#SHAREorg

System Logger Top 10 Problems

Nicholas R. Jones
IBM System Logger

Thursday, March 15, 2012: 3:00- 4:00PM
Session Number 10849

This presentation is for educational purposes only.
It is not a deliverable to customers.

© 2012 IBM Corporation

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.

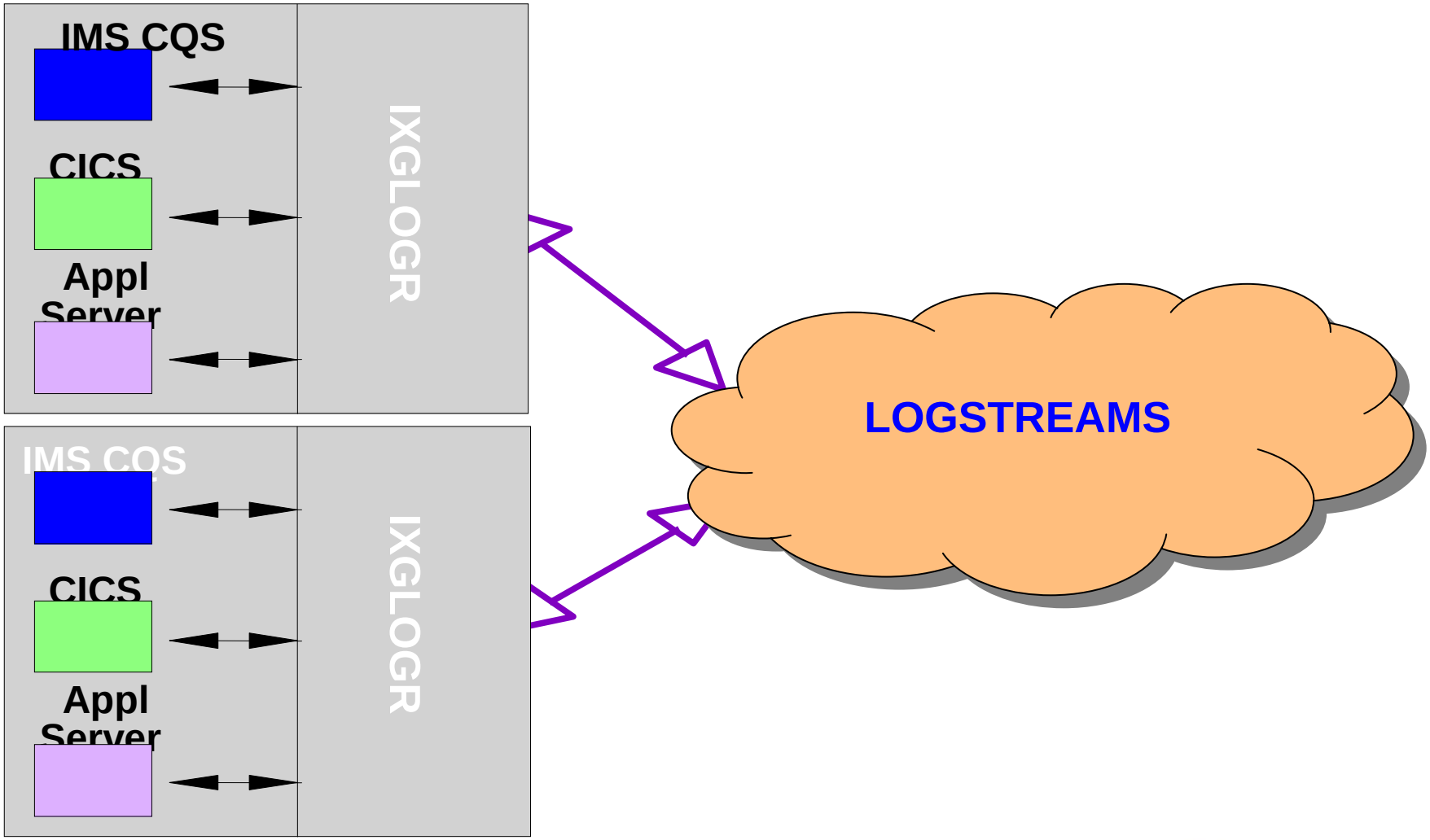
© 2012 IBM Corporation

System Logger Top 10 Problems

- 10) LOGR couple data set format level
- 9) Collecting doc after a logger failure
- 8) SM Duplex setup
- 7) Incorrect LOGR CDS specified at IPL
- 6) Log data retention
- 5) Manually deleting logger data sets
- 4) WTOR response behavior IXG115A, IXG272E, IXG312E
- 3) Structure sizing
- 2) Staging data set sizing
- 1) Offload data set sizing

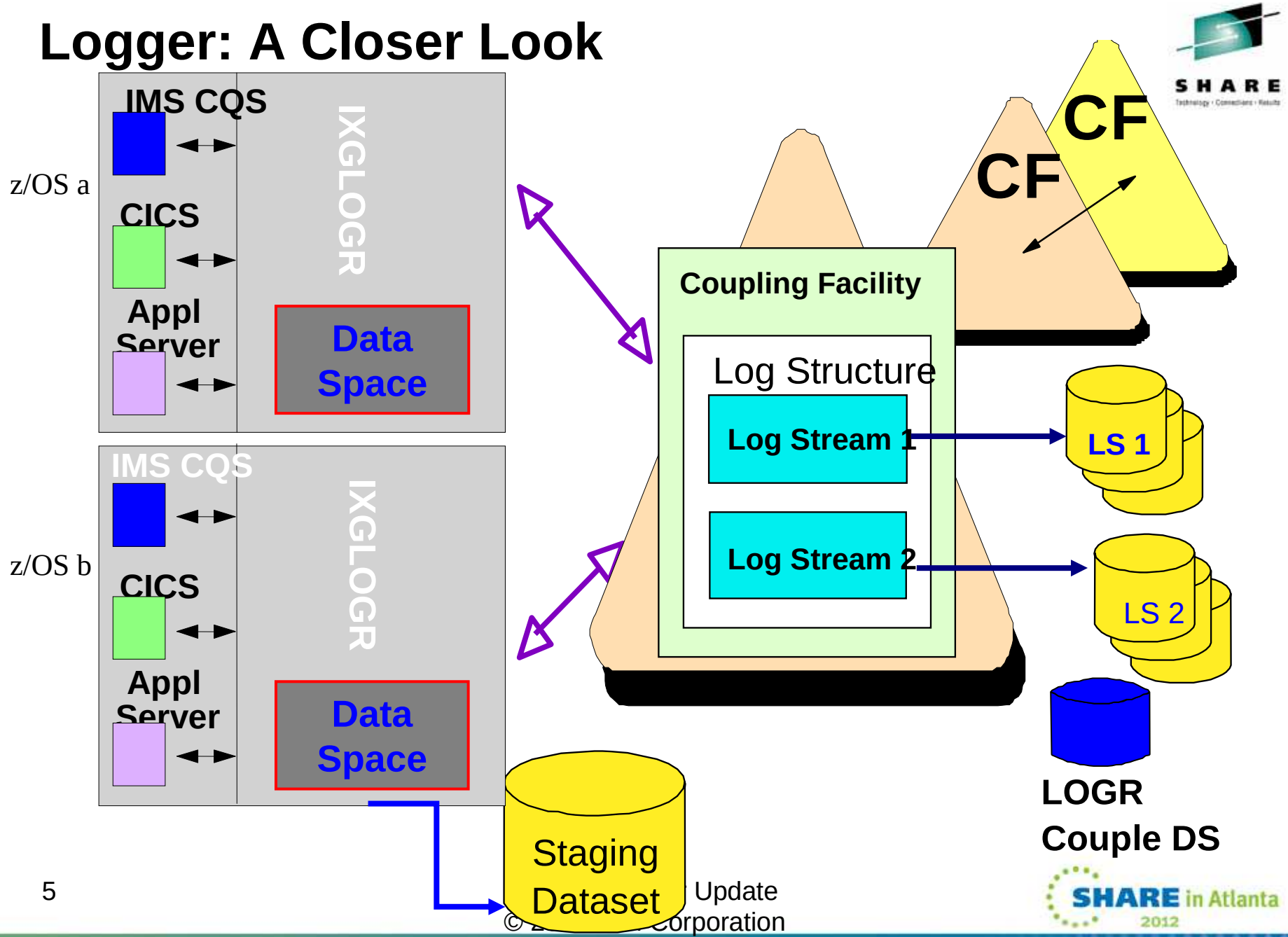
© 2012 IBM Corporation

Logger: 50,000 Foot View



© 2012 IBM Corporation

Logger: A Closer Look



Update
© Corporation

10

LOGR Couple Data Set Format Level

Clients with old LOGR CDS format levels are missing out on new enhancements.

Find out what they are and how to upgrade your CDS.

© 2012 IBM Corporation

10) LOGR Couple Data Set Format Level

What is the LOGR CDS?

- The LOGR couple data set holds system logger information about
  - Log streams
  - Log stream structures
  - Data set extents
  - Control information
- Uses XCF services
  - Formatted using IXCL1DSU utility
  - Activated using `SETXCF COUPLE,TYPE=LOGR`
- Three format levels

© 2012 IBM Corporation

10) LOGR Couple Data Set Format Level
CDS format levels

- HBB5520 (MVS/ESA SP5.2.0, 1995)
  - Initial level with LSR and LSTRR records
- HBB6603 (OS/390 V1R3, 1997)
  - Added DSEXTENT records
  - Added to LSR records
    - RETPD
    - AUTODELETE
    - USERDATA
- HBB7705 (z/OS V1R2, 2001)
  - **Must specify SMDUPLEX(1)**
  - Added to LSR records + reserved space…

© 2012 IBM Corporation

10) LOGR Couple Data Set Format Level
HBB7705 (z/OS V1R2) format level

- With HBB7705 (z/OS V1R2) CDS format level you get
  - CF system managed structure duplexing
  - OFFLOADRECALL – configure data set recalls during offload
  - EHLQ – extended high level qualifier
  - Pending keyword updates
  - Rename log stream (NEWSTREAMNAME)
  - GROUP(PRODUCTION | TEST)
  - **Dynamic duplexing updates**
    - applying STG_DUPLEX, DUPLEXMODE, and LOGGERDUPLEX keyword updates after a structure rebuild

© 2012 IBM Corporation

10) LOGR Couple Data Set Format Level
Updating your format level

- *make sure every system in your sysplex is at z/OS V1R2 or above*
- A) Check current record numbers

```
SY1  D XCF,COUPLE,TYPE=LOGR
SY1  IXC358I  10.51.00  DISPLAY XCF 539
LOGR COUPLE DATA SETS
PRIMARY    DSN: LOGGER.HBB6603.CDS
           VOLSER: LOGR2      DEVN: 0274
           FORMAT TOD         MAXSYSTEM
           09/19/2000 15:51:09        8
           ADDITIONAL INFORMATION:
            LOGR COUPLE DATA SET FORMAT LEVEL: HBB6603
             LSR(25) LSTRR(25) DSEXTENT(15)
             SMDUPLEX(0)
LOGR IN USE BY ALL SYSTEMS
```


© 2012 IBM Corporation

10) LOGR Couple Data Set Format Level
Updating your format level

- *Your new CDS must have record numbers greater or equal to the current CDS*
- B) Format a new LOGR CDS

```
//FORMAT JOB  MSGLEVEL=(1,1)
//         EXEC PGM=IXCL1DSU
//SYSPRINT DD   SYSOUT=*
//SYSIN    DD   *
 DEFINEDS SYSPLEX(PLEX1) DSN(LOGGER.HBB7705.CDS) VOLSER(LOGR3)
   DATA TYPE(LOGR)
     ITEM NAME(LSR)       NUMBER(25)
     ITEM NAME(LSTRR)     NUMBER(25)
     ITEM NAME(DSEXTENT) NUMBER(15)
     ITEM NAME(SMDUPLEX) NUMBER(1)
```

Make sure you have data set extents

Must Be set to 1

© 2012 IBM Corporation

10) LOGR Couple Data Set Format Level
Updating your format level

- C) Activate the new CDS

```
SY1  SETXCF COUPLE,ACOUPLE=(LOGGER.HBB7705.CDS,LOGR3),TYPE=LOGR
SY1  IXC309I SETXCF COUPLE,ACOUPLE REQUEST FOR LOGR WAS ACCEPTED
SY1  IXC260I ALTERNATE COUPLE DATA SET REQUEST FROM SYSTEM
SY1 FOR LOGR IS NOW BEING PROCESSED.
DATA SET:  LOGGER.HBB7705.CDS
SY1  IXC251I NEW ALTERNATE DATA SET
LOGGER.HBB7705.CDS
FOR LOGR HAS BEEN MADE AVAILABLE
```

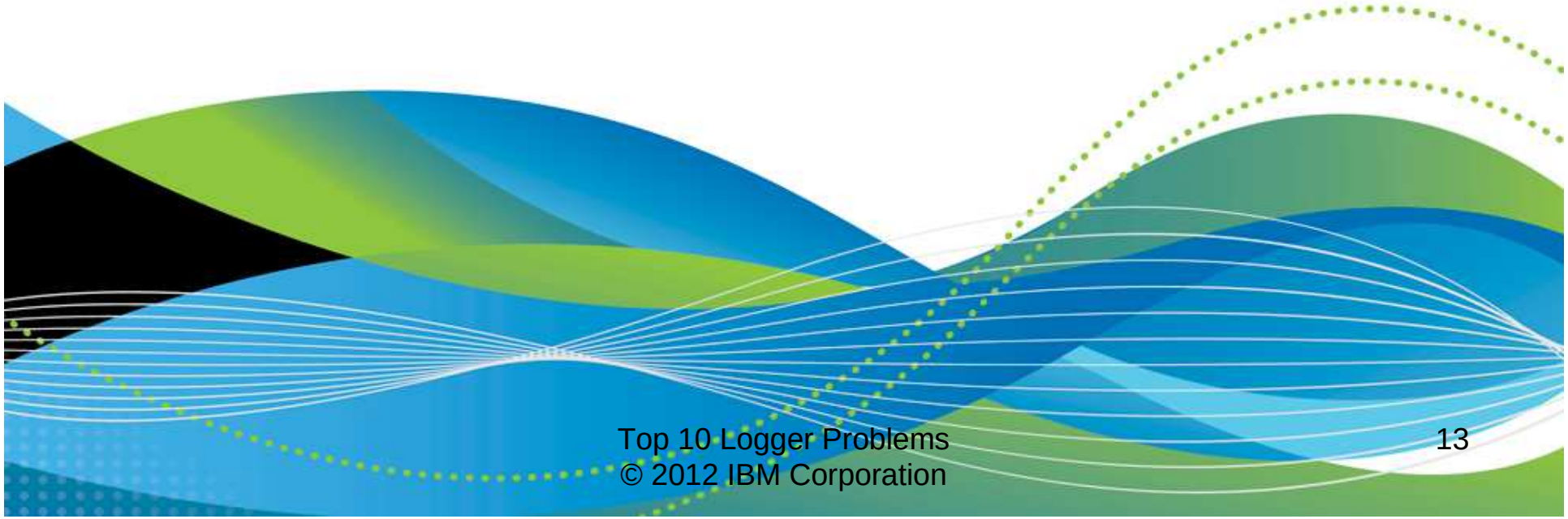
```
SY1  SETXCF COUPLE,TYPE=LOGR,PSWITCH
SY1  IXC309I SETXCF COUPLE,PSWITCH REQUEST FOR LOGR WAS ACCEPTED
SY1  IXC257I PRIMARY COUPLE DATA SET
LOGGER.HBB6603.CDS FOR LOGR
IS BEING REPLACED BY
LOGGER.HBB7705.CDS DUE TO OPERATOR REQUEST
SY1  IXC263I REMOVAL OF THE PRIMARY COUPLE DATA SET
LOGGER.HBB6603.CDS FOR LOGR IS COMPLETE
```


© 2012 IBM Corporation

9

Collecting Doc After A Logger Failure

Knowing which doc to collect after a logger failure can go a long way to assist IBM in debugging logger problems



© 2012 IBM Corporation

9) Collecting doc after a logger failure
Logger ctrace

- CTnLOGxx parmlib member contain logger trace setup information.
- CTILOG00 shipped in parmlib with defaults
- ```
  TRACEOPTS ON
  BUFSIZE(16M)
  OPTIONS('CONNECT,LOGSTRM,DATASET,SERIAL,MISC,LOCBUFF,RECOVERY')
  ```
- Use `D TRACE,COMP=SYSLOGR` to see your current options
- When zeroing in on a problem consider
  - `INVENTRY`
  - `STORAGE` – very chatty
  - `STRMNAME=(`*`ls1,ls2`*`)`…
  - Increasing the `BUFSIZE` as needed
- CTnLOGxx specifiable by IXGCNFxx parmlib in 1.13

© 2012 IBM Corporation
  ```

9) Collecting doc after a logger failure
Dumping logger

- See SYS1.SAMPLIB members **IEADMCLS,** IEADMCLG, IEADMCLX
- Consider Dumping
  - Address spaces **and data spaces**
    - IXGLOGR
    - XCFAS
    - GRS
    - RASP (RSM)
    - SMS
    - CATALOG
    - DFHSM
  - Remote systems
  - STRLIST – CF structure data

© 2012 IBM Corporation

9) Collecting doc after a logger failure
Other doc

- DISPLAY LOGGER output
- IXCMIAPU utility output
  - ```
    DATA TYPE(LOGR) REPORT(YES)
    LIST LOGSTREAM NAME(CICSA.TEST.DFHLOG) DETAIL(YES) LISTCAT
    LIST STRUCTURE NAME(DFHLOG_CICSA) DETAIL(YES)
    LIST LOGSTREAM NAME(SYSPLEX.*) DETAIL(YES)
    LIST STRUCTURE NAME(OPER*) DETAIL(YES)
    ```
- ADRDSSU utility to print a data set
  - LOGR couple
  - Offload
  - Staging

© 2012 IBM Corporation
    ```

9) Collecting doc after a logger failure

Other displays

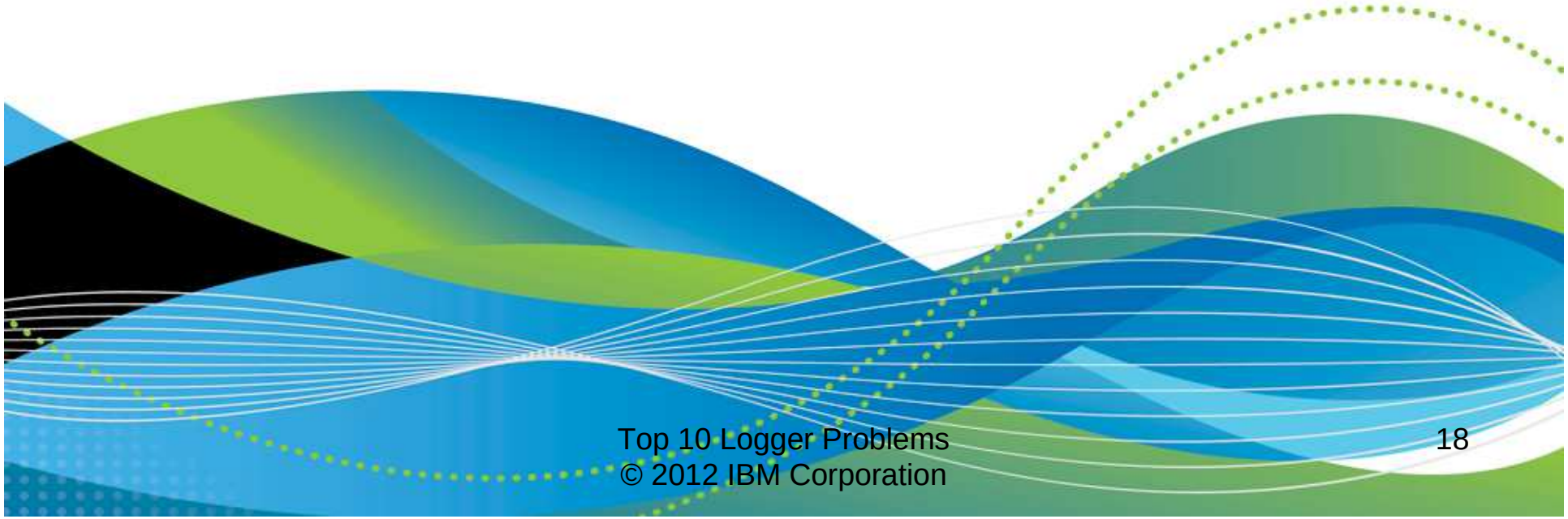
- `D GRS,Latch,Jobname=IXGLOGR`
- **`D GRS,ANALYZE,LATCH,DEPENDENCY,JOBNAME=IXGLOGR,DETAIL`**
- `D GRS,RES=(SYSZLOGR,*)`
- `D GRS,C`
- `D GRS,RNL=A`
- `D XCF,STR,STRNAME=`*`logger_structure`*
- `D XCF,COUPLE,TYPE=LOGR`

© 2012 IBM Corporation

8

System Managed Duplexing

Setup of system managed structure duplexing has numerous requirements and configuration choices.



© 2012 IBM Corporation

8) System Managed Structure Duplexing
Requirements

- At least two coupling facilities
  - Can be ICF
- Links between the CFs as well as links between each using systems and both CFs
- CFRM CDS with `SMDUPLEX(1)` and `SMREBLD(1)`
- CFRM structures with
  - `ALLOWAUTOALT(YES)`
  - `DUPLEX(ENABLED/ALLOWED)`
- LOGR CDS with `SMDUPLEX(1)`

© 2012 IBM Corporation

8) System Managed Structure Duplexing
Decision points

- Logger Setup
  - `STG_DUPLEX(YES|NO)`
    - *Do I want staging data sets?*
  - `DUPLEXMODE(COND|UNCOND)`
    - *Do I always want staging data sets?*
  - `LOGGERDUPLEX(COND|UNCOND)`
    - *Do I always want logger managed duplexing?*
- *What is the failure dependency?*
  - Between the system and the primary CF
  - Between the system and the secondary CF
  - Between the two CFs
- *What is the volatility of each CF?*

© 2012 IBM Corporation

8) System Managed Structure Duplexing
Cases

Case	Duplex?	Volatile?	Failure dependant – SYS - CF	Failure dependant – CF - CF
1	Simplex	Volatile or dependant		*n/a*
2	Simplex	Non-volatile	independent	*n/a*
3	Duplex	Volatile or dependant		any
4	Duplex	Non-volatile	independent	dependent
5	Duplex	Non-volatile	independent	independent

© 2012 IBM Corporation

8) System Managed Structure Duplexing
The matrix

Structure Mode / Case #	LOGGER DUPLEX	STG_DUPLEX = NO	STG_DUPLEX = YES	
			DuplexMode = COND	DuplexMode = UNCOND
SIMPLEX				
Case 1	UNCOND/ COND	Local Buffers	Staging DS	Staging DS
Case 2	UNCOND/ COND	Local Buffers	Local Buffers	Staging DS
DUPLEX				
Case 3	UNCOND/ COND	Structure, Local Buffers	Structure, Staging DS	Structure, Staging DS
Case 4	UNCOND/ COND	Structure, Local Buffers	Structure, Local Buffers	Structure, Staging DS
Case 5a	UNCOND	Structure, Local Buffers	Structure, Local Buffers	Structure, Staging DS
Case 5b	COND	Structure	Structure	Structure

© 2012 IBM Corporation

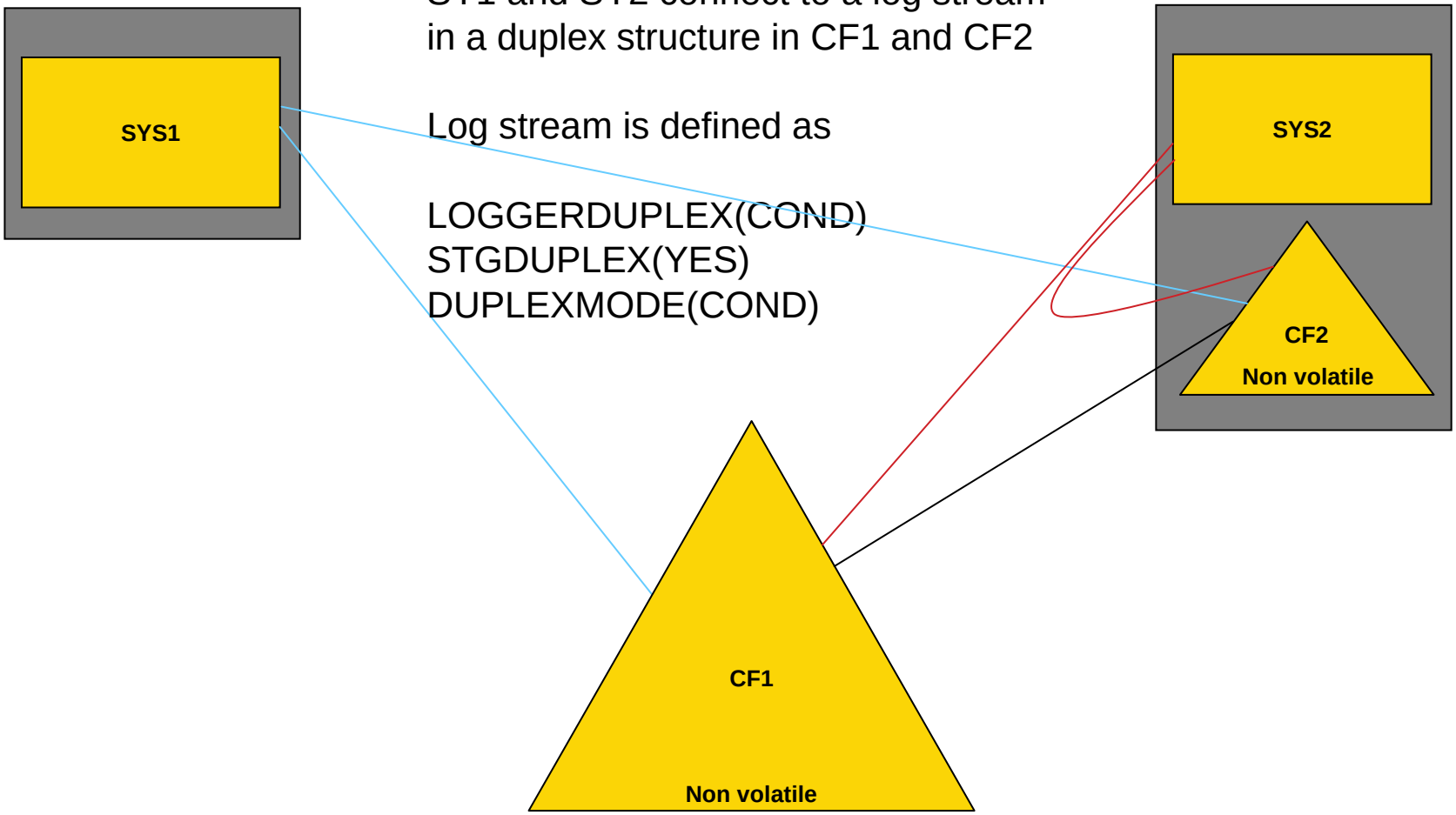
8) System Managed Structure Duplexing

What's the logger duplexing?

SY1 and SY2 connect to a log stream in a duplex structure in CF1 and CF2

Log stream is defined as

LOGGERDUPLEX(COND)
STGDUPLEX(YES)
DUPLEXMODE(COND)



© 2012 IBM Corporation

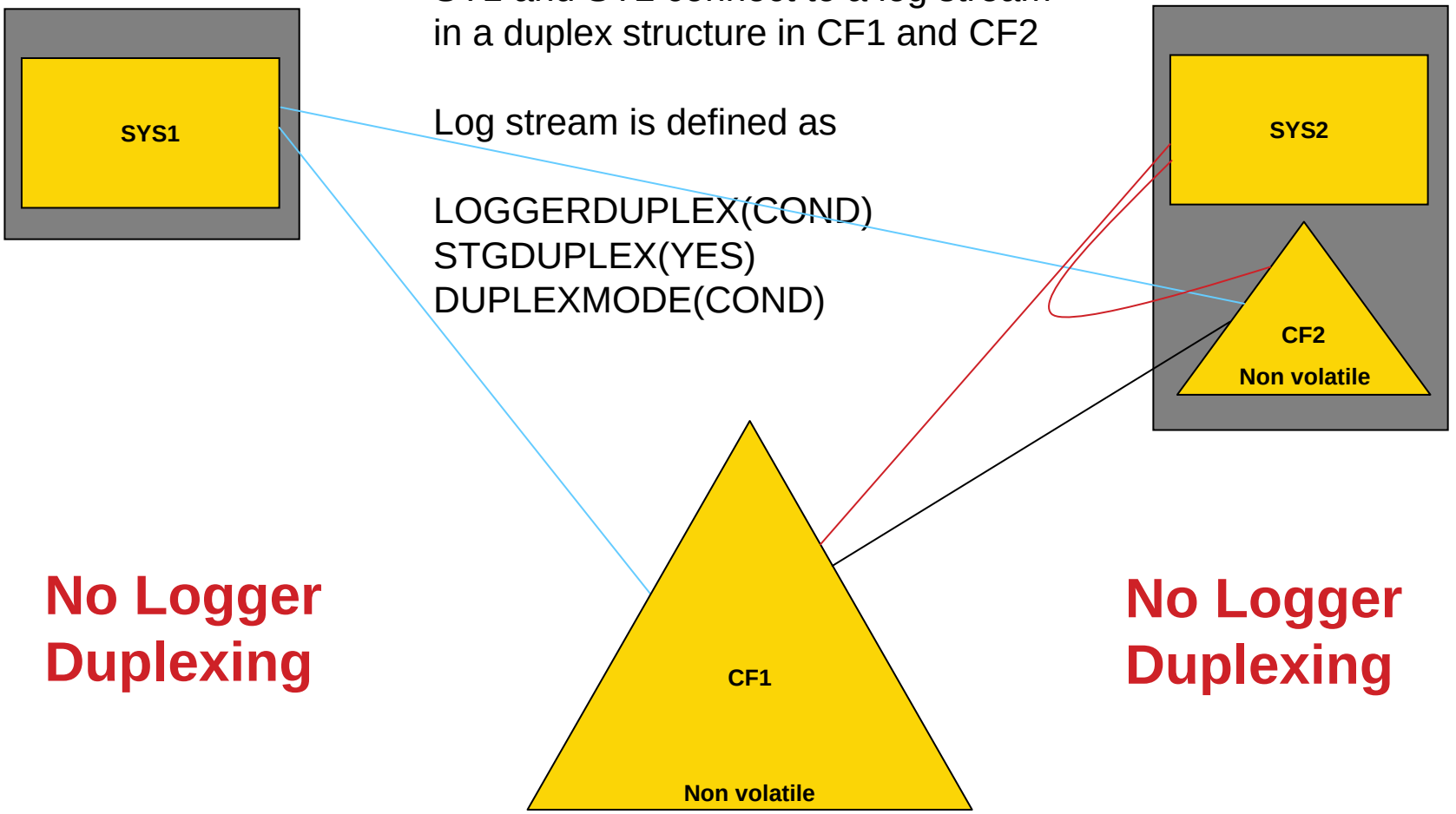
8) System Managed Structure Duplexing

What's the logger duplexing?

SY1 and SY2 connect to a log stream in a duplex structure in CF1 and CF2

Log stream is defined as

LOGGERDUPLEX(COND)
STGDUPLEX(YES)
DUPLEXMODE(COND)



No Logger Duplexing

No Logger Duplexing

© 2012 IBM Corporation

7

Incorrect LOGR CDS specified at IPL

Use extreme caution when specifying a used LOGR couple data set during IPL; the results could be disastrous.

© 2012 IBM Corporation

7) Incorrect LOGR CDS specified at IPL

- If you see these messages at IPL, stop and consider the consequences of moving forward
  - **IXC248E** COUPLE DATA SET *logr.couple.dataset* ON VOLSER *logvol* FOR LOGR MAY BE IN USE BY ANOTHER SYSPLEX.
  - **IXC247D** REPLY U TO ACCEPT USE OR D TO DENY USE OF THE COUPLE DATA SET FOR LOGR.

© 2012 IBM Corporation

© 2012 IBM Corporation

7) Incorrect LOGR CDS specified at IPL

- This messages indicate this is the first IPL of a system in a sysplex
- If the LOGR CDS you are about to bring in doesn't match the logger configuration…
  - Data in **CF list structures**
  - Logger offload and staging data sets for log streams
- .. results could be catastrophic
- Problems such as
  - Data corruption
  - Data access failures
  - Multi-system application outages
  - Logger abends
- Problems may appear long after an IPL

© 2012 IBM Corporation

7) Incorrect LOGR CDS specified at IPL

- This ‘new’ sysplex can be
  - A test sysplex
  - For disaster recovery testing
  - A production sysplex spawned from an existing sysplex
  - A production sysplex after a sysplex outage
- We have several possibilities here:

		CDS Log Stream Connection State		
		Never Connected	No Current Connections	Current Connections
CF State	Empty	ok		missing data
	Good Data	n/a		ok
	Bad Data	Problem 1	Problem 2	Problem 3

© 2012 IBM Corporation

7) Incorrect LOGR CDS specified at IPL

- Problems 1 and 2: never connected log stream or log stream with no current connections finds used CF lists
  - Logger cleans out the list structures before use for never connected log streams as of V1R7
  - Logger cleans out the list structures before use for log streams with no connections as of V1R9

- Note the potential to inadvertently clear CF data

© 2012 IBM Corporation

7) Incorrect LOGR CDS specified at IPL

- Problem 3: log stream with current connections finds used CF lists
  - Default action is - Logger assumes the CF data matches
    - Great if it does
    - Really bad if it does not
  - Logger tries to write the data in the CF lists to the log stream
    - If data doesn't conform to the log stream's expectations
      - *Logger abends at the next offload*
      - ***Could be days after IPL***
    - May fill the log stream with 'bad' data
  - No current solution, other than vigilance

© 2012 IBM Corporation

7) Incorrect LOGR CDS specified at IPL
Tips and Suggestions

- When using SETXCF COUPLE to change the primary dataset, Update COUPLExx too
- When IPLING a new sysplex consider –
  - Start with a new LOGR CDS
    - + Logger will initialize the structure as needed
    - - Have to define all needed log streams and structures
  - Copying an existing CDS with care
    - For CDSs with existing log stream connections
      - *Disconnect cleanly from all log streams before copying the CDS*
      - *Empty the CFs in the new sysplex*
    - Offload data sets for log streams may be defined either -
      - *Accept the access errors*
      - *Delete log block on log streams before copying the CDS*
      - *Copy all logger data sets with the CDS*

© 2012 IBM Corporation

7) Incorrect LOGR CDS specified at IPL

Tips and Suggestions

- Mirroring considerations
  - Mirror the LOGR CDS with the offload and staging data sets
  - Use staging data sets with applications that cannot tolerate loss of primary storage data

© 2012 IBM Corporation

6

Log Data Retention

How much and how long? It's up to the installation to decide how to manage log data, but installations can run out of data set extents or data set pool space before selecting a policy.

© 2012 IBM Corporation

6) Log Data Retention
How much and how long?

- It's improbable to expect log data to remain in the log stream forever
- Key factors
  - How much data is funneling into the log stream?
  - How long will the data be kept?
- When done with the data…
  - Data can be tossed
  - Data can be migrated out of the log stream

© 2012 IBM Corporation

6) Log Data Retention
The Options

- Application managed –
  - Writes and actively deletes log data
  - Manages how much and how long
  - Examples – IMS CQS, CICS DFHSHUNT
- Installation Managed –
  - The writing application does not delete log data
  - Options:
    - `RETPD / AUTODELETE` combinations
    - Application provided utility
      - ex. `IFASMFDL` for SMF or `IEAMDBLG` for Operlog
    - Application provided subsystem exit
      - ex. `IFBSEXIT` for Logrec – run EREP – see `SYS1.SAMPLIB(IFBEREPS)` for an example

© 2012 IBM Corporation

6) Log Data Retention
Notes

- `IXGDELET` marks the blocks logically deleted
- Blocks are physically deleted after a successful offload
- Retention period processing takes effect only on an offload
- Data sets are deleted during an offload
- Data sets are deleted when all the data in the data set is eligible for deletion
  - See next slide for when blocks are deleted
- `IXGBRWSE` requests with `VIEW(ALL)` may show deleted log blocks

© 2012 IBM Corporation

6) Log Data Retention
RETPD / AUTODELETE(NO) combinations

- `RETPD(0) AUTODELETE(NO)` – Default –
  - Blocks are removed after an `IXGDELET`
- `RETPD(0) AUTODELETE(YES)`
  - Blocks may be removed as soon as they are written
  - throw-away log stream
- `RETPD(>0) AUTODELETE(NO)`
  - Blocks are removed only after the retention period **AND** an `IXGDELET` operation
- `RETPD(>0) AUTODELETE(YES)`
  - Blocks are removed after the retention period **OR** an `IXGDELET` operation

© 2012 IBM Corporation

5

Manually Deleting Logger Data Sets

Logger manages the deletion of log stream offload and staging data sets. Logger notices when data sets are missing and can mark the log stream damaged.

© 2012 IBM Corporation

5) Manually Deleting Logger Data Sets

- Don’t delete logger data sets
- Use logger methods to manage data retention
- Missing log stream data sets cause Logger to
  - Alert the exploiter to a loss of data
  - Mark the log stream damaged
  - Part of the log becomes unrecoverable
- Recommendation: Security protection
  - Logger should have update access
    - To the HLQ.* or EHLQ.* data sets (ex.: IXGLOGR.*)
    - No one else needs access to logger data sets
  - Still doesn’t prevent admins from deleting data sets…
- To clean up a loss of data condition
  - Delete past the condition –or--
  - Delete all blocks (S IXGDELAB,LOGSTRM=xxx) –or--
  - Delete and redefine the log stream

© 2012 IBM Corporation

4

WTOR Responses

Logger messages IXG115A, IXG272E, and IXG312E represent significant delays to logger exploiters, but many installations have trouble determining the appropriate responses

© 2012 IBM Corporation

4) WTOR responses

Task Monitoring IXG271, IXG272

- Indicates a logger task is stuck performing data set
  - Allocation
    - New
    - Shared
    - Deletion
  - Recall
- May affect other log stream and other systems

- `IXG271I LOGGER DATA SET REQUEST IN group taskname SERVICE TASK DELAYED DURING THE PAST seconds SECONDS FOR LOGSTREAM logstream staging DSN=dsname, DIAG=diag`
- `IXG272E LOGGER group taskname TASK DELAYED, REPLY "MONITOR", "IGNORE", "FAIL", "EXIT".`

© 2012 IBM Corporation

4) WTOR responses
IXG272 replies

- `MONITOR` – continue to check the status, IXG272 issued again if delay continues
- `IGNORE` – stop monitoring, IXG272 stops for this instance
- `FAIL` – interrupt the request
  - Allocation is treated as if it failed
  - Logger task can do other work
  - May lead to abends
- `EXIT` – stop monitoring this delay, and all task delays
  - Use with extreme caution
  - Notification of logger task delays is lost
  - Status reset when logger is re-started

© 2012 IBM Corporation

4) WTOR responses
Offload Monitoring IXG310-IXG312

- Indicate an offload is stuck and log stream is stalled moving data from primary to secondary
- Warning message shown first
  - `IXG310I SYSTEM LOGGER CURRENT OFFLOAD IS NOT PROGRESSING FOR LOGSTREAM` *`logstream`* `STRUCTURE:` *`strname request`* `DSN=`*`dsnhlq.dsnlsn.dsnllq`*
- Action messages follow
  - `IXG311I SYSTEM LOGGER CURRENT OFFLOAD HAS NOT PROGRESSED DURING THE PAST` *`seconds`* `SECONDS FOR LOGSTREAM` *`logstream`* `STRUCTURE:` *`strname request`* `DSN=`*`dsnhlq.dsnlsn.dsnllq`*
  - `IXG312E OFFLOAD DELAYED FOR` *`logstream`*`, REPLY "MONITOR", "IGNORE", "FAIL", "AUTOFAIL", OR "EXIT".`

© 2012 IBM Corporation

4) WTOR responses
Offload Monitoring IXG310-IXG312

Default wait periods for monitoring messages (seconds)			
Type	Message	Allocating	Recall
Warning	IXG310	30	60
Action	IXG312	60	120

- Configurable with the IXGCNF parmlib in V1R13
- If the delay continues,
  - Primary storage will fill
  - Writes to the log stream will fail
  - This could be an application outage across multiple systems

© 2012 IBM Corporation

4) WTOR responses
IXG312 responses

- `MONITOR` – continue to check the status, IXG312 issued again if delay continues
- `IGNORE` – stop monitoring, IXG312 stops for this instance
- `FAIL` – cancels this offload
  - Another offload is automatically started
- `EXIT` – stop monitoring this offload, and all offloads on this system
  - Use with extreme caution
  - Notification of logger offload delays is lost
- `AUTOFAIL` – cancels this offload, and all future offload delay notifications on this system
  - Use with extreme caution
  - Ability to tailor monitor behavior is lost

- `AUTOFAIL` and `EXIT` status reset when logger is restarted

© 2012 IBM Corporation

4) WTOR responses
IXG115A

- Often follows messages `IXG310`, `IXG311`, `IXG312`, `IXG271`, and `IXG272`.

- `IXG115A CORRECT THE OFFLOAD CONDITION ON sysname FOR strname OR REPLY TASK=END TO END THE STRUCTURE TASK.`

- A `TASK=END` response
  - Allows the installation to resume requests on the connection task such as
    - Log stream connects and disconnects
    - Structure events
    - Log stream deletion requests
  - Might clear the allocation condition

© 2012 IBM Corporation

4) WTOR responses
IXG115A responses

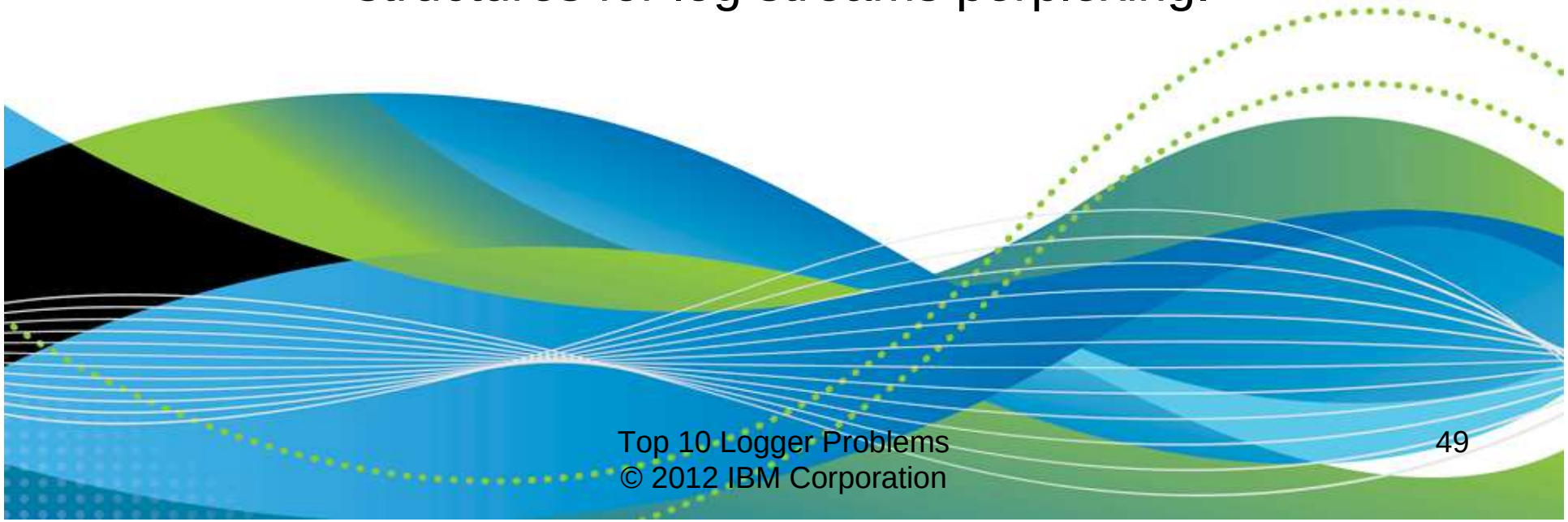
- Operator can
  - Wait for the condition to clear or
  - Reply to outstanding `IXG312` or `IXG272` messages or
  - Reply `TASK=END` to shoot the task
    - You will lose log stream connections related to the CF structure on that system
    - You may see related logger dumps and other diagnostic abends
    - You can reconnect to the log streams on the structure
    - You may still see offload hangs that caused the initial problem after a reconnect
    - Probably not a good idea to automate `TASK=END` reply

© 2012 IBM Corporation

3

Incorrect Structure Sizing

Sizing of logger resources is generally a difficult area for installations to configure. The wide array of possibilities make sizing CF structures for log streams perplexing.



© 2012 IBM Corporation

3) Incorrect Structure Sizing
Factors

- Incorrect structure sizing can lead to application delays and outages
- For some good recommendations see the CF sizer: http://www.ibm.com/systems/support/z/cfsizer/
- Many factors need to be take into account sizing a structure
  - `AVGBUFSIZE` – Average size of each log block
  - `MAXBUFSIZE` – Maximum size of each log block
  - `LOGSNUM` – Number of log streams per structure
  - `HIGHOFFLOAD` – % of structure space when the log stream offload starts
  - `LOWOFFLOAD` – % of structure space when the log stream offload finishes
  - *ResidencyTime* – how long the data is desired to remain in the CF
  - *WritePerSec* – number of log stream writes of AVGBUFSIZE per second

© 2012 IBM Corporation

3) Incorrect Structure Sizing
Notes

- Goal – make the log stream available for exploiter writes
  - Prevent the structure from becoming full during operation
  - Ensure an offload finishes before another starts
- These values can change
  - With increased workloads
  - With hardware changes
  - and require periodic monitoring and adjusting
- ‘Active Type’ log streams
  - Size the structure to hold all the active data
  - Account for workload spikes
- ‘Funnel Type’ log streams
  - Structure needs to handle incoming rates and make sure offload keeps up with spikes

© 2012 IBM Corporation

3) Incorrect Structure Sizing
Tips

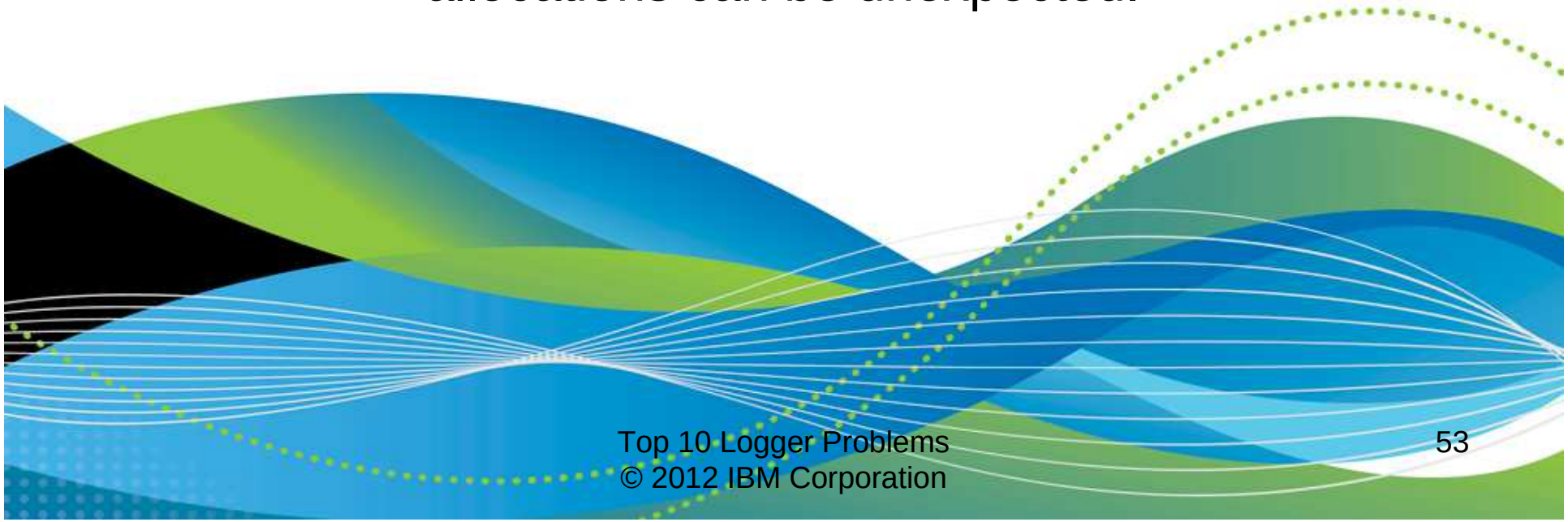
- Lower `HIGHOFFLOAD` value can
  - + Make more structure space available
  - + Decrease outage in cases of offload problems
  - - More CF space goes unused
- A higher `LOWOFFLOAD` value can
  - + Allow more data to be resident in the CF
  - + Shorten offload duration
  - - Data may remain in the CF longer than anticipated
- `AVGBUFSIZE` is dynamically managed after log stream connections
  - Is done to ensure optimal entry to element ratios
  - But might affect sizing expectations
- Have more than 1 `LOGSNUM`
  - Even if you intend 1 log stream per structure, choose 2
  - Easier to define and move resources among structures

© 2012 IBM Corporation

2

Incorrect Staging Data Sizing

For DASDONLY log streams, sizing staging data sets has the similar issues as sizing CFs. For CF log streams staging data set allocations can be unexpected.



© 2012 IBM Corporation

2) Incorrect Staging Data Set Sizing DASDONLY Log Streams

- For DASDONLY log streams consider the tips in sizing CF resources
- Default sizing - Not specifying a size will cause problems
  - An allocation default size may be used
  - This could be as small as 2 tracks (~100K)
  - This is too small
- Follow exploiter recommendations for sizing if given
- Logger recommends a minimum of 10MB in size for staging data sets
  - `STG_SIZE(2560)`
- The maximum staging data set size is 4GB-1
- For dasdonly log stream offloads, staging data set space is not freed until the end of the log stream offload
  - Less runway when writing during an offload
  - Consider a lower `HIGHOFFLOAD`

© 2012 IBM Corporation

2) Incorrect Staging Data Set Sizing
CF Log Streams

- CF log streams default to allocate staging data sets of the same size as the CF
  - Prevents the default minimum problems

- `DUPLEXMODE(COND)`
  - Prepare for data set allocation
    - may require setting up data classes
    - reserving dasd space
  - A volatility state change can cause use
- `STG_DUPLEX(NO)`
  - Potential for staging data set allocation

© 2012 IBM Corporation

1

Incorrect Log Stream Data Set Sizing

Log stream data set size often default to 2 tracks (~100k), which can create many small data sets that consume DIRCT, catalog and system resources and slow offloads.

© 2012 IBM Corporation

1) Incorrect Log Data Set Sizing

- Small data set problems
  - Frequent allocations slow down offloads
  - Eat up data set extents
    - Can cause extent shortages
    - Leads to log stream full conditions
    - And log stream exploiter outages across the sysplex
- Suggestions
  - Size data sets so that you do not need more than one new data set per offload
  - Set an offload data set size for all log streams
    - Even ‘Active Type’ log streams optimized to have all data in primary
    - Heavy log activity might drive an unexpected offload
    - This has caused outages
    - Especially when coupled with no data set extents.

© 2012 IBM Corporation

1) Incorrect Log Data Set Sizing

- Many options for specifying log stream data set size
  - `LS_SIZE` parameter
  - `LS_DATACLAS` parameter pointing to a DFSMS DATACLASS definition that contains a size value
  - DFSMS ACS routine pointing to a DATACLAS
  - Allocation exit setting the DALIKE text unit that points to a model data set
- Data set sizes can range between 64KB and 4GB-1
- Follow exploiter recommendations for sizing data sets
- Logger recommended minimum is 1MB for offload data sets
  - `LS_SIZE(256)`

© 2012 IBM Corporation

Backup

- See ‘System Logger Update’ Presentation for more logger information – session #10848

- 10) LOGR Couple Data Set Format Level and example
- 8) System Managed Structure Duplexing more examples

© 2012 IBM Corporation

10) LOGR Couple Data Set Format Level
CDS tips

- To prevent a single point of failure
  - Create a alternate couple data set and activate it
  - Prepare a spare CDS
  - Put CDSs on separate volumes
- For primary, alternate, and spare CDSs
  - Consider adding more
    - DSEXTENT records
    - LSR and LSTRR records

© 2012 IBM Corporation

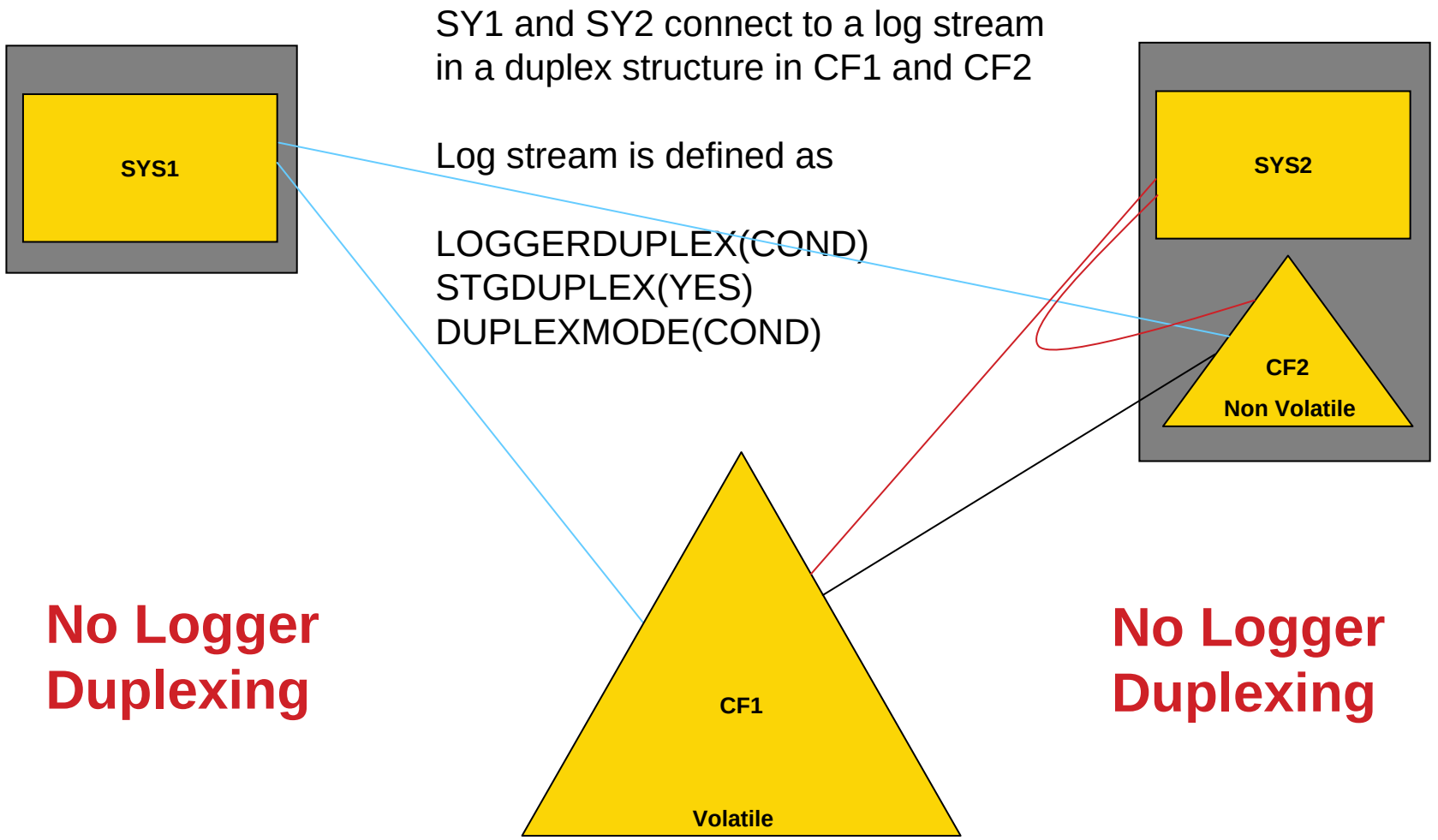
10) LOGR Couple Data Set Format Level
CDS switch examples

Example CDS switching:	Primary CDS	Alternate CDS	Spare CDS
Start: A is HBB6603 FMT level B is HBB7705 FMT level C is like B but with more DSEXTENT records	A	B	C
Installation wants to get to newest format level SETXCF COUPLE,PSWITCH,TYPE=LOGR is issued	B	n/a	C
Installation sets a new alternate with more DSEXTENT records SETXCF COUPLE,ACOUPLE=(C),TYPE=LOGR is issued	B	C	n/a
CDS D is defined like C but with more DSEXTENT records	B	C	D
Installation needs more DSEXTENT records.. SETXCF COUPLE,PSWITCH,TYPE=LOGR issued	C	n/a	D
…			

© 2012 IBM Corporation

8) System Managed Structure Duplexing

What's the logger duplexing?



© 2012 IBM Corporation

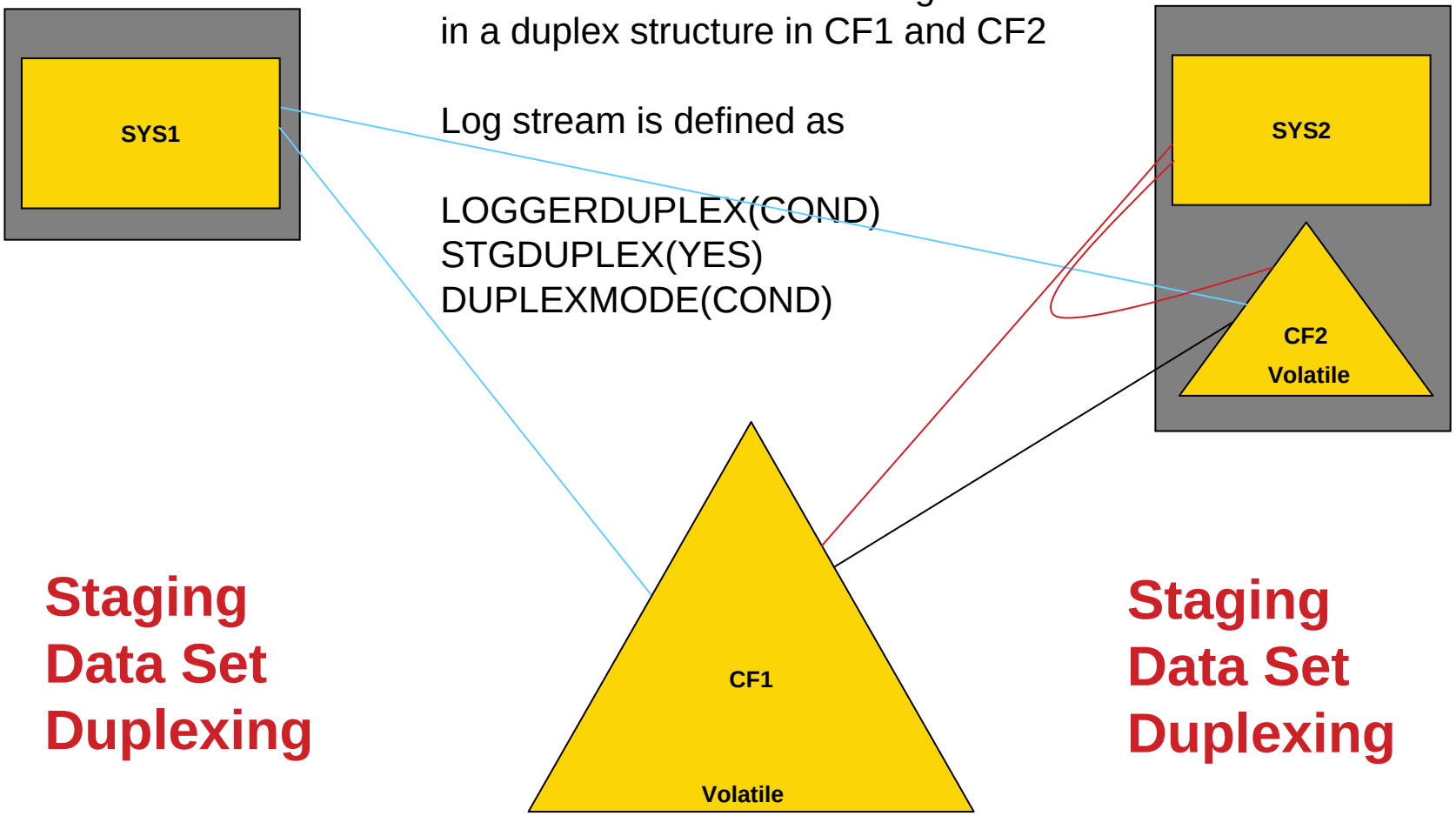
8) System Managed Structure Duplexing

What's the logger duplexing?

SY1 and SY2 connect to a log stream in a duplex structure in CF1 and CF2

Log stream is defined as

LOGGERDUPLEX(COND)
STGDUPLEX(YES)
DUPLEXMODE(COND)



Staging Data Set Duplexing

Staging Data Set Duplexing

© 2012 IBM Corporation

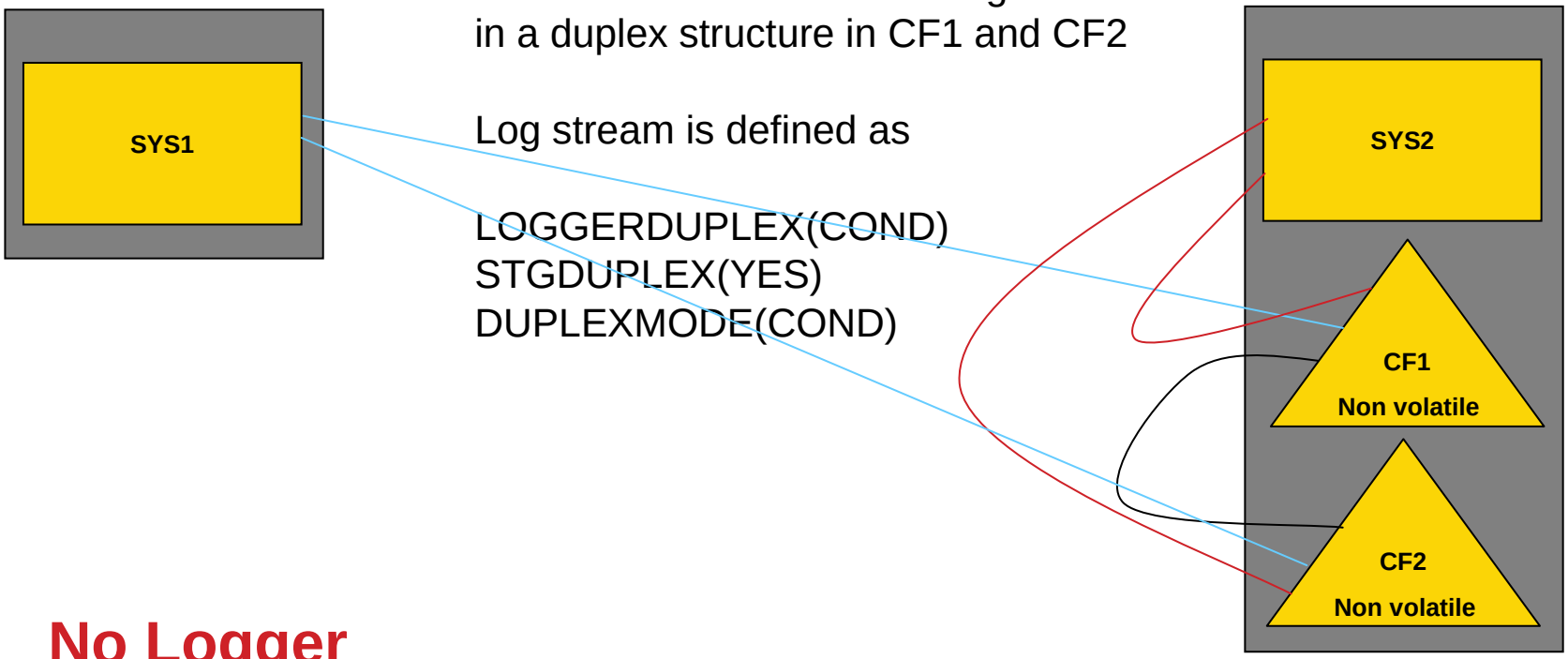
8) System Managed Structure Duplexing

What's the logger duplexing?

SY1 and SY2 connect to a log stream in a duplex structure in CF1 and CF2

Log stream is defined as

LOGGERDUPLEX(COND)
STGDUPLEX(YES)
DUPLEXMODE(COND)



No Logger Duplexing

Local Buffer Duplexing

© 2012 IBM Corporation

8) System Managed Structure Duplexing
Cases

1. Simplex structure and system to CF link is failure dependant or the CF structure is volatile
2. Simplex structure and system to CF link is failure independent and the CF structure is non-volatile
3. Duplex structure and both system to CF links are failure dependant or both CF structures are volatile
4. Duplex structure with
   - At least one non volatile CF
   - At least one failure independent system to CF link
   - Failure dependant link between the CFs
5. Duplex structure with
   - At least one non volatile CF
   - At least one failure independent system to CF link
   - Failure independent link between the CFs

© 2012 IBM Corporation