once a system is in the sysplex
 - Risks data base corruption
 - Systems might not be able to join the sysplex
 - Systems might be removed from the sysplex (implies non-restartable wait-state)

*Some installations have applications that require TOD clock be set to local time.
Not a good place to be.*

Things I did not talk about

- Sysplex Timer, ETR, Mixed-CTN
 - Older technology for implementing a common time reference
 - Unfortunately, *Setting Up A Sysplex* is still mostly geared towards these
- Sysplex contained entirely within a single CEC can use processor clock as the common time reference
 - Specify SIMETRID in CLOCKxx parmlib member and avoid use of STP, but ...
 - There are good reasons for using STP even for a single CEC
- Failures and recovery
 - You have work to do as understanding and resolving the various failure scenarios is critical
 - A system that loses access to the “common time reference” cannot participate in the sysplex
 - In other words, it will wait-state

STP References for Additional Information

▪ Redbooks

- Server Time Protocol Planning Guide, SG24-7280
 - <http://www.redbooks.ibm.com/abstracts/sg247280.html>
- Server Time Protocol Implementation Guide, SG24-7281
 - <http://www.redbooks.ibm.com/abstracts/sg247281.html>
- Server Time Protocol Recovery Guide, SG24-7380
 - <http://www.redbooks.ibm.com/abstracts/sg247380.html>

▪ TechDocs

- <http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102019> (avoid outages)
- <http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102037> (recovery voting)
- <http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TD105103> (restore STP config after power on reset)
- <http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102081> (STP and leap seconds)
- <http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS2398> (STP overview)

▪ Education

- Introduction to Server Time Protocol (STP)
 - Available on Resource Link
 - www.ibm.com/servers/resourcelink/hom03010.nsf?OpenDatabase

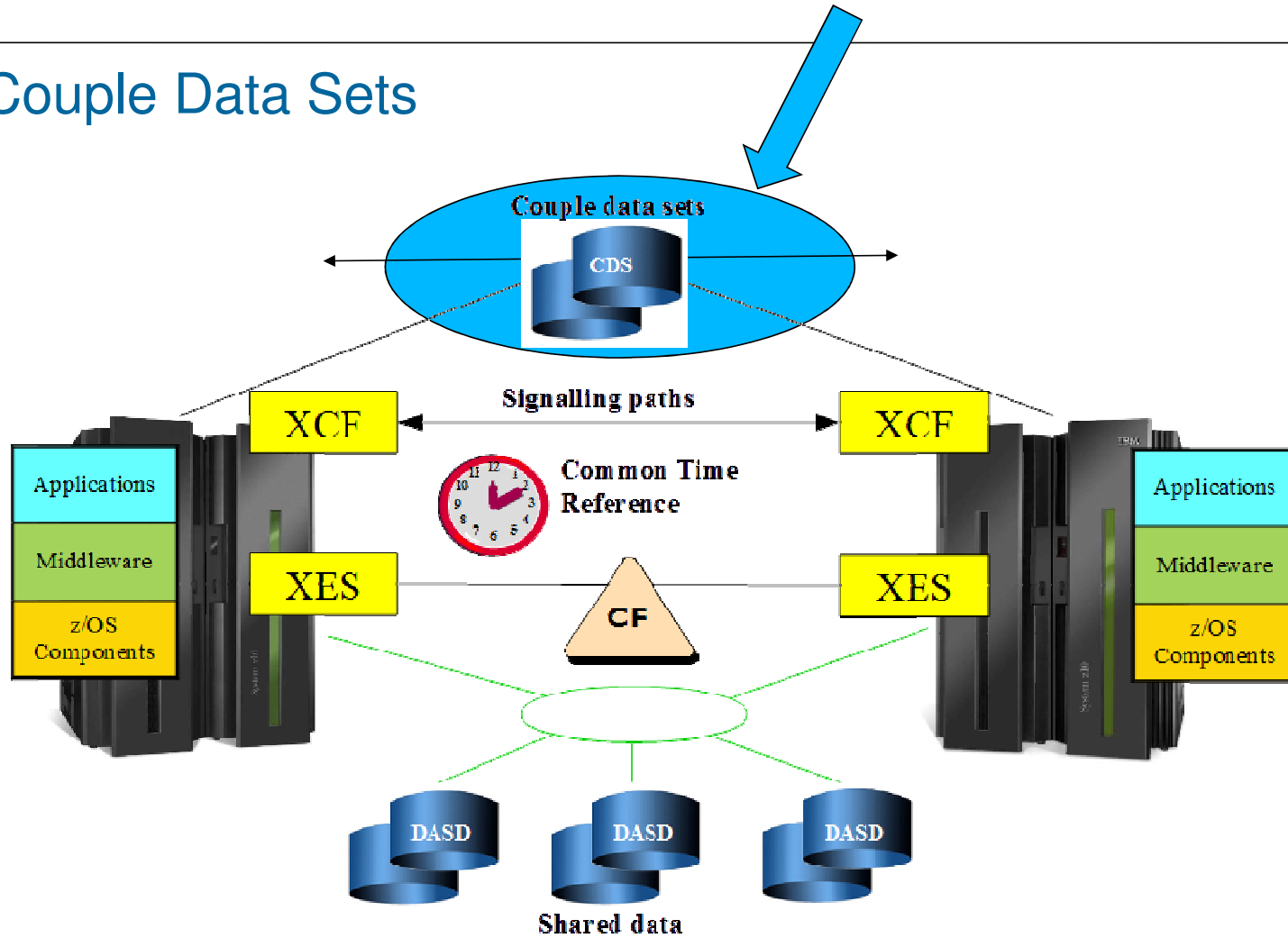
▪ STP Web site

- www.ibm.com/systems/z/pso/stp.html

▪ Systems Assurance

- The IBM team is required to complete a Systems Assurance Review (SAPR Guide, SA06-012) and to complete the Systems Assurance Confirmation Form via Resource Link
- <http://w3.ibm.com/support/assure/assur30i.nsf/WebIndex/SA779>

Sysplex: Couple Data Sets



Couple Data Sets Two Categories

- Sysplex Couple Data Sets which “define” the sysplex
- Function Couple Data Sets which support sysplex “functions”

Type

CFRM

ARM

SFM

WLM

LOGR

BPXCMDS

Exploiter

Coupling Facility Resource Manager

Automatic Restart Manager

Sysplex Failure Manager

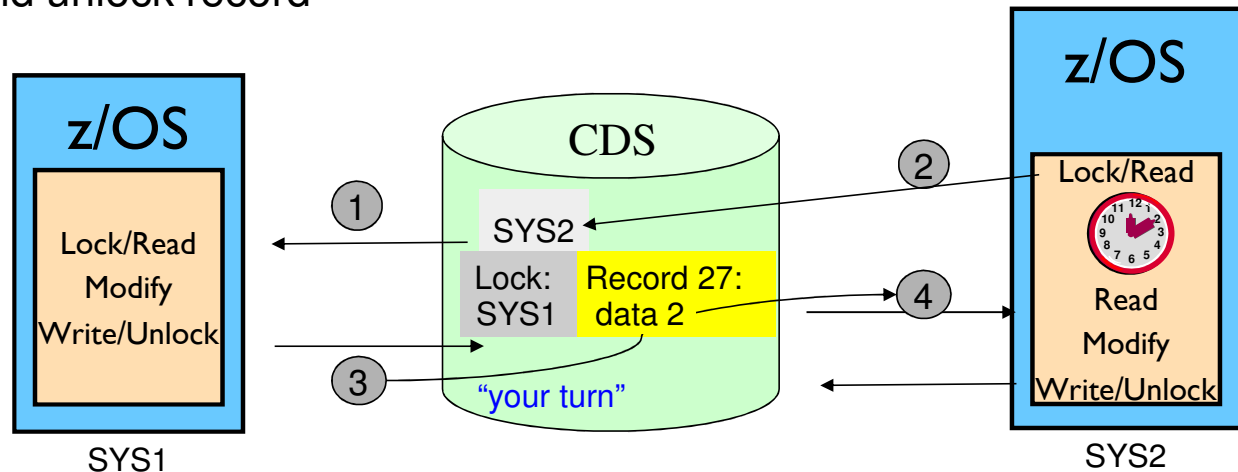
Workload Manager

z/OS Logger

z/OS UNIX System Services

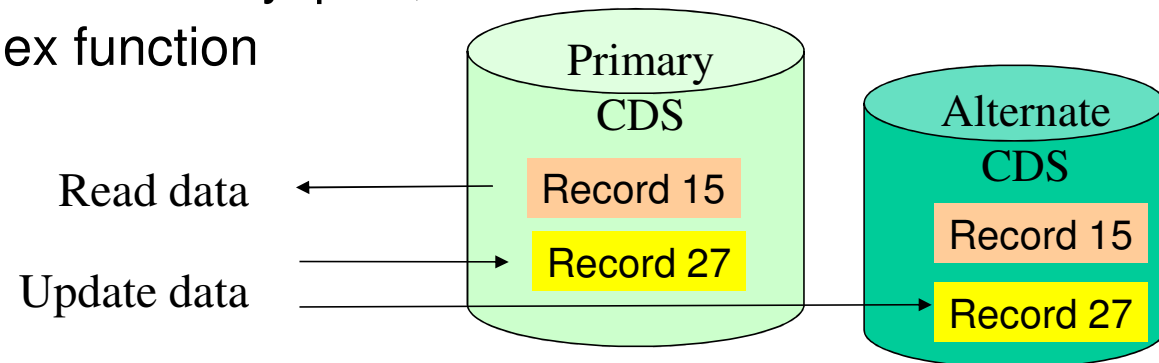
Couple Data Sets (CDS)

- Provide means to harden data and share it between systems in the sysplex under serialization
- Accessed via XCF channel programs and protocols
 - Typical usage
 - Lock record and read content into storage
 - Modify in-store copy
 - Write modified content to CDS and unlock record
 - “Lock steal protocols” to mitigate sympathy sickness

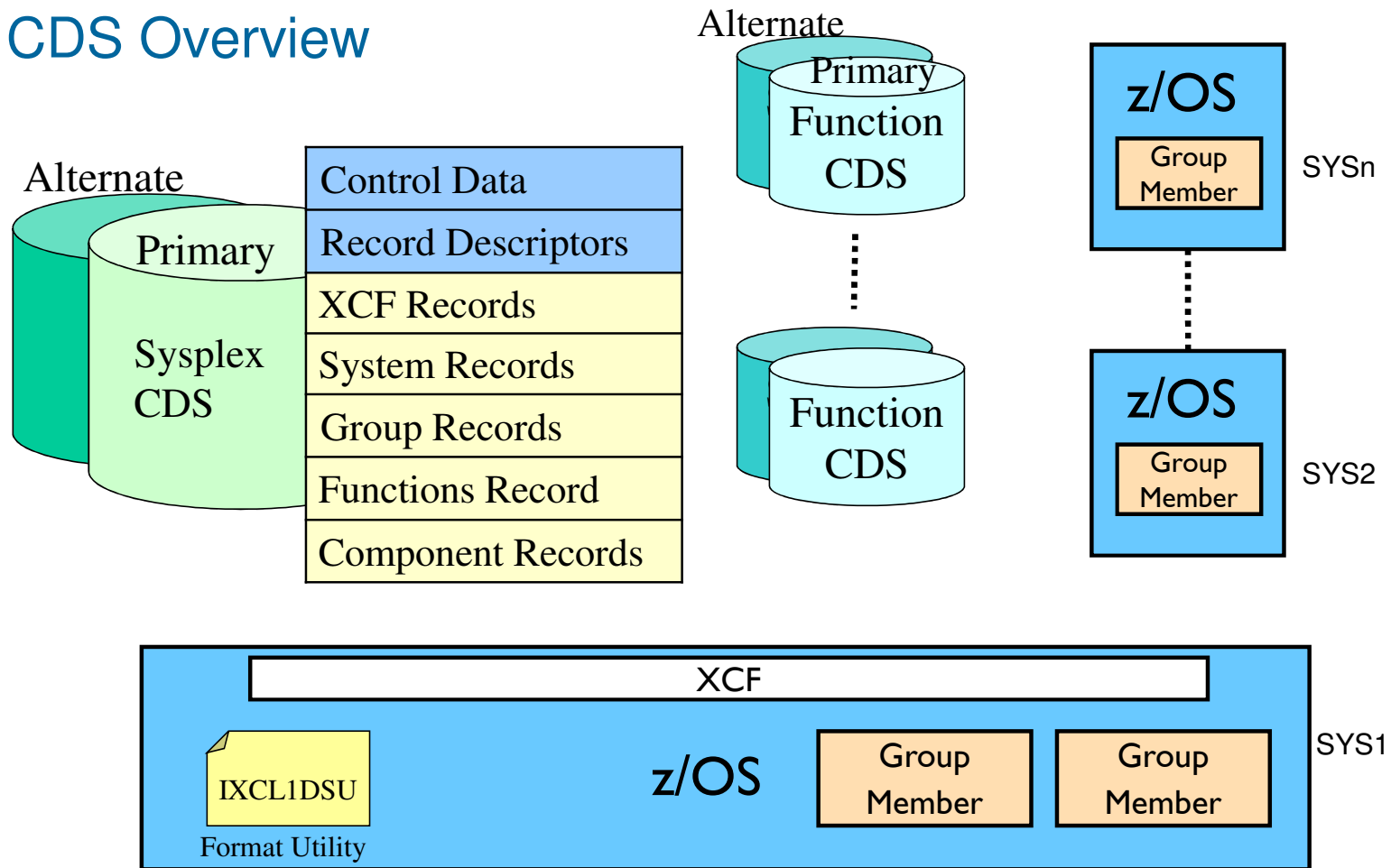


Couple Data Sets ... Primary and Alternate

- Normally run with both Primary and Alternate CDS
 - Read requests directed to primary
 - Update requests written first to primary, then to alternate
 - Both must complete for I/O request to finish
- Sysplex automatically switches to alternate if primary fails
- **Loss of both primary and alternate can be disastrous**
 - Wait-state of every system in the sysplex, or
 - Significant loss of sysplex function



Sysplex CDS Overview



System Programmer Tasks ... Initial Setup

- Create the necessary couple data sets
 - CDS Format Utility (IXCL1DSU)
 - Sysplex CDS is required (if not XCF-Local mode)
 - Zero or more function CDS's as needed
- Make the CDS available to the sysplex
 - COUPLExx parmlib member
 - SETXCF COUPLE command
- As needed, put data into the (function) CDS
 - Some functions use “policies” (data) to control their behavior
 - Installation defines the specific policies appropriate to given function
 - The function CDS contains the policies (data)
 - Administrative Data Utility (IXCMIAPU) creates, updates, deletes, or reports on the data in the CDS (note: not used for WLM)
- As needed, activate desired policy for relevant functions
 - SETXCF START,POLICY command
 - Perhaps COUPLExx

IXCMIAPU also called the
“Administrative Policy Utility”

System Programmer Tasks ... Changing CDS

- PSWITCH – switch to a new primary CDS
 - Stop using current primary CDS
 - Make current alternate be new primary
 - *Now you should ACOUPLE since you have a single point of failure (SPOF)*
- ACOUPLE – bring a new alternate CDS into service
 - Stop using current alternate (if any)
 - Make indicated CDS be new alternate (content of primary copied to alternate)
- PCOUPLE – initiate use of a function
 - If sysplex already using function, use the CDS already in use by the sysplex
 - If not, make indicated CDS be primary CDS for indicated function
- Keep COUPLExx parmlib member in sync with CDS configuration
 - This is very, very important

System Programmer Tasks ... Changing Policies

- Create, update, or delete an administrative policy
- Activate a new or changed policy
- Stop using a policy

- All policy based functions have similar concepts, but there are differences to be understood
 - CFRM, ARM, SFM
 - LOGR
 - WLM

Policies are not applicable to SYSPLEX or BPXMCDS

System Programmer Tasks ... Related Concerns

- CDS Placement (next slide)
 - Availability
 - Performance
 - Failure isolation
- CDS Failures
 - Avoiding single points of failure
 - In conjunction with unresponsive/failed systems
- Sysplex IPL and Disaster Recovery
 - What CDS's to use
- Various Risks
 - Mirroring of CDS
 - Using an old CDS copy

CDS Placement ... Considerations

- Accessible volumes
 - No RESERVE
 - No migration/backups (mitigate)
- Performance
 - Enable DASD Caching (fast write)
 - Avoid volumes with highly used data sets
 - No synchronous mirroring of CDS
- Mitigate Contention
 - Isolate Sysplex CDS, CFRM CDS, possibly LOGR CDS
 - CFRM Message Based Processing, Enable CFLCRMGMT
 - Monitors
- LOGR CDS
 - Might need to be part of consistency group for DR
 - [Exception to no synchronous mirroring](#)
- Failure Isolation
 - Primary vs Alternate (vs spare)

Consider dedicating
volume exclusively
for CDS use

Do not synchronously mirror
sysplex CDS or CFRM CDS

What about async mirroring?

CDS Considerations ... Restrictions

- Multiple extents not supported
- Cannot span multiple volumes
- Can only be used by one sysplex

Security

- Protect the CDS
 - Only XCF should be accessing these data sets
- Format Utility
 - Always creates new data sets
 - So nothing beyond normal data set naming profiles
 - Still, think about who can issue SETXCF or change COUPLExx
- Administrative Data Utility
 - Need to control use since the utility can change content of CDS in a way that could be detrimental
 - Programs using it must reside in APF Authorized library
- Some functions may have additional security profiles that need to be set up

Couple Data Set Format Utility ... Overview

- The format utility program IXCL1DSU creates and formats a Couple Data Set (CDS) using parameters that you specify
- These parameters determine type and number of records that need to be created in the CDS
- Should create a primary, alternate, and spares

```

//FMTDCS   JOB      ....
//STEP1    EXEC    PGM=IXCL1DSU
//STEPLIB  DD      DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD      SYSOUT=A
//SYSIN    DD      *
DEFINEDS   SYSPLEX(name of sysplex)
           DSN(cds data set name) VOLSER(volume)
           MAXSYSTEM(number of systems)
           DATA TYPE(type name)
           ITEM NAME(record name) NUMBER(count)
           ITEM NAME(record name) NUMBER(count)
           ....

```

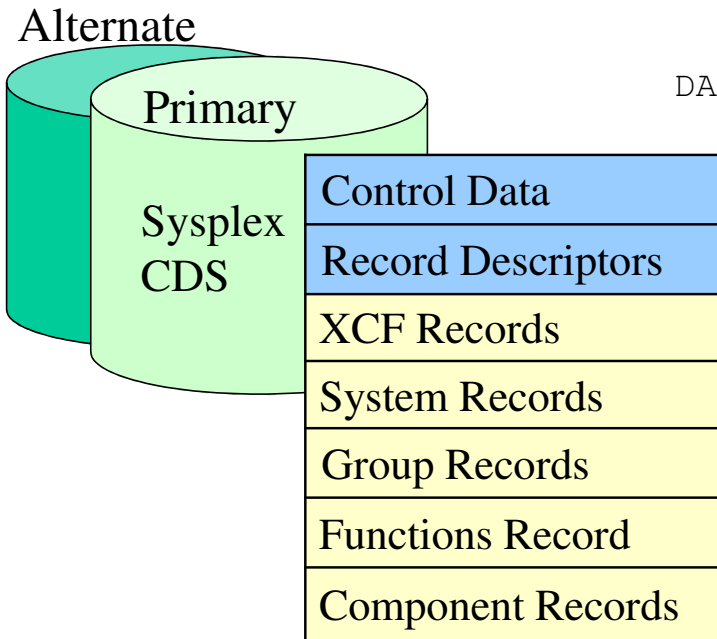
Can STEPLIB to appropriate version/level

Must match sysplex name specified in COUPLExx

One DEFINEDS for each CDS to be created

These ITEMS are unique to indicated TYPE

Couple Data Set Format Utility ... SYSPLEX CDS



DEFINEDS

```

...
MAXSYSTEM(#systems in sysplex)
DATA TYPE (SYSPLEX)
ITEM NAME (GROUP) NUMBER(#groups to support)
ITEM NAME (MEMBER) NUMBER(#members per group)
ITEM NAME (GRS) NUMBER(1) ← If GRS-Star mode
ITEM NAME (CLUSTER) NUMBER(1)
ITEM NAME (SSTATDET) NUMBER(1) ← A "must have"

```

- Note more kinds of records than ITEMS
- Some choices are release dependent
 - ITEMS, even NUMBER values
 - Implies CDS "versioning"
- Choices might impact number of records or their size, and protocols
- Discovering groups and members ?
- Allow for growth, but

See SYS1.SAMPLIB(IXCSYSPF)

Perils of too much white space

- White space implies
 - Longer records
 - More records
- Which can elongate
 - CDS I/O response time
 - IPL time
 - ACOUPLE time
 - PSWITCH time
- Can dynamically grow as needed, but changing to a smaller CDS implies:
 - **Sysplex IPL**
 - For Sysplex or CFRM CDS
 - **Loss of function**
 - For ARM, SFM, WLM, LOGR, or BPXMCDS CDS
 - Perhaps only temporary

Charts extracted from IBM Redbook: SG24-7816
 “System z Mean Time To Recovery Best Practices”

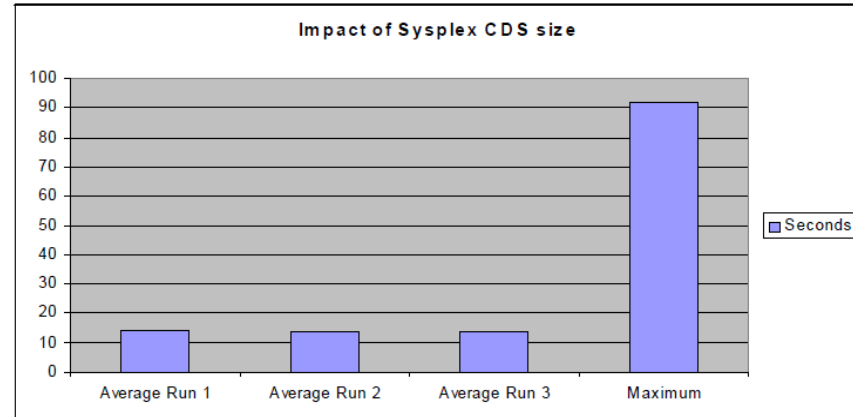


Figure 4-11 Impact of sysplex CDS size on IPL times

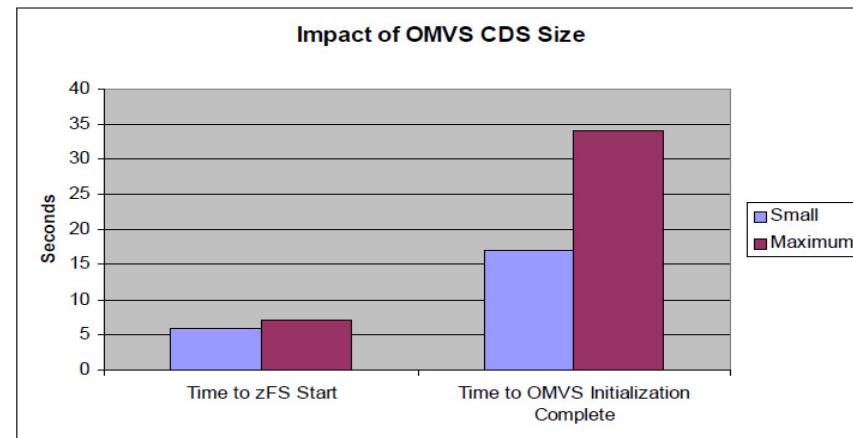


Figure 5-2 Results of BPXMCDS size on OMVS initialization time

Creating Sysplex CDS Content

- Nothing for you to do beyond:
 - Formatting the CDS
 - Bringing CDS into service
- All data is updated dynamically by the systems in the sysplex

Specifying Couple Data Sets at IPL Time

- COUPLExx parmlib member

Designates Sysplex CDS {
 COUPLE SYSPLEX(*sysplex name*)
 PCOUPLE(*primary_cdsname*, *volser*)
 ACOUPLE(*alternate_cdsname*, *volser*)
 <other keywords for COUPLE statement>

Designates functions to be used by system {
 DATA TYPE(*type name*)
 PCOUPLE(*primary_cdsname*, *volser*)
 ACOUPLE(*alternate_cdsname*, *volser*)
 <other statements>

Must be same name specified when the CDS were formatted

- Every system in sysplex needs to be using the same Sysplex CDS
- All systems using a given function must use the same Function CDS
 - For most practical cases, all systems will need access to the function CDS
 - Though this is not necessarily required (it depends)

Dynamically Defining a Primary Function CDS

- In general, you will almost always use COUPLExx to indicate which couple data sets are to be used by a system
 - COUPLExx is the only way to get a system to use a Sysplex CDS
- You can use the SETXCF COUPLE command to dynamically define a primary function CDS for a given service in these particular two cases:
 - The function is not in use by any system in the sysplex
 - The function is in use somewhere in the sysplex, but not by this system
- SETXCF COUPLE,TYPE=*typename*,PCOUPLE=(*cdsname*,*volser*)
 - typename* is one of CFRM, ARM, SFM, WLM, LOGR, BPXMCDS
 - If the function CDS has multiple types, you can code TYPE=(type1,type2,...)
 - cdsname* is name of data set formatted with IXCL1DSU
 - volser* is optional if CDS is cataloged

Beware !

- Once you PCOUPLE the CFRM CDS to a system, the system will wait-state if it loses access (to both primary and alternate)
 - Even if you never start a CFRM policy
- Loss of access to any other function CDS generally implies loss of function

Why Might CDS Configuration Need to Change?

- Growth
 - CDS needs more capacity
 - Support new function
 - Support new features of an existing function
- Failure
 - Loss of either primary or alternate CDS
- Placement
 - Reconfiguring DASD
 - Reconfiguring sysplex

I did not include any information about how to determine if you are running out of room in your CDS. See Redbook “System z Parallel Sysplex Best Practices”

Changing the CDS Configuration

■ Your toolkit

- Format utility IXCL1DSU
- SETXCF COUPLE,TYPE=*typename*,ACOUPLE=(*cdsname*,*volser*)
- SETXCF COUPLE,TYPE=*typename*,PSWITCH

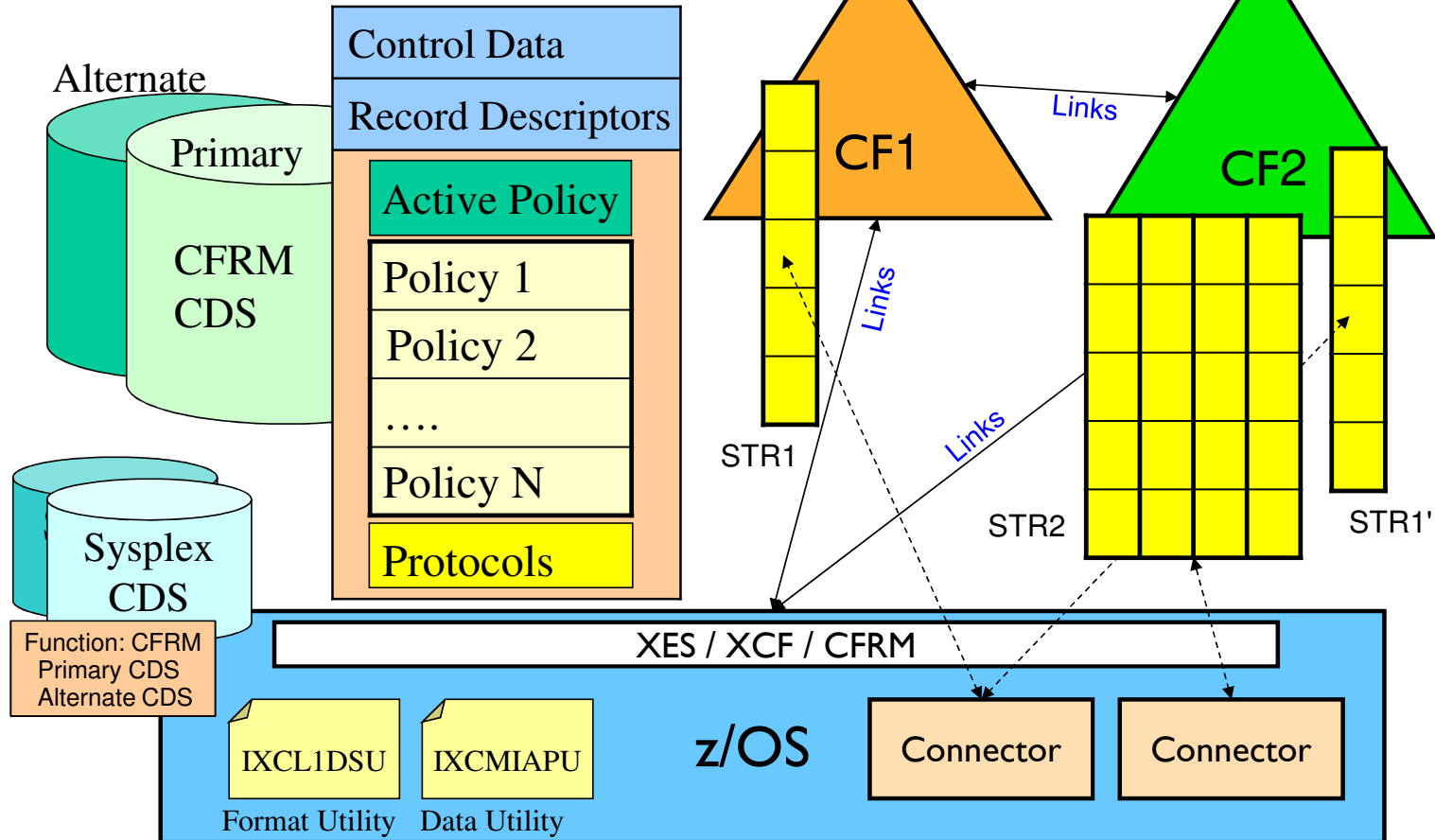
If GDPS, follow their procedures



■ Do one or more of the following as needed in an appropriate order:

- Format new CDS (one or more)
- ACOUPLE to bring new CDS into service as alternate CDS
- PSWITCH to make current alternate CDS become primary CDS
- ACOUPLE to bring another alternate CDS into service to eliminate SPOF
- Update COUPLExx parmlib member(s) to reflect new CDS configuration being used by sysplex
- Delete old CDS if no longer needed

CFRM CDS Overview



Creating CFRM CDS Content

- You must
 - Format the CFRM CDS
 - Bring the CDS into service by the sysplex
 - Create one or more administrative policies
 - Choose and activate the policy to be used by the sysplex
- The CFRM CDS contains data from different sources
 - Protocols Static data set by Format Utility (you)
 - Administrative policies Static data set by Data Utility (you)
 - Active policy Dynamic data set by systems in the sysplex
- Your influence on the active policy is limited to the exact point in time that you activate a given static copy of a policy
 - When you activate a policy, the static copy of the administrative policy is copied into active policy
 - If you later change the static copy, it does nothing to the active copy
 - Sysplex only looks at an administrative policy when you activate it

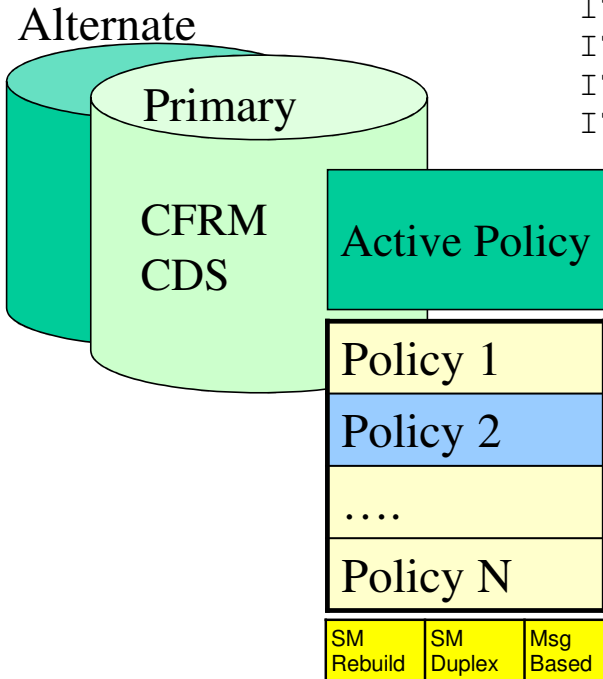
Couple Data Set Format Utility CFRM CDS

DEFINEDS

```

...
MAXSYSTEM( #systems )
DATA TYPE (CFRM)
ITEM NAME (POLICY)    NUMBER ( #admin policies )
ITEM NAME (CF)        NUMBER ( max #CF defined in a policy )
ITEM NAME (STR)       NUMBER ( max #structures in a policy )
ITEM NAME (CONNECT)   NUMBER ( max #connectors per str )
ITEM NAME (MSGBASED)  NUMBER ( 1 )
    
```

Must match sysplex CDS



- Some choices are release dependent
 - ITEMS, even NUMBER values
 - Implies CDS “versioning”
- Choices might impact number of records or their size, and protocols
- Discovering structures ?
- Allow for growth, but not too much

See SYS1.SAMPLIB(IXCCFRMF)

Creating CFRM Policies

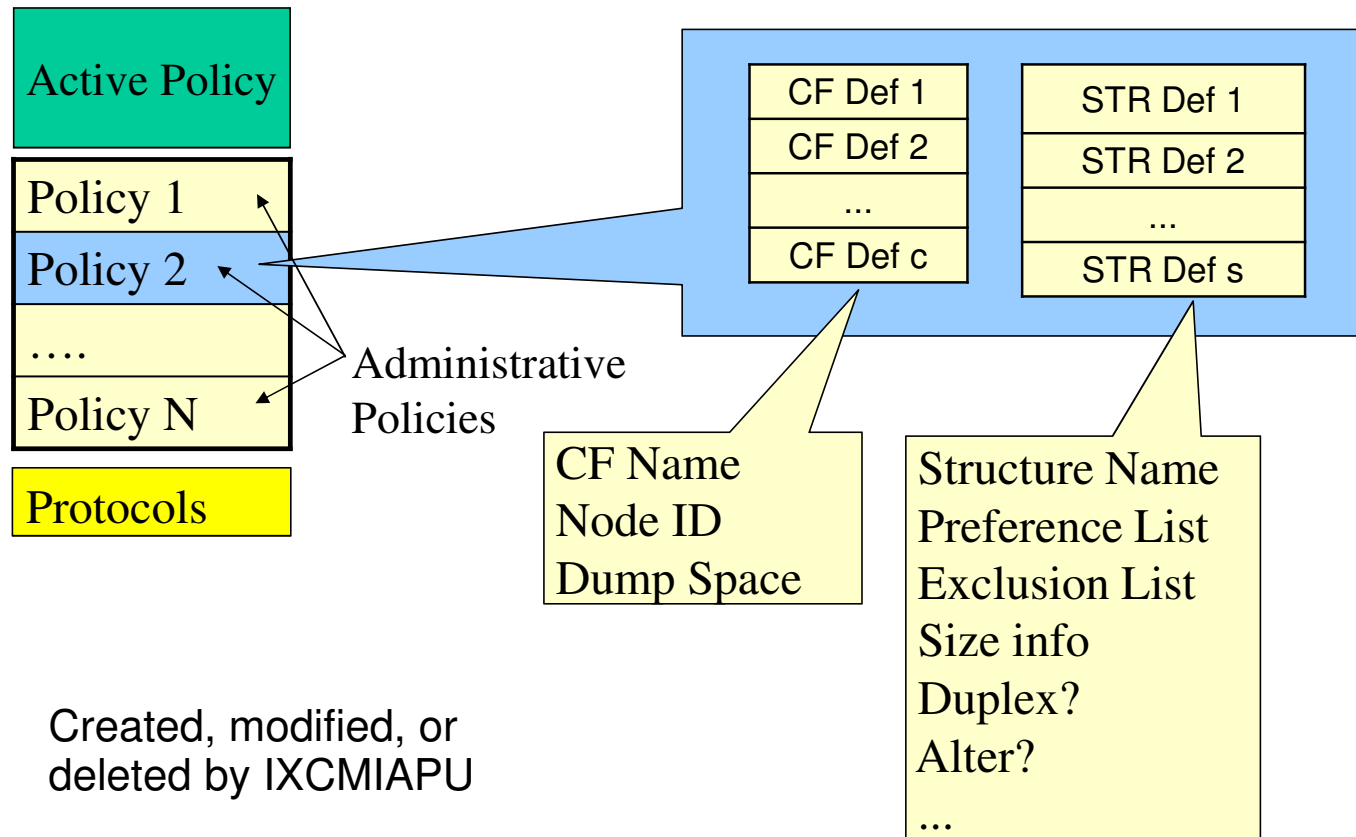
- The data utility program IXCMIAPU is used to manipulate administrative policies in a Couple Data Set formatted for CFRM
- The parameters you specify define the policy, which in turn describes how CFRM is to manage coupling facility resources
- The data utility can
 - Create new policies
 - Replace existing policies
 - Delete policies
 - Report on the policies currently defined in the CDS

- See SYS1.SAMPLIB(IXCCFRMP)

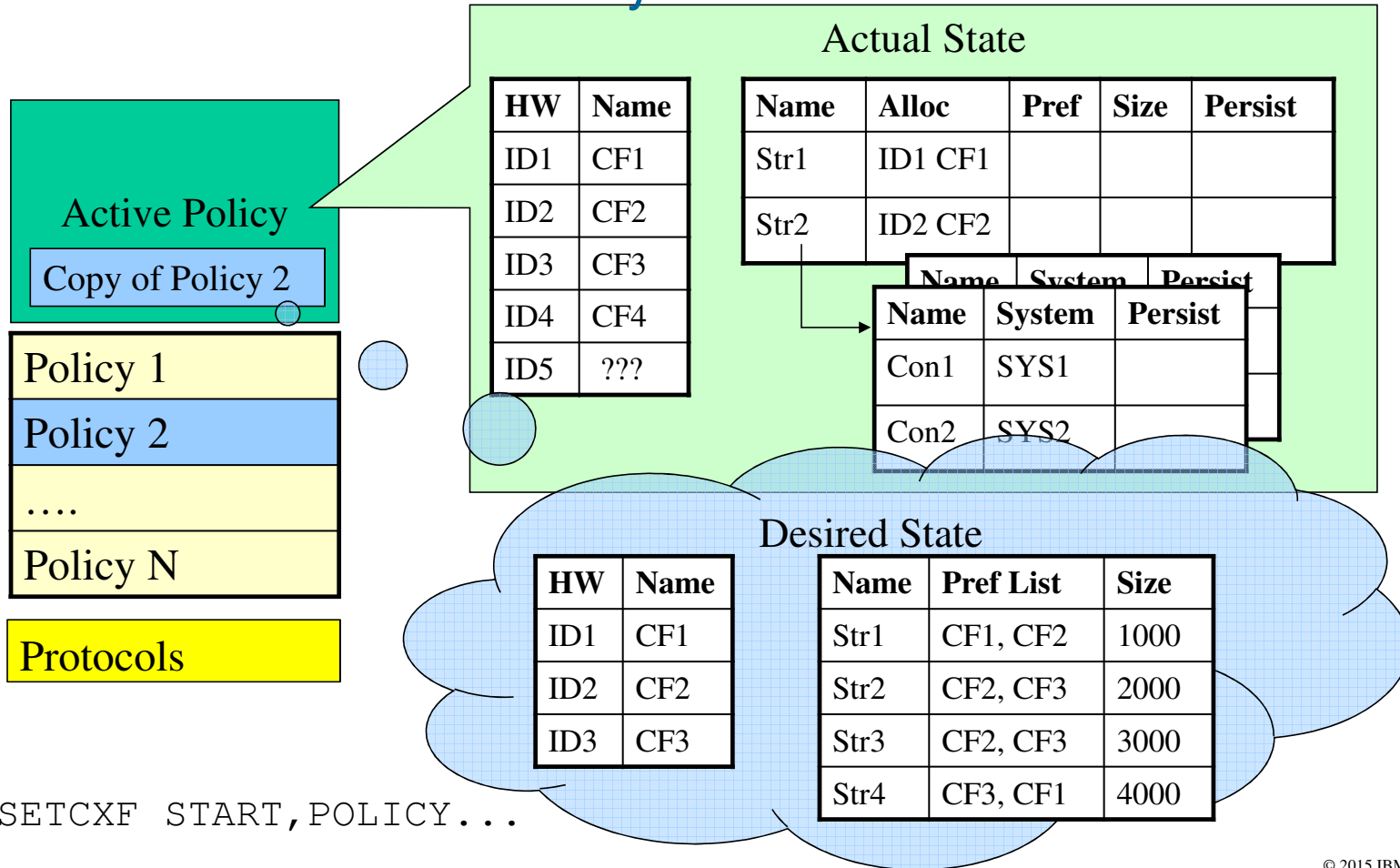
CFRM Policies Are Used to Manage CF Resources

- Define coupling facilities to be used by the sysplex
 - CF can only be used by one sysplex at any one time
- Define structures that can be used by the sysplex
- For each structure you specify:
 - Name
 - List of CFs that are candidates to host structure
 - List of other structures that should not be in the same CF as this one
 - Amount of space structure can consume in host CF
 - Protocols and thresholds to be applied to the structure
 - Duplexing of structure content in an alternate CF
 - Dynamic reconfiguring of structure to match exploiter usage
 - Full threshold for alerts and reconfiguration actions
 - Connectivity threshold at which to initiate rebuild of structure

CFRM CDS Content - Administrative Policies



CFRM CDS Content - Active Policy



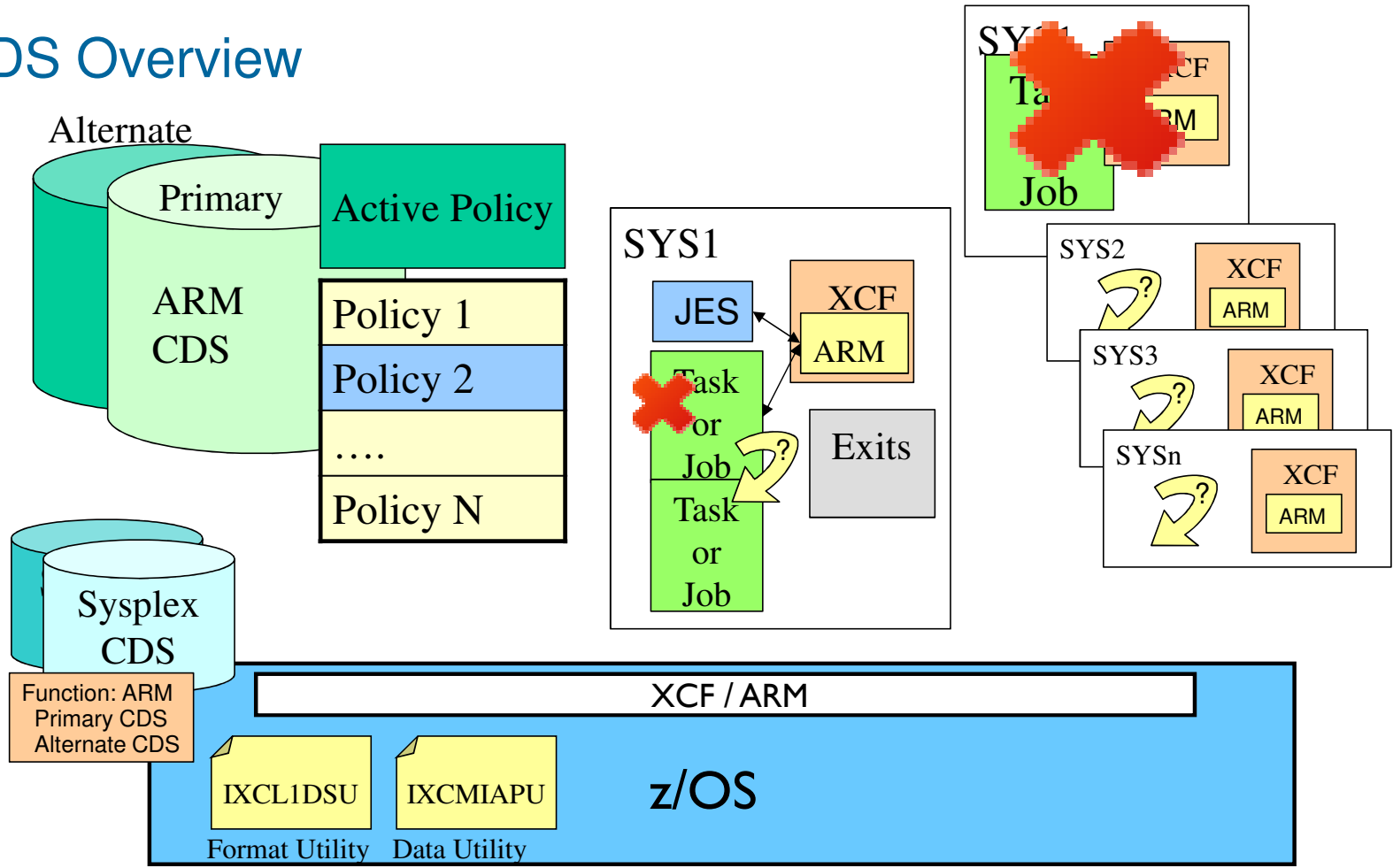
CFRM Policies – Key Things to Understand

- Format utility IXCL1DSU creates and formats CFRM CDS
 - Determines number and size of policy records, but not their content
- Data utility IXCMIAPU manipulates the administrative policy records
 - Creates, updates, or deletes content of administrative records
 - Has no impact on the active policy because the active policy uses a point in time copy of an administrative policy taken when the policy is started
- A policy must be started to get the sysplex to use it
 - SETXCF START,POLICY=polname,TYPE=CFRM
 - If you change the administrative copy, you must issue a new start command to have the changes reflected in the active copy (even if same policy name)
- Once a policy is started, the active copy remains in force
 - Persists in CFRM CDS until stopped or a new policy started
 - In particular, active policy will be in effect upon sysplex IPL with that CDS

Sometimes ACOUPLE and IXCMIAPU Confusion

- For example:
 - Format new CDS
 - Run data utility to create new policies in the new CDS
 - ACOUPL the new CDS
 - Which copies content of current primary CDS to the new alternate
 - Oops, lost the new policies that were put in the new CDS
- When using the data utility, you either:
 - Manipulate administrative policies in primary CDS currently used by sysplex
 - XCF will automatically propagate changes to the alternate CDS
 - Manipulate administrative policies in an off-line CDS that is to become the primary CDS for:
 - A new sysplex, or
 - An existing sysplex that has never before used the function
 - An existing sysplex for which you somehow made it through the task of eliminating use of the function CDS

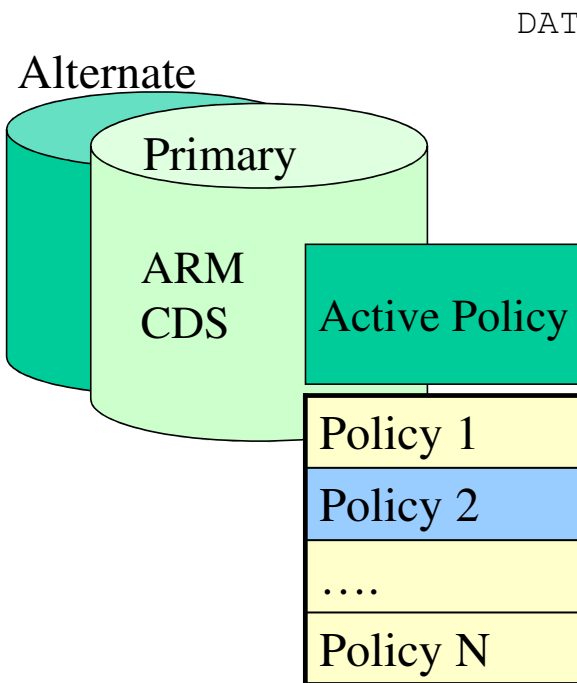
ARM CDS Overview



ARM Policies Control Restart of Work After a Failure

- Whether restarts will occur for some, none, or all elements
 - The method by which an element is to be restarted
 - Grouping and dependencies among elements
 - Which systems are candidates for hosting a restarted element
 - Restart intervals, thresholds, and time out values
-
- Note: the policy that governs the restart is the policy that is active when ARM processes the restart, not the policy that was active when the element registered with ARM

Couple Data Set Format Utility ARM CDS



```
DATA TYPE (ARM)
ITEM NAME (POLICY) NUMBER (#admin policies)
ITEM NAME (MAXELEM) NUMBER (max #elements per policy)
ITEM NAME (TOTELEM) NUMBER (max #registered elements)
```

- MAXELEM determined by your policies
- Number of restart groups and restart elements is a function of MAXELEM
- Determining TOTELEM ?

See SYS1.SAMPLIB(IXCARMF)

Creating ARM CDS Content

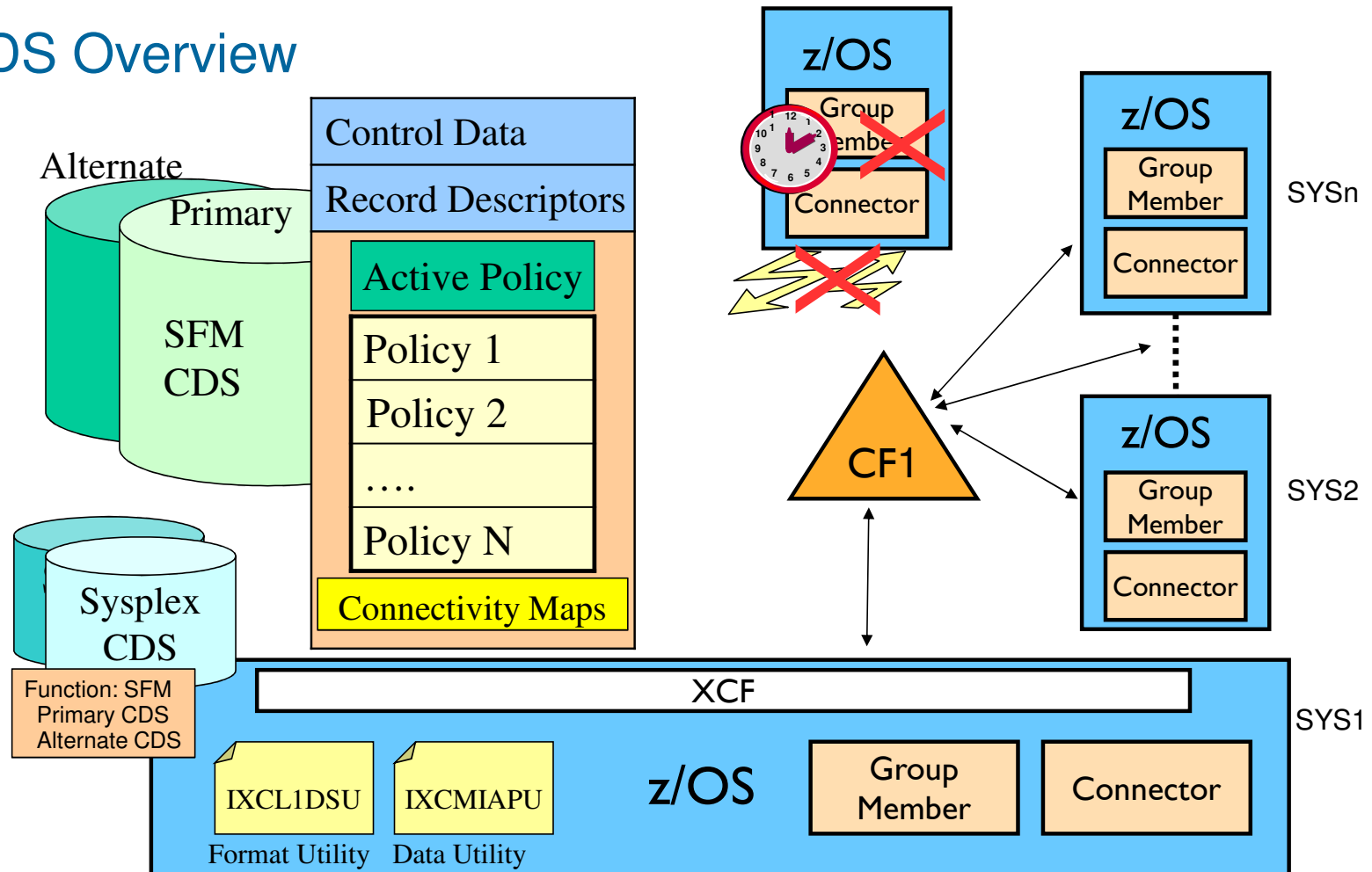
- You must
 - Format the ARM CDS
 - Create zero or more administrative policies
 - There is an implicit “no name” default policy
 - Choose and activate the policy to be used by the sysplex
- The ARM CDS contains data from different sources
 - Administrative policies Static data set by Data Utility (you)
 - Active policy Dynamic data set by systems in the sysplex
- Your influence on the active policy is limited to the exact point in time that you activate a given static copy of a policy
 - When you activate a policy, the static copy of the administrative policy is copied into the active policy
 - If you later change the static copy, it does nothing to the active copy
 - Sysplex only looks at an administrative policy when you activate it

Creating ARM Policies

- The data utility program IXCMIAPU is used to manipulate administrative policies in a formatted ARM Couple Data Set
- The parameters you specify define the policy, which in turn describes how ARM is to behave
- The data utility can
 - Create new policies
 - Replace existing policies
 - Delete policies
 - Report on the policies currently defined in the CDS

- See SYS1.SAMPLIB(IXCARMP0)

SFM CDS Overview



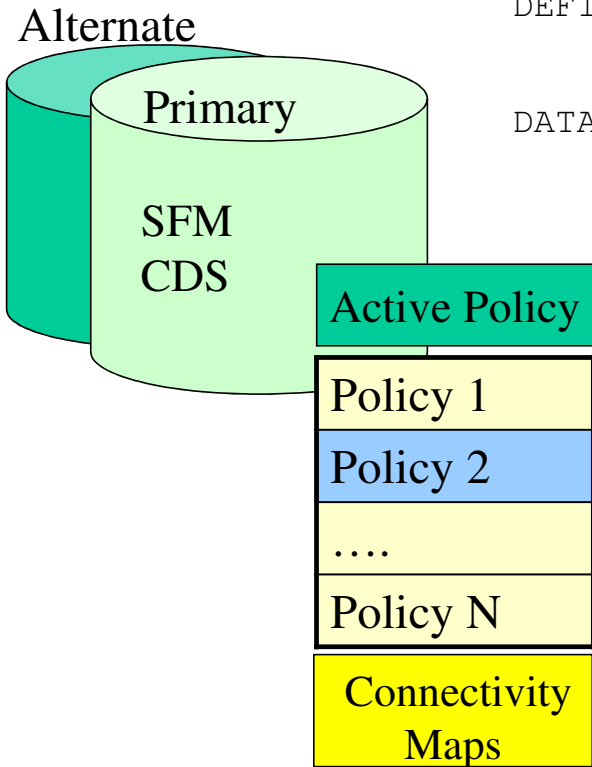
SFM Policies Help Manage Sympathy Sickness

- The systems in the sysplex can detect various problems
 - Inability of a system to participate in a sysplex
 - Sick but not dead issues
- The SFM Policy determines whether and how the sysplex is to resolve these problems
 - Unresponsive system (ISOLATETIME)
 - Based on COUPLExx Failure Detection Interval (FDI, aka INTERVAL)
 - Loss of signal connectivity (CONNFAIL)
 - Unresponsive group member (MEMSTALLTIME)
 - Unresponsive connector (CFSTRHANGTIME)

If you don't use SFM, you must understand the problems it was designed to resolve and provide appropriate mitigation through system automation and/or operational procedures.

As of z/OS V1R11, ISOLATETIME(0) is the default action unless you create an SFM policy that explicitly specifies some other action for dealing with an unresponsive system.

Couple Data Set Format Utility SFM CDS



DEFINEDS

```

...
MAXSYSTEM( #systems )
DATA TYPE ( SFM )
ITEM NAME (POLICY)      NUMBER ( #admin policies )
ITEM NAME (SYSTEM)     NUMBER ( max #system statements in a policy )
ITEM NAME (RECONFIG)   NUMBER ( max #reconfig statements in a policy )
    
```

↑
Probably not applicable

- MAXSYSTEM vs ITEM NAME(SYSTEM)

Creating SFM CDS Content

- You must
 - Format the SFM CDS
 - Create one or more administrative policies
 - Choose and activate the policy to be used by the sysplex
- The SFM CDS contains data from different sources
 - Administrative policies Static data set by Data Utility (you)
 - Active policy Dynamic data set by systems in the sysplex
- Your influence on the active policy is limited to the exact point in time that you activate a given static copy of a policy
 - When you activate a policy, the static copy of the administrative policy is copied into the active policy
 - If you later change the static copy, it does nothing to the active copy
 - Sysplex only looks at an administrative policy when you activate it

Creating SFM Policies

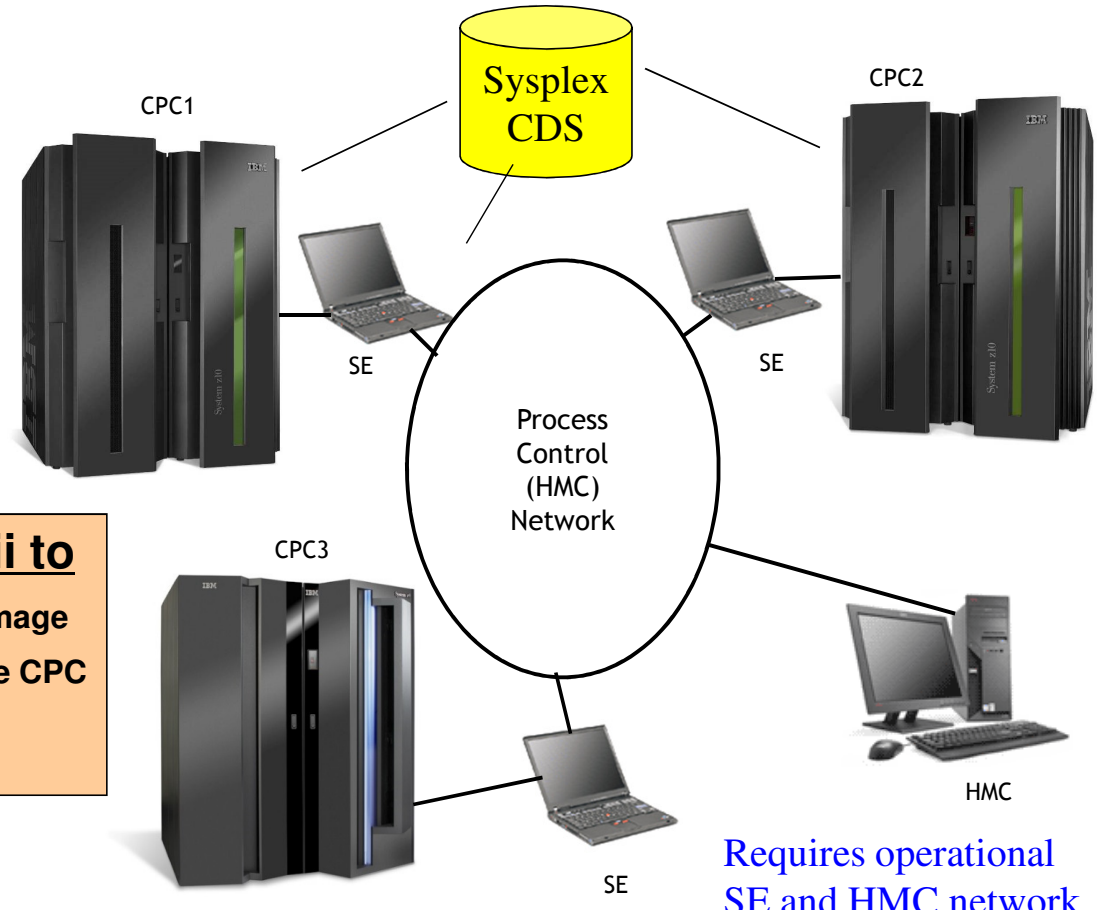
- The data utility program IXCMIAPU is used to manipulate administrative policies in a formatted SFM Couple Data Set
- The parameters you specify define the policy, which in turn describes how SFM is to behave
- The data utility can
 - Create new policies
 - Replace existing policies
 - Delete policies
 - Report on the policies currently defined in the CDS

- See SYS1.SAMPLIB(IXCSFMP)

SFM with BCPii

z/OS Images
(not VM guests)

- XCF uses BCPii to**
- Obtain identity of an image
 - Query status of remote CPC and image
 - Reset an image



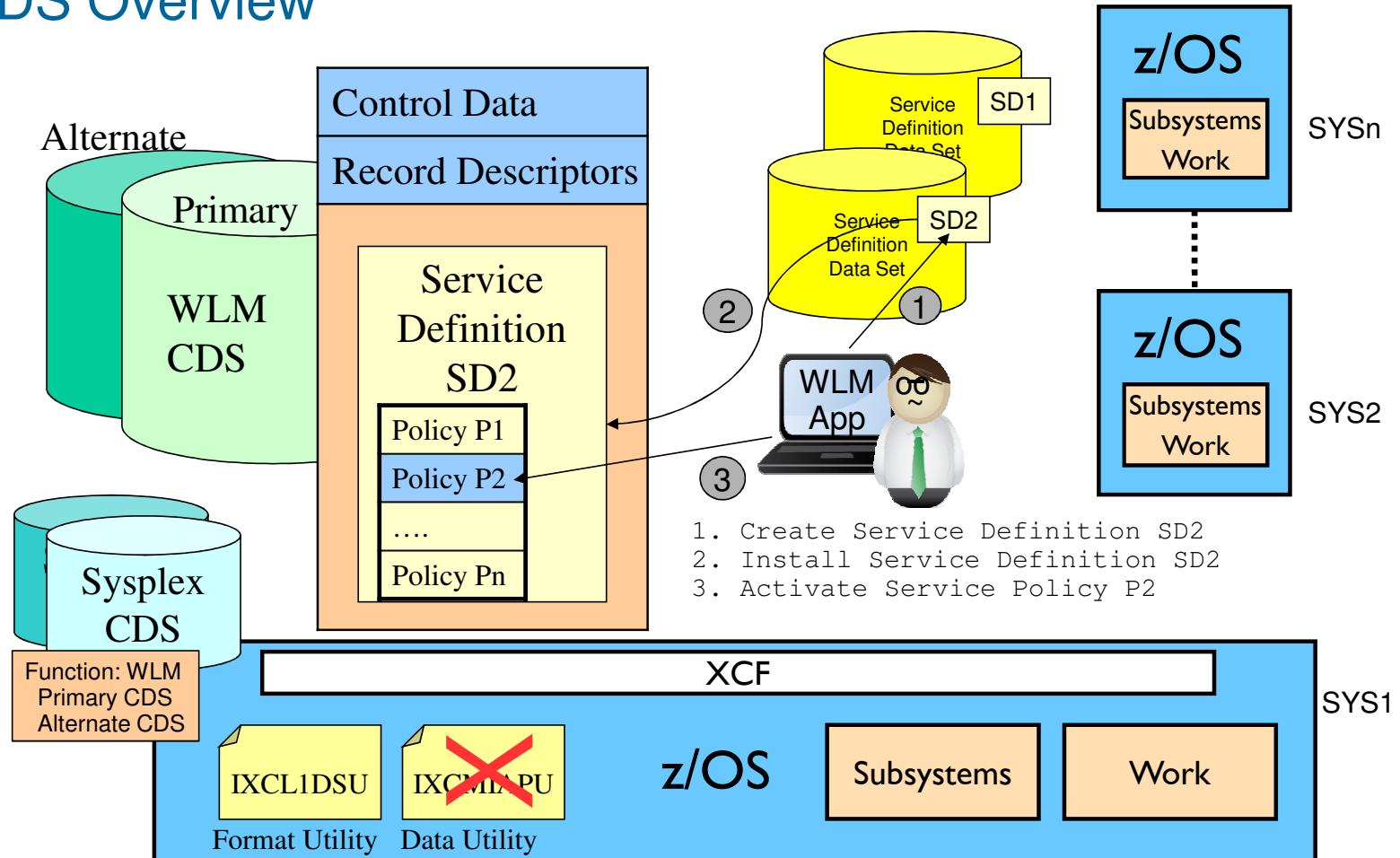
Why SFM with BCPii ?

- Expedient removal of unresponsive or failed systems is essential to high availability in sysplex
- XCF exploits BCPii services to:
 - Detect failed systems
 - Reset systems
- **Benefits:**
 - Improved availability by reducing duration of sympathy sickness
 - No waiting for FDI to expire
 - Eliminate manual intervention in more cases
 - Avoid IXC102A, IXC402D, IXC409D
 - Potentially prevent human error that can cause data corruption
 - Validate “down”

Enabling SFM with BCPii

- SFM will automatically exploit BCPii as soon as the required configuration is established:
 - Pairs of systems running z/OS 1.11 or later
 - **BCPii configured, installed, and available**
 - XCF has security authorization to access BCPii defined FACILITY class resources or TRUSTED attribute
 - z10 GA2, z196, z114, zEC12, or zBC12, z13
 - **Primary sysplex CDS in use by sysplex must have been formatted with:**
 - ITEM NAME(SSTATDET) NUMBER(1)

WLM CDS Overview



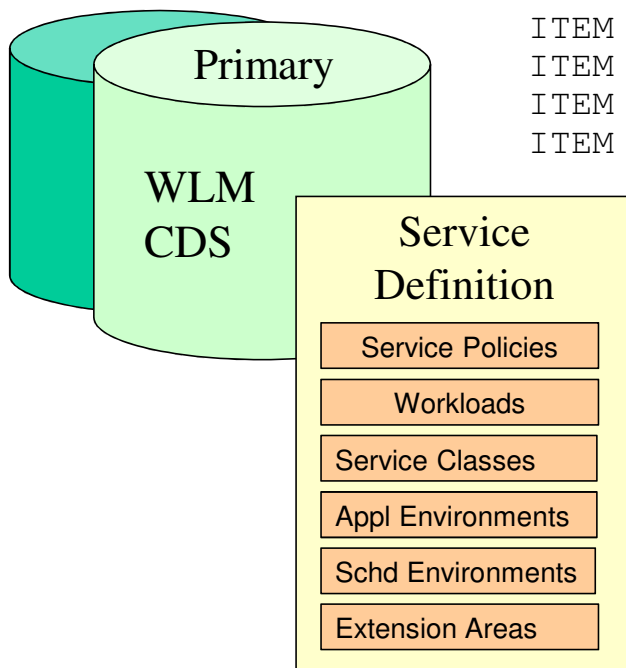
WLM Manages Work in Sysplex to Achieve Your Goals

- WLM manages competing workloads throughout the sysplex via:
 - Workload distribution
 - Workload balancing
 - Resource distribution
- In order to meet performance goals and business objectives
- As defined in a “service definition”

Couple Data Set Format Utility ... WLM CDS

DATA	TYPE (WLM)	
ITEM NAME (POLICY)	NUMBER	(<i>#workload policies</i>)
ITEM NAME (WORKLOAD)	NUMBER	(<i>#workloads</i>)
ITEM NAME (SRVCLASS)	NUMBER	(<i>#service classes</i>)
ITEM NAME (APPLENV)	NUMBER	(<i>#application environments</i>)
ITEM NAME (SCHENV)	NUMBER	(<i>#scheduling environments</i>)
ITEM NAME (SVDEFEXT)	NUMBER	(<i>#KB for service definitions</i>)
ITEM NAME (SVDCREXT)	NUMBER	(<i>#KB for classification rules</i>)
ITEM NAME (SVAEAEXT)	NUMBER	(<i>#KB for application env area</i>)
ITEM NAME (SVSEAEXT)	NUMBER	(<i>#KB for scheduling env area</i>)

Alternate



- You can use IXCL1DSU , but ...
- The WLM Administrative Application provides support to create the WLM CDS:
 - Allocate couple data set
 - Allocate couple data set using CDS values

See SYS1.SAMPLIB(IWMFTCDS)

Creating WLM CDS Content

- You must
 - Format the WLM CDS
 - Use tools to create one or more service definitions
 - Use tools to install the service definition in the CDS for use by the sysplex
 - Optionally activate a service policy in the installed service definition
- The WLM CDS contains data from different sources
 - Service Definition Static data set when “installed”
 - Active service policy Designation set when policy “activated”
- Your influence on the service definition is limited to the exact point in time when installed in the WLM CDS (or when activating policy)
 - When you install a service definition, you copy it into the CDS from the service definition data set
 - When you update the service definition, you update the copy that resides in the service definition data set. The installed copy in the CDS is not changed.
 - Sysplex only looks at the installed copy.

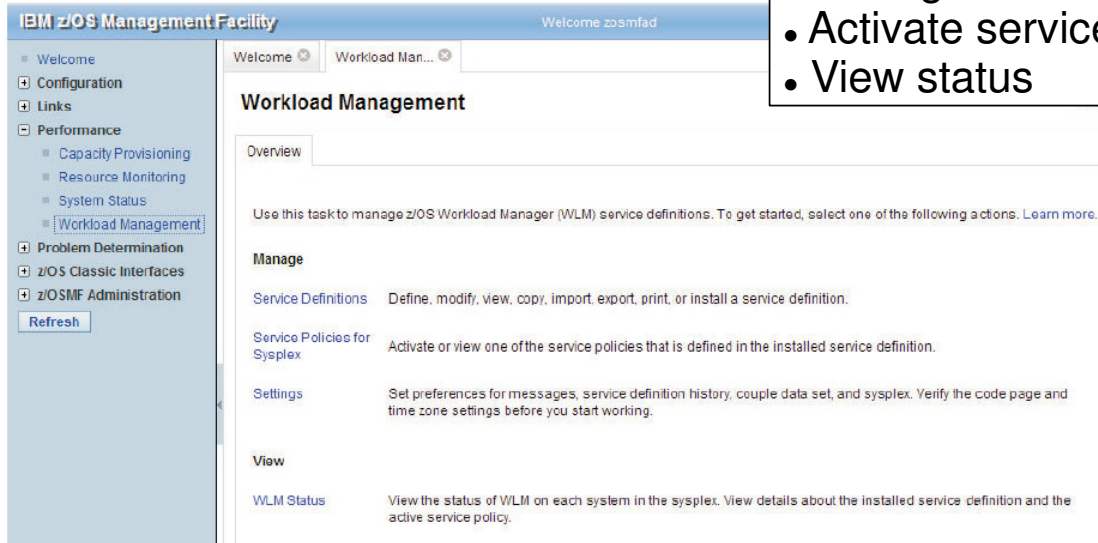
Creating WLM Policies (Service Definitions)

- z/OS WLM Administrative Application
 - Also called the “WLM ISPF application”
 - EX 'SYS1.SBLSCLI0(IWMARIN0)'
- z/OSMF

The Data Utility IXCMIAPU
is not used for WLM

Either one can be used to:

- Manage service definitions
- Activate service policy
- View status



IBM z/OS Management Facility Welcome zosmfad

Workload Management

Workload Management

Overview

Use this task to manage z/OS Workload Manager (WLM) service definitions. To get started, select one of the following actions. [Learn more...](#)

Manage

Service Definitions Define, modify, view, copy, import, export, print, or install a service definition.

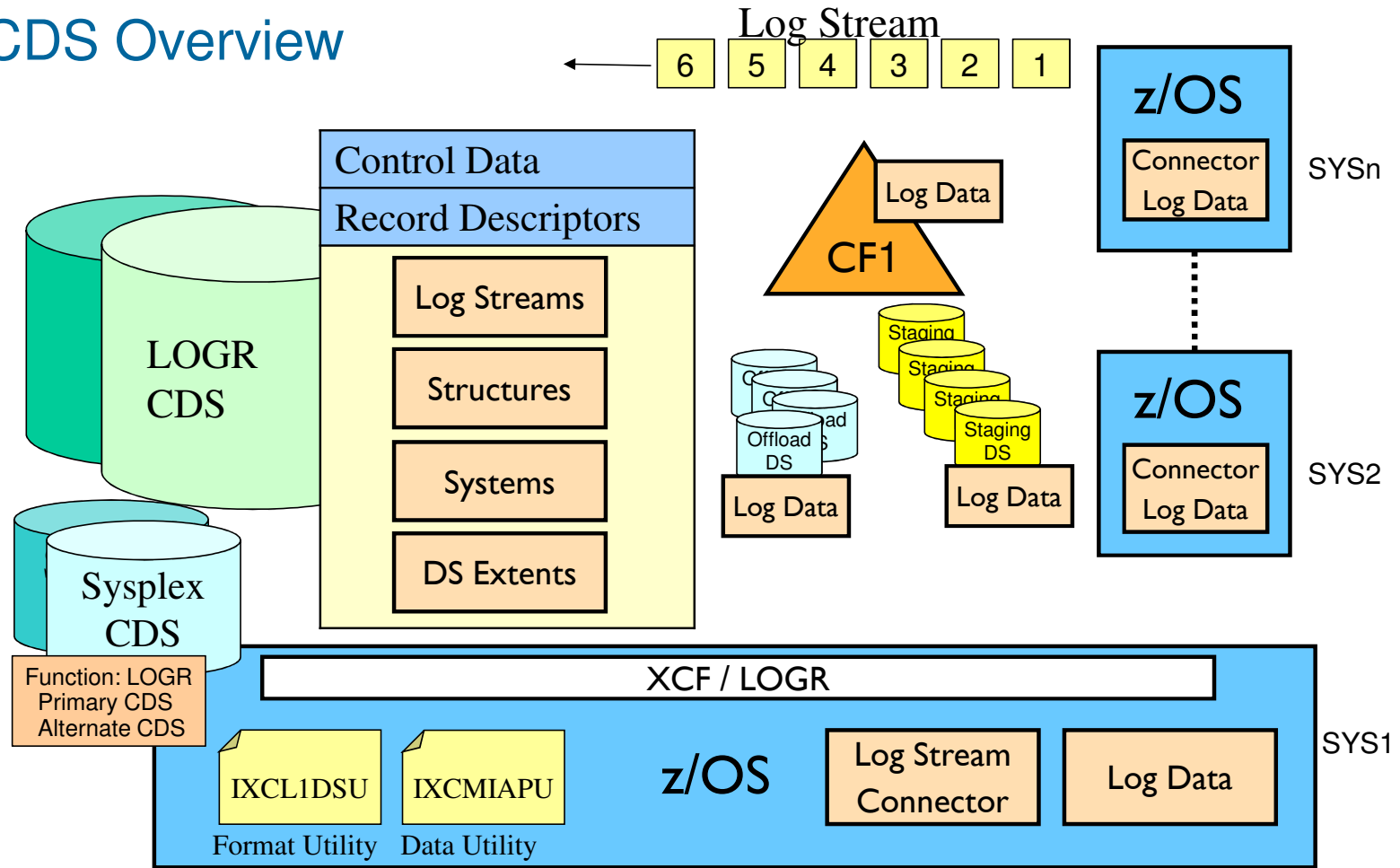
Service Policies for Sysplex Activate or view one of the service policies that is defined in the installed service definition.

Settings Set preferences for messages, service definition history, couple data set, and sysplex. Verify the code page and time zone settings before you start working.

View

WLM Status View the status of WLM on each system in the sysplex. View details about the installed service definition and the active service policy.

LOGR CDS Overview



System Logger

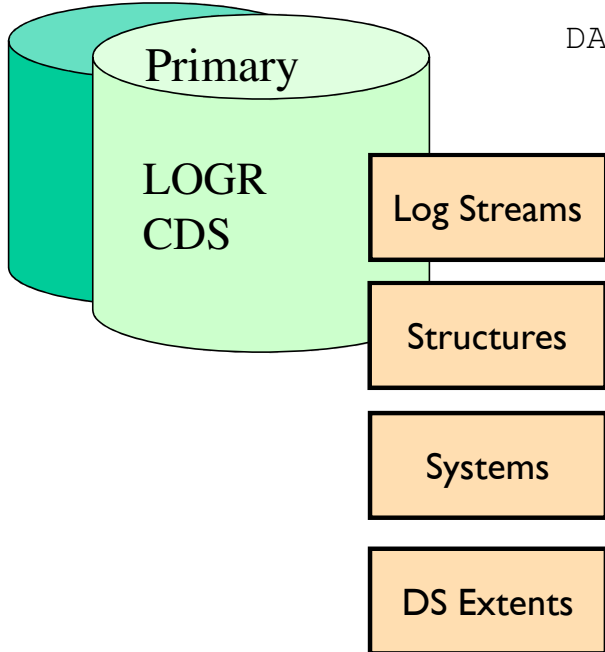
- Provides common set of logging services for applications to:
 - Save data
 - Archive data
 - Retrieve data
 - Delete expired data
- Enables multiple instances of an application scattered throughout the sysplex to have one common log

Logger Policy (or Logger Inventory)

- Defines
 - Characteristics of each log stream
 - Coupling facility structure to be used when there are active connections to the log stream
- Which in turn determines how Logger will manage the log stream
- Contains transient status information such as log stream connectors
- As well as persistent status information needed when applications need to recover from their logs
 - Such as data to construct the log stream from the various data sets in which the log data might reside

Couple Data Set Format Utility LOGR CDS

Alternate



DATA TYPE (LOGR)

ITEM NAME (LSR)	NUMBER (<i>max #logstreams in policy</i>)
ITEM NAME (LSTRR)	NUMBER (<i>max #logstream structures in policy</i>)
ITEM NAME (DSEXTENT)	NUMBER (<i>#additional directory extents</i>)
ITEM NAME (SMDUPLEX)	NUMBER (1)

- Number of logstreams ?
- Number of structures needed for CF logstreams ?
- Directory extents increase offload capacity
- **Specify SMDUPLEX** for latest level of functionality

Creating LOGR CDS Content

- You must

- Format the LOGR CDS
- Create one or more log stream definitions in the CDS

Nothing needs to
be explicitly started

- The LOGR CDS contains data from different sources

- Log stream definitions Static data set by Data Utility (you)
- CF Structures Static data set by Data Utility (you)
- Log stream status Dynamic data set by systems in the sysplex

- You influence the “active policy” every time the utility is run

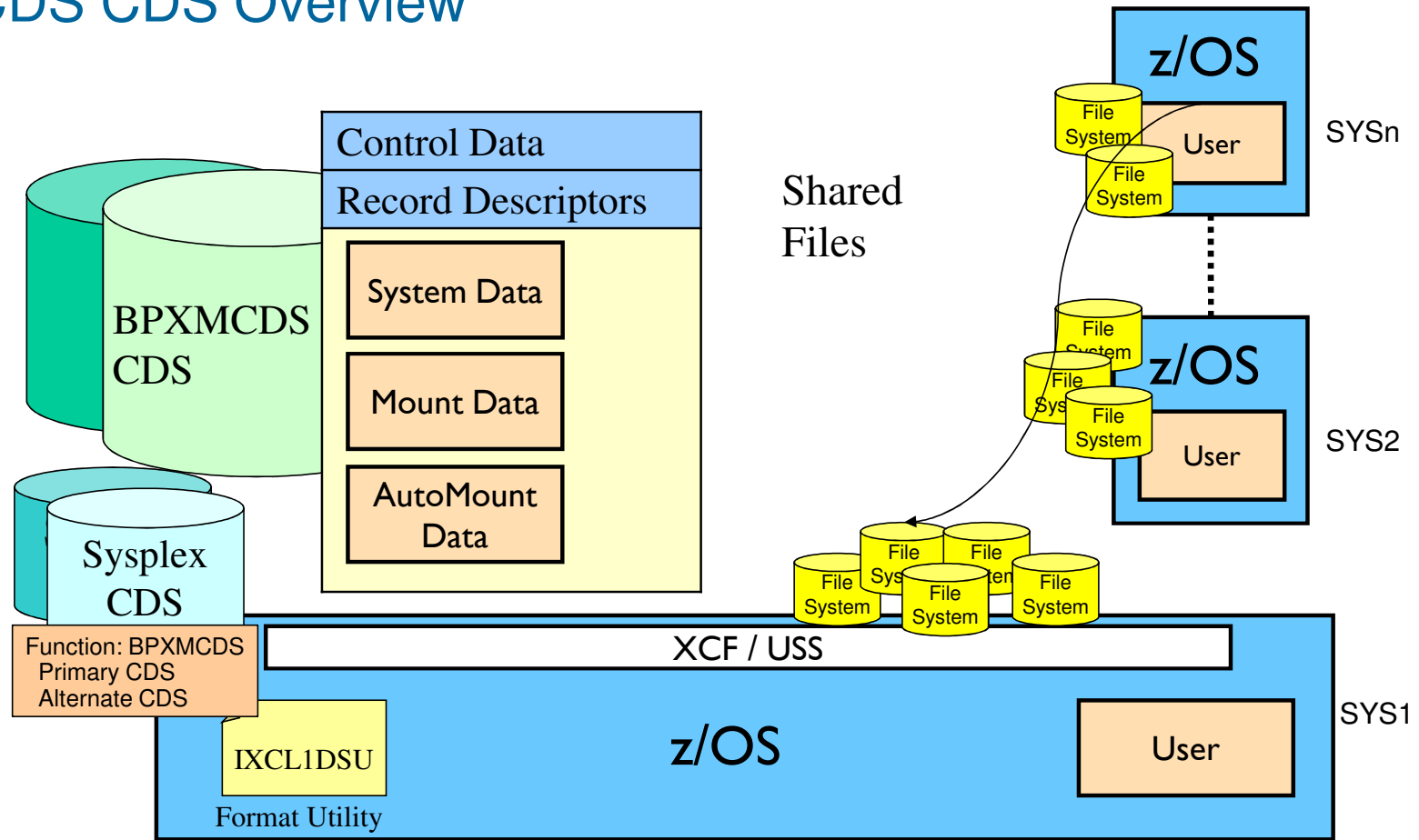
- You change the log stream definitions being used by the sysplex
- Some changes to a log stream will be rejected if the log stream is in use
- The accepted changes are immediately visible to the sysplex

Not at all like CFRM,
ARM, and SFM

Creating LOGR “Policies”

- The data utility program IXCMIAPU is used to manipulate log stream definitions in a formatted LOGR Couple Data Set
- The parameters you specify define
 - The log stream characteristics
 - The coupling facilities to be used
- Which in turn determines how Logger will manage the log stream
- The data utility can
 - Create new log stream definitions
 - Update log stream definitions
 - Delete log stream definitions
 - Report on the log streams currently defined in the CDS
- You might run the utility to manipulate an individual log stream
 - Perhaps a new application

BPXMCDS CDS Overview

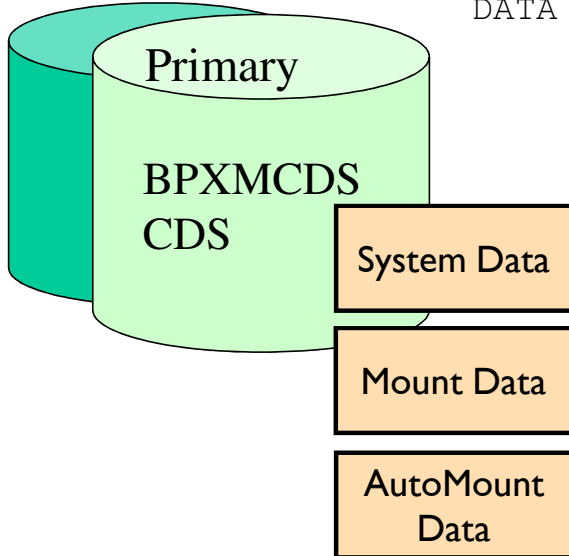


BPXMCDS Function Supports Shared File System

- Shared File System
 - A file mounted on one system can be accessed by a different system without the user needing to log onto the other system
- The BPXMCDS Function CDS enables this support by providing a central repository of information about which files are mounted to what system

Couple Data Set Format Utility BPXMCDS CDS

Alternate



DATA TYPE (BPXMCDS)

ITEM NAME (MOUNTS) NUMBER (*#mounts to support*)

ITEM NAME (AMTRULES) NUMBER (*#automount rules to support*)

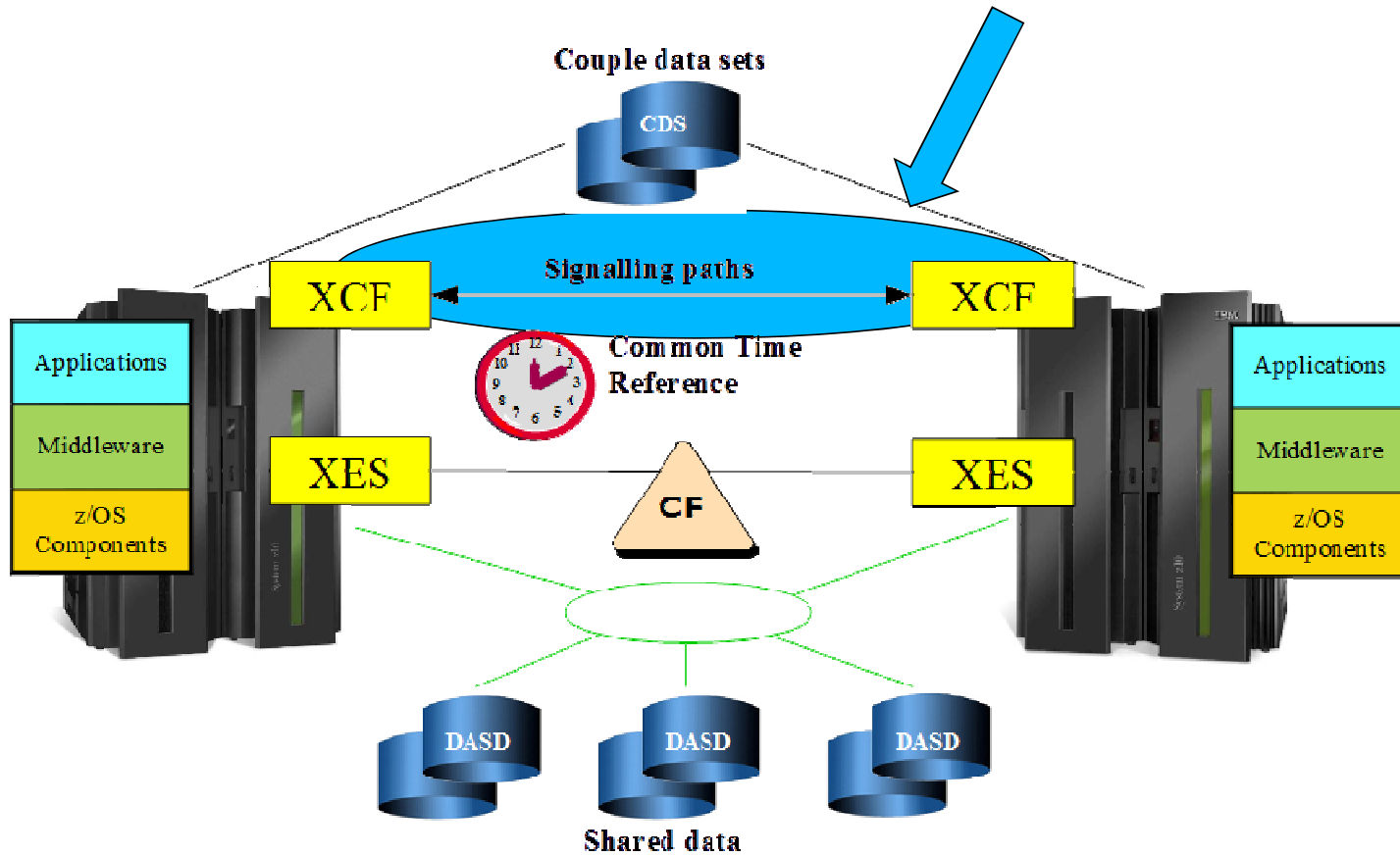
- MOUNTS to be supported, including concurrent automounts
- AMTRULES must accommodate number of automounts in your automount policy

See SYS1.SAMPLIB(BPXISCD S)

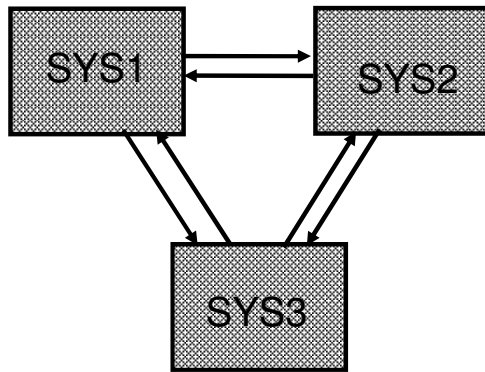
Creating BPXMCDS CDS Content

- Nothing for you to do beyond:
 - Formatting the CDS
 - Bringing CDS into service
 - Specifying SYSPLEX(YES) in BPXPRMxx parmlib member
 - System will be part of file sharing group
 - Other customization likely desirable as well
- All data is updated dynamically by those systems in the sysplex that participate in the file sharing group

Parallel Sysplex: XCF Signal Paths

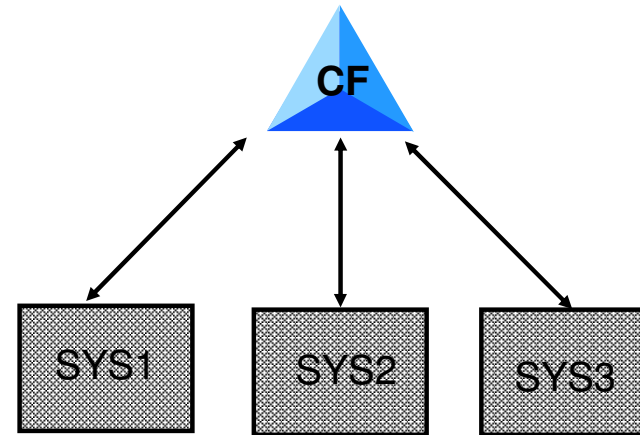


Signal Path Types



■ CTC Signalling Path

- ▶ Requires a device for each direction
- ▶ You define point to point connectivity
- ▶ Message stream
- ▶ Good for small messages
- ▶ Harder to define when lots of systems $O(n^2)$
- ▶ Drives SAP utilization



■ CF List Structure Signalling Path

- ▶ Can be used in both directions
- ▶ XCF self defines list paths for full connectivity
- ▶ Mail box approach
- ▶ Good for all message sizes
- ▶ Easier to define when lots of systems $O(n)$
- ▶ Increased SRB time in XCF address space
- ▶ Fastest data transfer rate

XCF Signalling Service

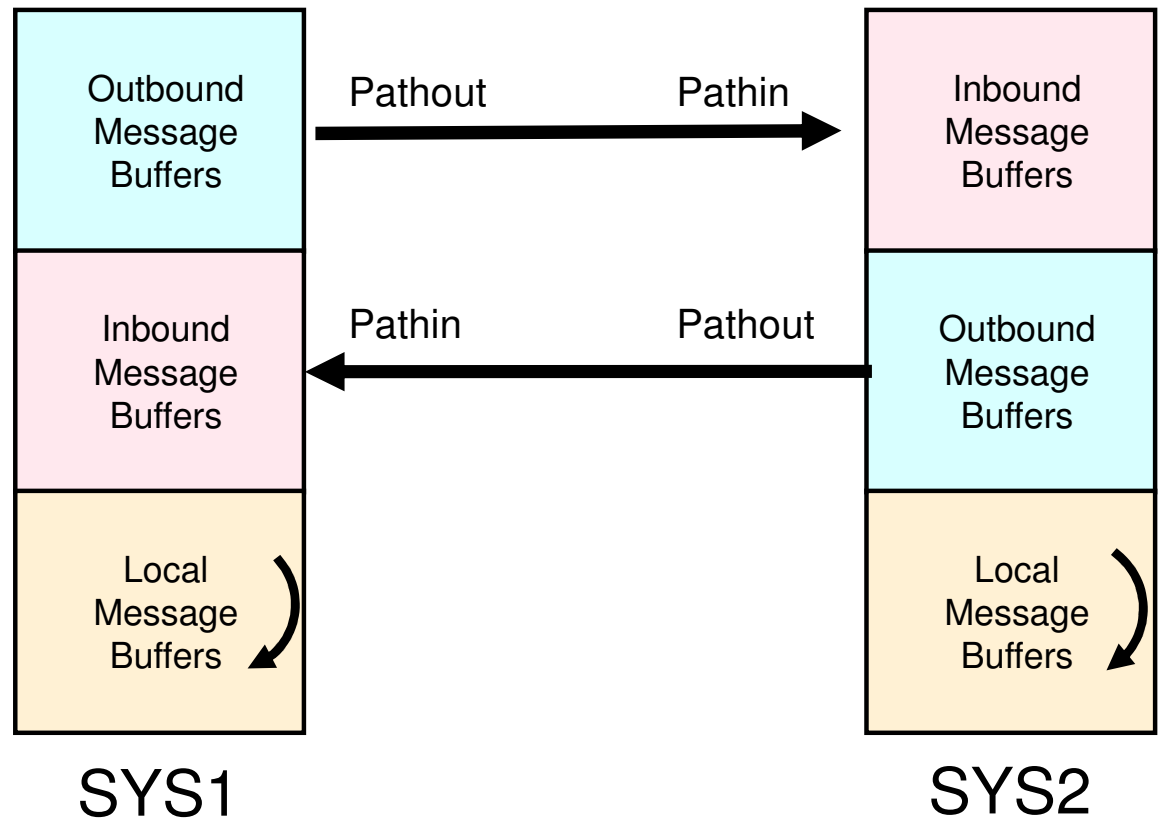
- Delivers messages to destination
- Manages physical resources used to transport messages

Configuring XCF Signal Service

- Sysplex requires any-to-any connectivity
- Signal paths are unidirectional
 - So require at least two paths, one in each direction between every pair of systems
 - More for redundancy
- Two types of signal path are supported
 - CTC (latest machines require FICON; older ones used ESCON, 3088)
 - Signal structure (coupling facility)
 - List path (point to point connection for one pair of systems)
- Transport Classes
 - Segregate resources, restrict XCF's choices
 - By size, group, or both
 - Consists of signal buffers and signal paths

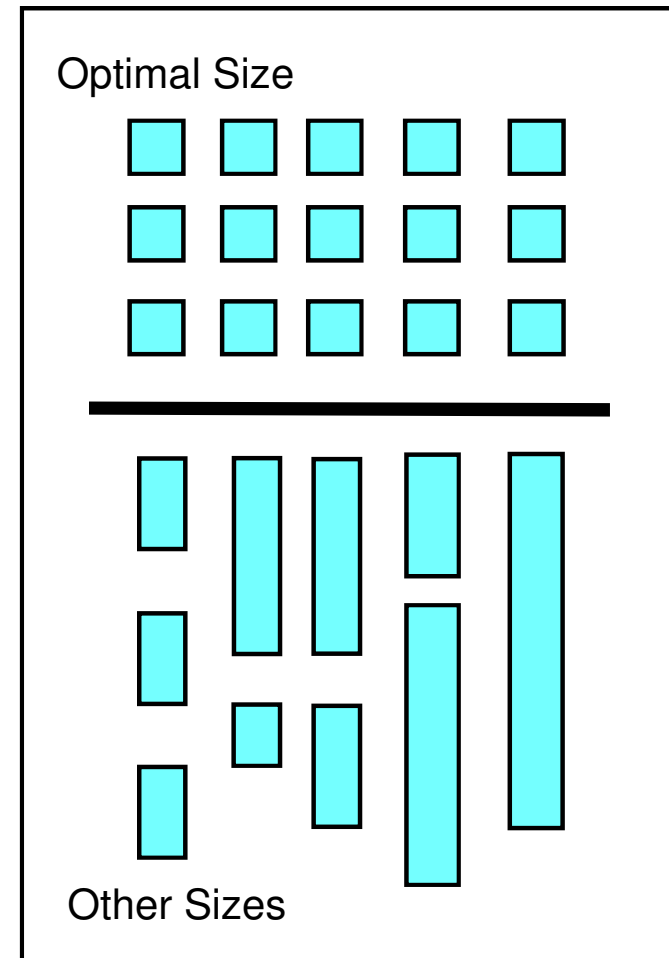
Signal Resources

- Signal Buffers
- Buffer Pools
- Signal Paths
 - Unidirectional
 - Buffer usage, types, protocols, capacity, performance
- Transport Classes
 - Segregate resources
 - Restrict XCF's choices

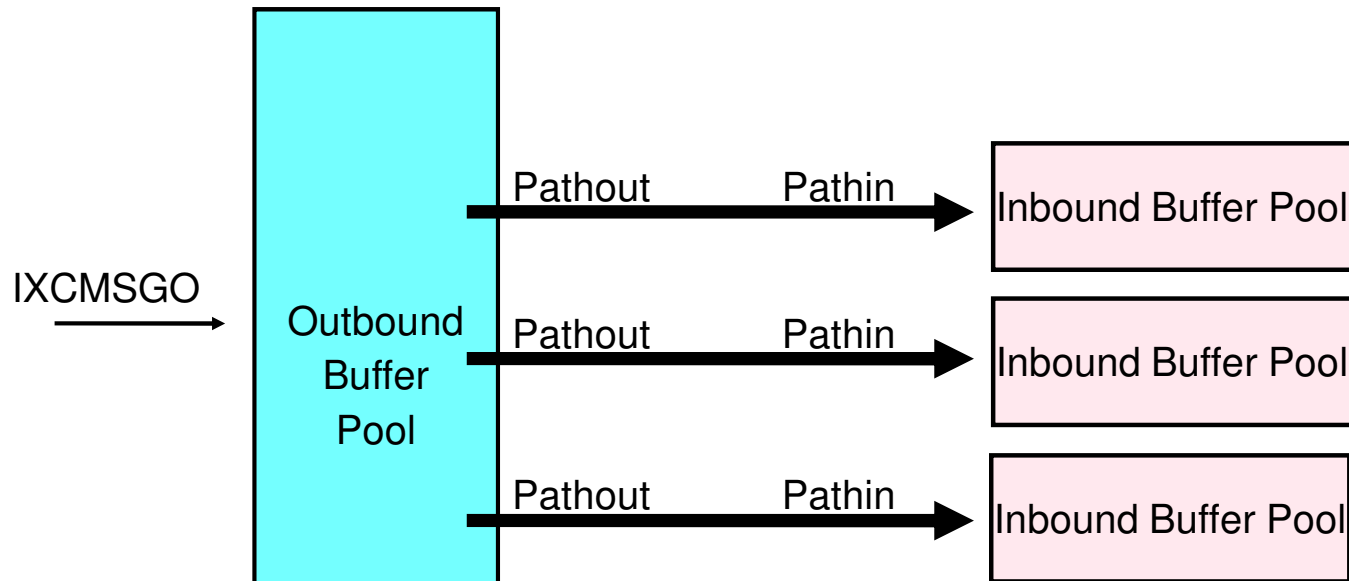


Buffer Pool

- Buffers used only as needed
- XCF returns unused buffers
- Premise: one optimal size
- Mixture of sizes possible
- Pseudo-best fit
- Space limits protect system

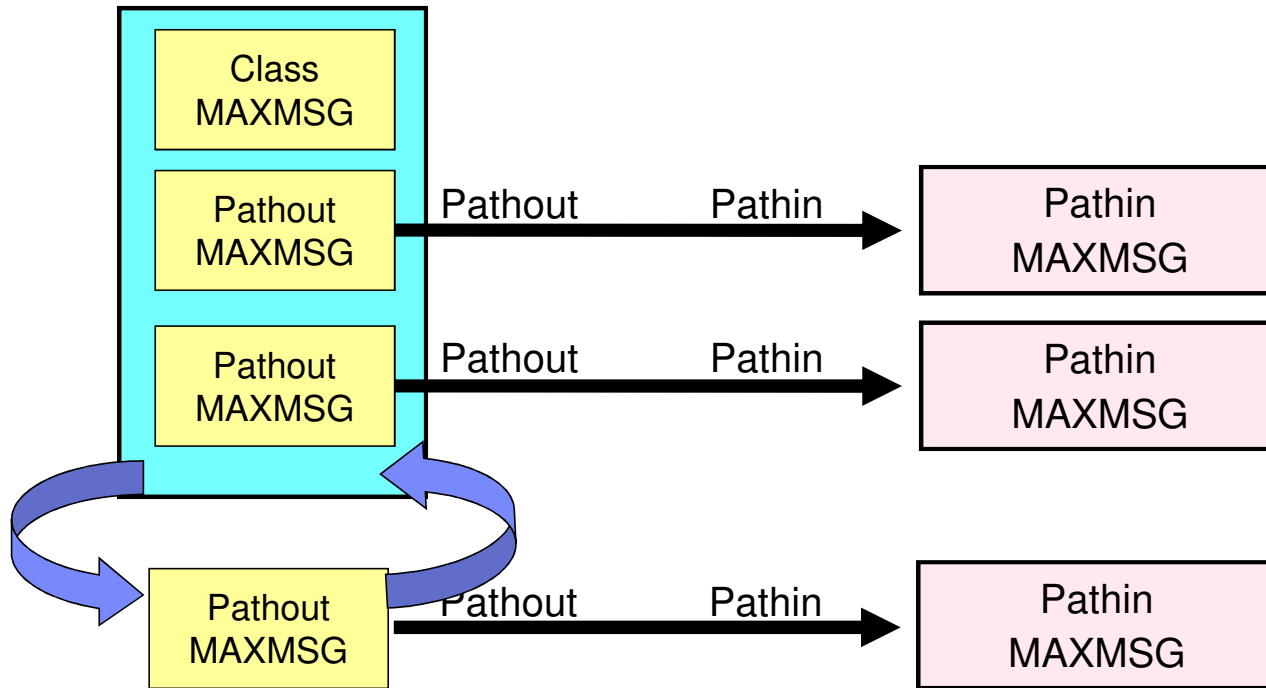


Signal Paths and Buffer Pools



- MAXMSG value(s) limit pool space
- Inbound pool dedicated to inbound path
- Outbound pool shared by outbound paths
- Pathout/Pathin agree on one buffer size

Buffer Pools and MAXMSG

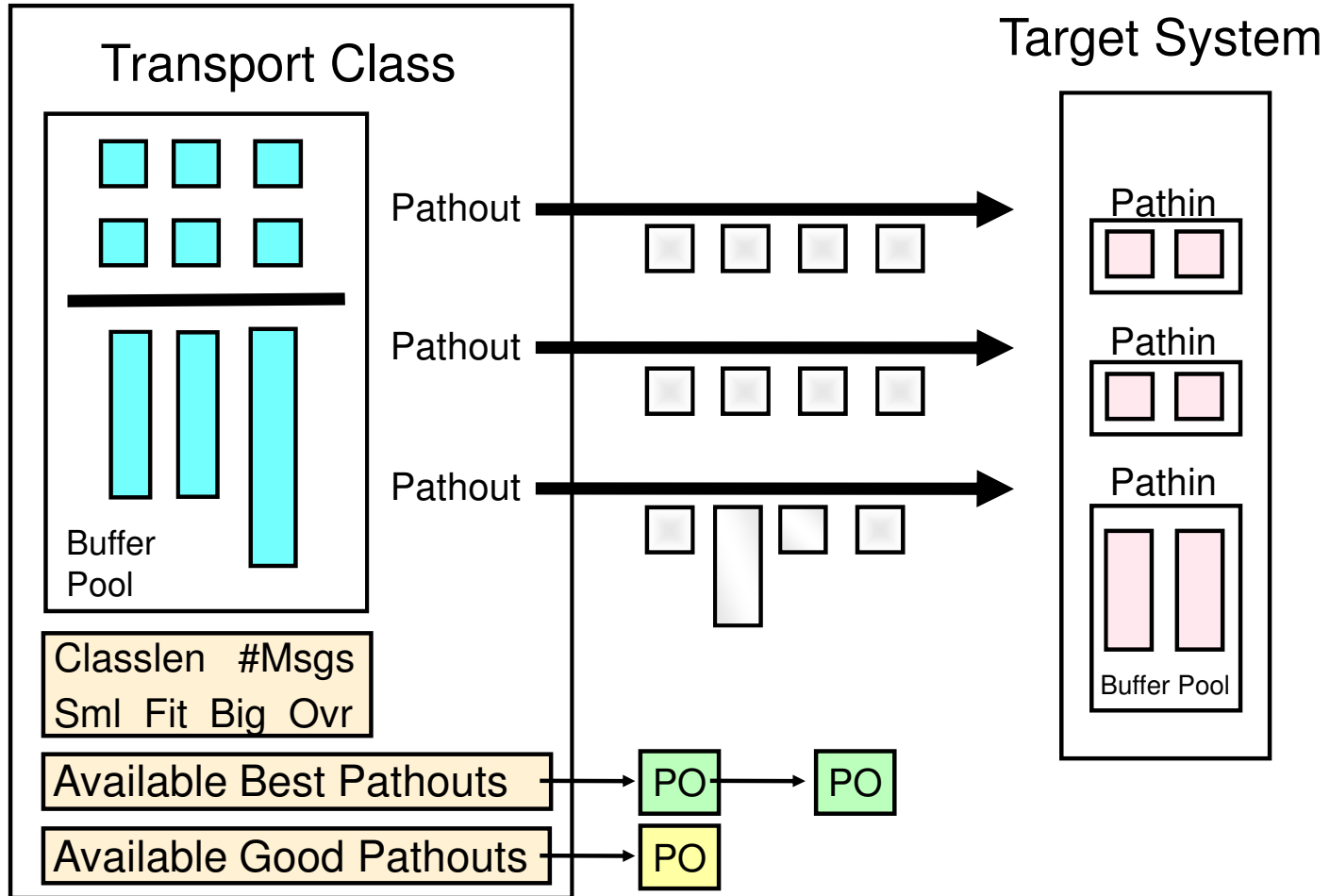


Amount of buffer space in outbound buffer pool adjusted when outbound path is established or removed from service

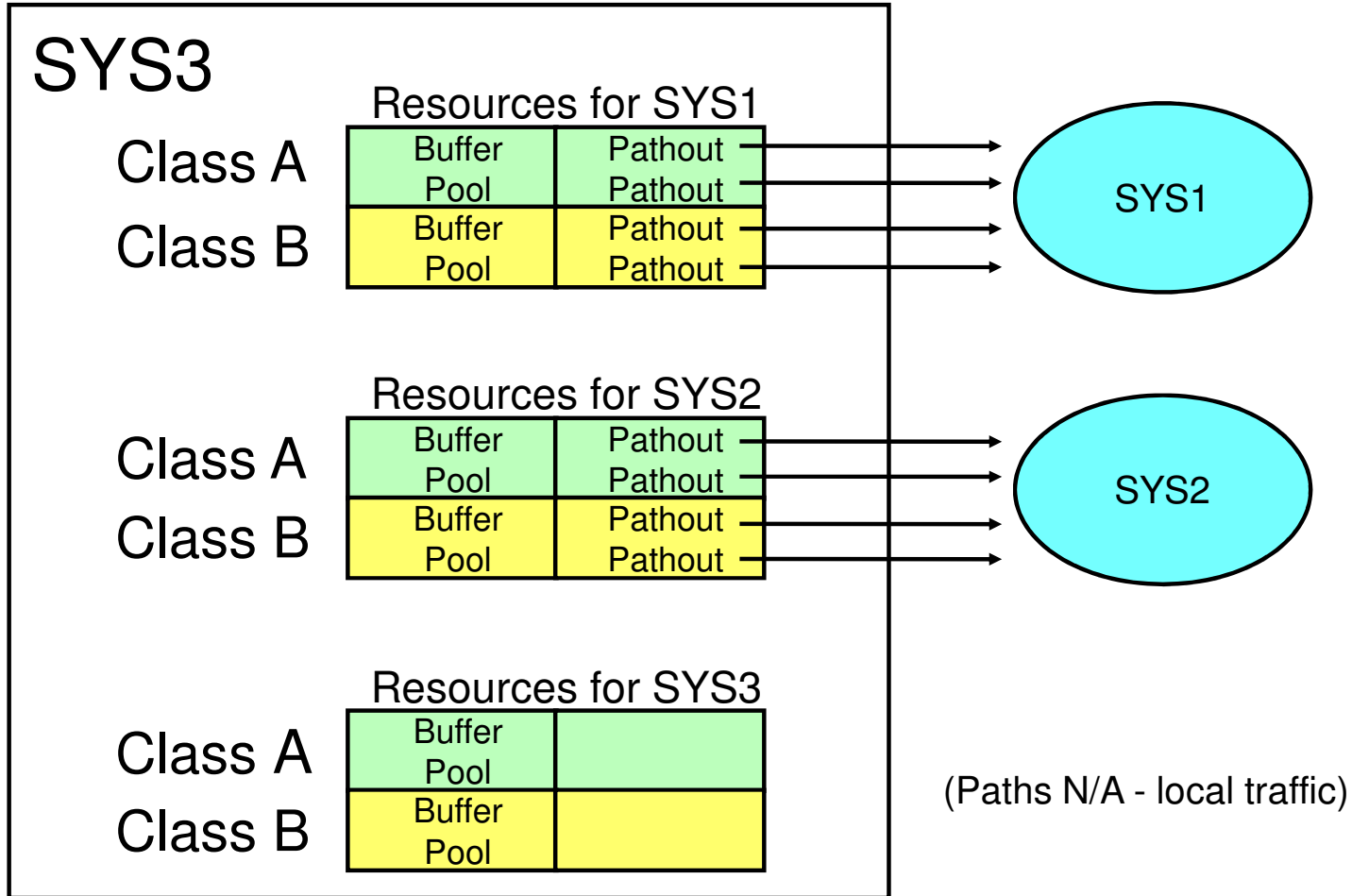
Transport Classes

- Named collection of signal resources (buffers and paths)
- One resource pool per system per classdef
 - ▶ Dedicated to particular target system
 - ▶ Individually monitored and tuned by XCF
- Segregates signals by size (and/or group)
 - ▶ Signal rejected if relevant class has no buffer
 - ▶ Lose segregation if pool has no path

Transport Class



Transport Classes

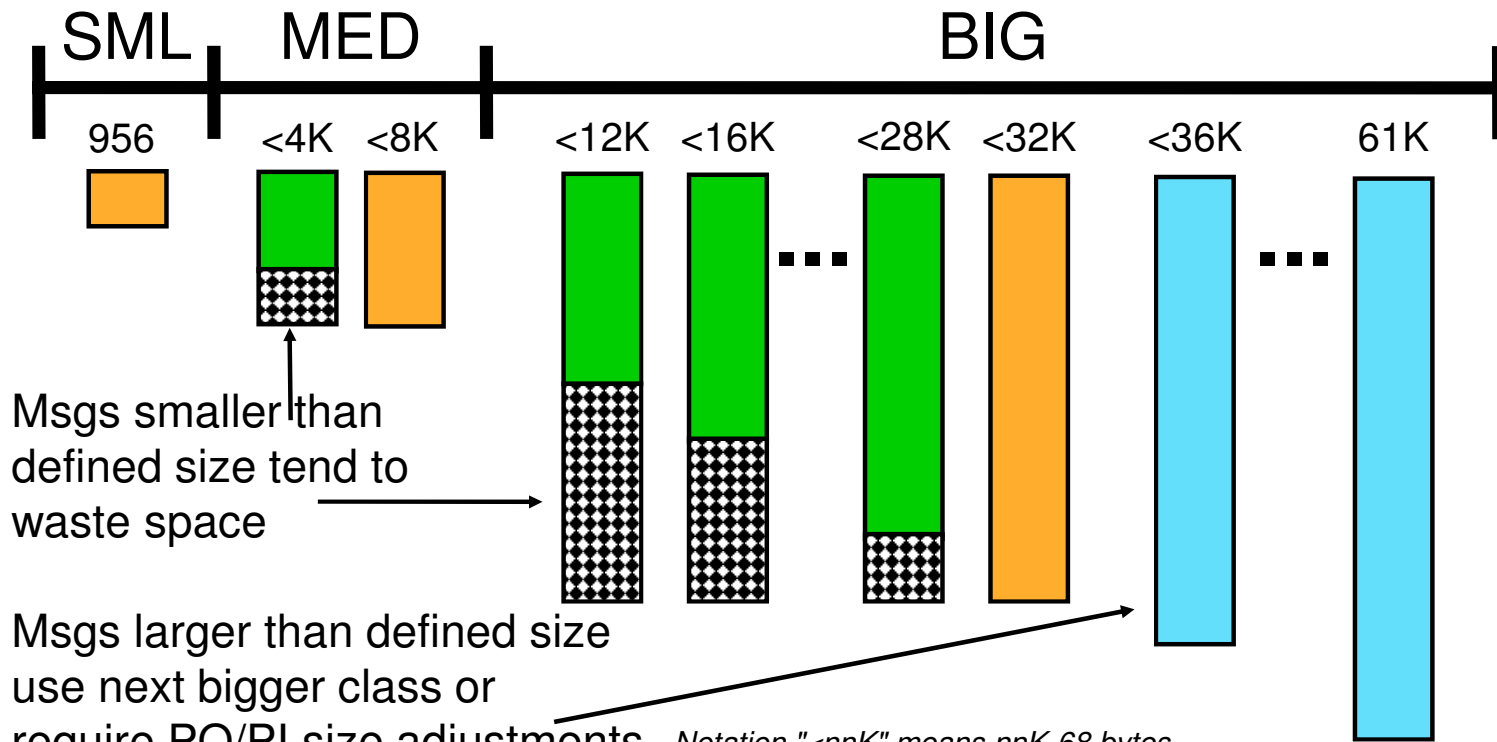


Classlen Segregation

```

CLASSDEF CLASS (SML) CLASSLEN (956)
CLASSDEF CLASS (MED) CLASSLEN (8000)
CLASSDEF CLASS (BIG) CLASSLEN (32000)
    
```

SML	=	0 956
MED	=	957 . . . 8124
BIG	=	8125 . . 62464



Msgs smaller than defined size tend to waste space

Msgs larger than defined size use next bigger class or require PO/PI size adjustments

Notation "<nnK" means nnK-68 bytes, indicates length of longest message that fits in buffer

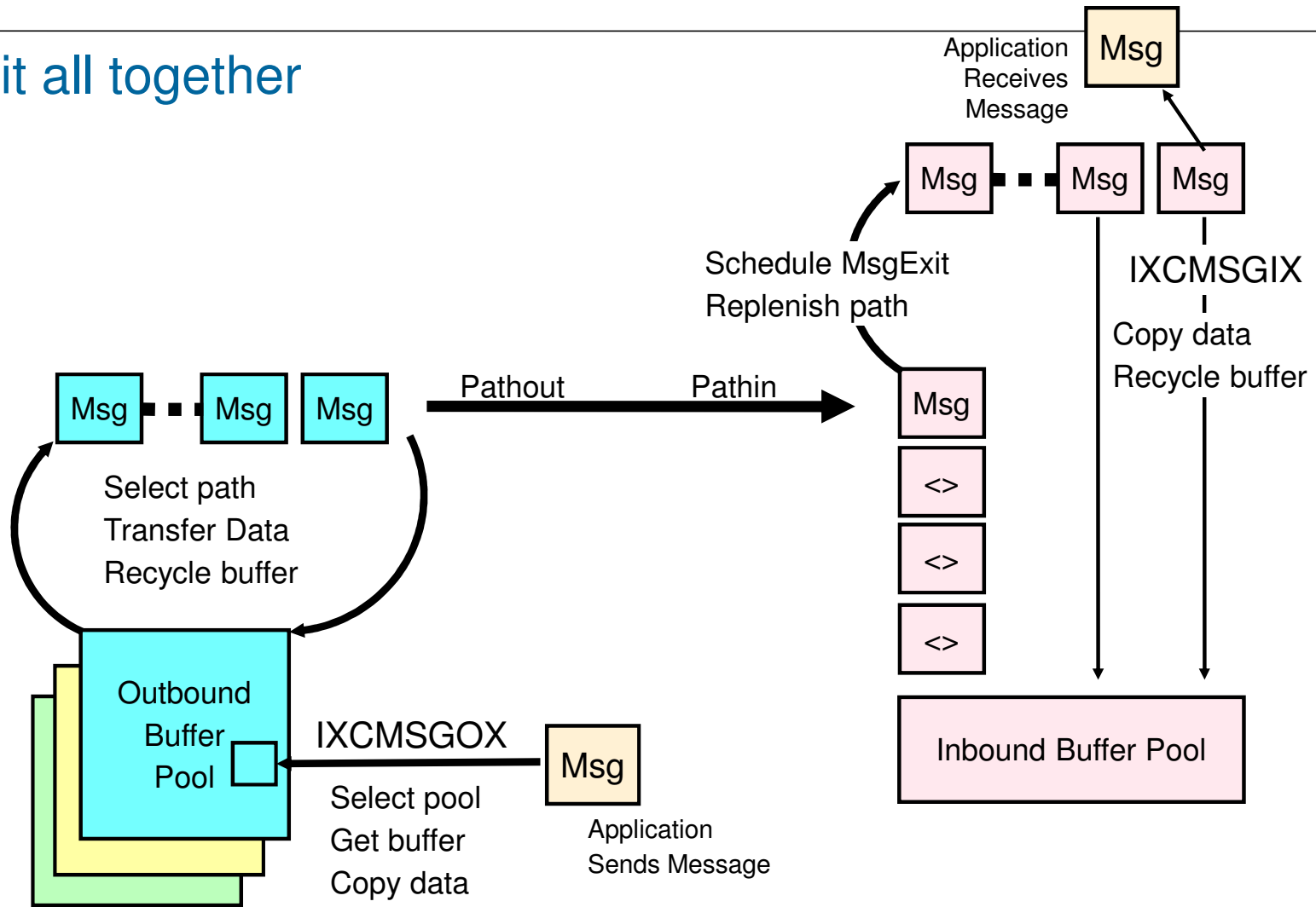
Transport Class Buffer Selection

- Determine set of classes that sender's group is allowed to use
- From that set, select those whose defined classlen provides the best fit
- From that set, get a buffer
 - Given a choice of classes
 - Classes with class connectivity are preferred
 - Next, classes whose tuned size is best fit are preferred

Transport Class Path Selection

- First, look for operational path in same class as buffer
- If none, choose the first set of classes that has class connectivity:
 - Classes that sender's group is allowed to use, else
 - Classes to which no group is explicitly assigned, else
 - All classes
- Then, look for operational path from one of those classes
 - Given a choice of classes
 - Try defined size (then smallest tuned size)
 - Try larger sizes (then smallest tuned size)
 - Try smaller sizes (then smallest bigger, then biggest smaller)

Putting it all together



COUPLExx – Signal Definitions

COUPLE

....

MAXMSG(nnnn)

CLASSDEF CLASS(name) CLASSLEN(nnnnn)

LOCALMSG MAXMSG(nnnn) CLASS(NAME)

PATHOUT DEVICE(nnnn) CLASS(name)

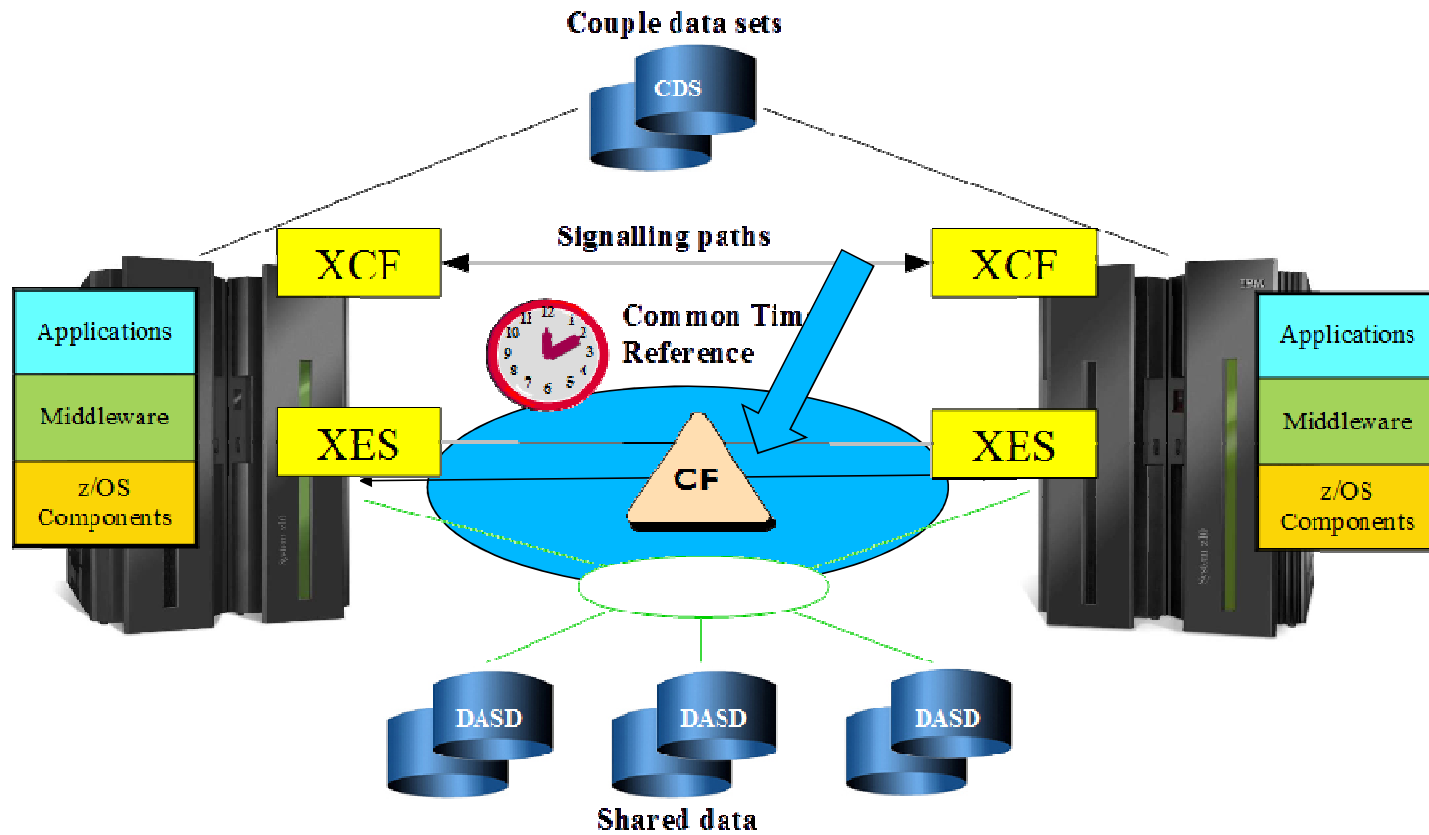
PATHIN DEVICE(nnnn)

PATHOUT STRNAME(IXC....) CLASS(name)

PATHIN STRNAME(IXC....)

*MAXMSG specification
for PATHIN might be the
most important thing to
worry about.*

Parallel Sysplex: Coupling Facility



Coupling Facility (CF)

- The CF is an LPAR that runs a special operating system, the Coupling Facility Control Code (CFCC)
 - The CFCC has “releases” which are referred to as CF Levels (CFLEVEL)
 - Each successive level generally provides one or more of the following:
 - A new feature of interest to some exploiter
 - Support for new hardware
 - Internal improvements (performance, RAS, capacity, ...)
 - In general, each successive release incorporates functionality of its predecessors
- The CFCC spins in an infinite loop, looking for work
 - Work arrives in the form of a request received via a CF link
 - Requests originate from z/OS, exploiters, even other CF's
 - Exploiters use API's provided by z/OS to send their requests to “the” CF

Coupling Facility

- The CF is a pure memory system
 - The amount of memory is determined by its LPAR configuration
 - The CF does not do paging
 - Indeed, the CF does not do traditional I/O and thus cannot access disk at all
 - CFLEVEL 19 support for flash memory might seem like a form of paging
- CF Memory is organized into “structures”
 - The active CFRM policy determines how much CF memory is assigned to a given structure
 - Three types of structures supported: list model, lock model, and cache model
 - Each model has its own set of high level operations
 - So a “request” manipulates memory in the CF via an operation that is appropriate to the model

Coupling Facility

- Requests can be processed synchronously or asynchronously
 - Synchronous: send request to CF then “wait” for results to be returned
 - The concern here is opportunity cost for the z/OS image
 - XES may decide that it is more cost effective to process the request asynchronously
 - Asynchronous: send request to CF, then do other work while the CF processes the request. z/OS will eventually observe that the request has completed.
 - The concern here is application response time
 - How quickly can z/OS observe completion and get the results to the application
 - This is really a function of z/OS interaction with the CF link as opposed to the CF itself
- XES heuristic algorithm may convert a synch request to async in an attempt to maximize work on the z/OS image
 - As of z/OS V2R1 you can choose the conversion thresholds (most should not do so)

Use thin interrupts on z/OS side for better async response times

Coupling facility configuration

- Use the HMC to configure the LPAR
 - HMC can also issue commands to the CF
- CF Considerations
 - Placement relative to z/OS images with respect to recovery
 - Memory, white space
 - CPU capacity, white space
 - Dedicated vs shared engines
 - Dynamic dispatch mode, Thin interrupts
 - No more than one generation behind
 - In fact, often first to be upgraded to new hardware
- Keep up to date with MCL changes
- Use latest links

Coupling Facility Management

- CFRM CDS
- CFRM Policy
 - Coupling Facility Definitions
 - Structure Definitions
 - CF Sizer

References

- White Paper: Coupling Facility Configuration Options
 - public.dhe.ibm.com/common/ssi/ecm/en/zsw01971usen/ZSW01971USEN.PDF
- White Paper: Coupling Thin Interrupts and Coupling Facility Performance in Shared Process Environments
 - www.ibm.com/support/techdocs/atmastr.nsf/WebIndex/WP102400
- Processor Resource/Systems Manager Planning Guide (SB10-7156)
 - Available on Resource Link
- www.redbooks.ibm.com/Redbooks.nsf/RedbookAbstracts/tips0237.html
 - Written in 2003, so no mention of thin interrupts
 - But nicely explains considerations for DYNDISP=ON or OFF

Things we didn't cover that you need to learn about

- Maintenance
 - CF maintenance mode and REALLOCATE
 - Rolling IPLs
- Failure recovery
- Operations, operator commands
- Critical messages, automation
- RMF reports
- Health Checks
- Configuring other sysplex enabled components of z/OS (GRS, Consoles, RACF...)
- Middleware exploitation and configuration (DB2, CICS, IMS, MQ, ...)
- Application enablement and governance
 - Customer applications are often the biggest impediment to achieving high availability

Summary

- Understand the business requirements
- Build a sysplex to meet those requirements
 - Much flexibility, which can lead to complexity
- z/OS and CF configuration should be dictated by application requirements
 - Number of systems in sysplex
 - Placement of z/OS relative to CF
 - CF configuration
- Sysplex Infrastructure
 - Server Time Protocol
 - Couple Data Sets
 - XCF signal service

For More Information

■ z/OS Publications

- z/OS MVS Setting Up A Sysplex
- z/OS UNIX System Services Planning
- z/OS MVS Planning: Global Resource Serialization
- z/OS MVS Planning: Workload Management

■ Redbooks

- System z Parallel Sysplex Best Practices
- System z Programmer's Guide to: z/OS System Logger

■ Learn from the misfortune of others

- WP102281 Couple Data Sets: Best Practices and Avoiding Disasters
- FLASH10786: Where is my coupling facility?
- Hot Topics Article: Mirror, mirror on the wall, should these data sets be mirrored at all?

Please complete your session evaluation

Sysplex 101: Setting up a Sysplex
Session 17914

Complete your session evaluations online at www.SHARE.org/Orlando-Eval