Automatic Restart Management (ARM) with IMS

Rich Lewis
IBM
IMS Advanced Technical Support

Copyright IBM Corporation 2001, 2002

Abstract and Trademarks

Abstract

Automatic Restart Management (ARM) is a function of OS/390. It may be used to automatically restart MVS jobs and started tasks. IMS/ESA 5.1 and later releases include ARM support. If IMS abends, ARM can restart it without any operator intervention. If MVS fails, ARM can restart IMS on another MVS system in the sysplex. ARM can also restart any CICS and DB2 systems used with IMS. This presentation includes an overview of ARM, an explanation of IMS's support for it, and recommendations for implementing ARM with IMS systems.   It explains how to increase the availability of IMS systems using data sharing, shared queues, DBCTL, or DB2.

The following are trademarks of International Business Machines Corporation or Tivoli Systems.

DB2	IBM	IMS	IMS/ESA	CICS
CICS/ESA	CICSPlex SM	MVS	MVS/ESA	OS/390
Parallel Sysplex	TME 10	VTAM	NetView	

CA-OPS/MVS is a registered trademark of Computer Associates International, Inc.

ARM and IMS

- **Agenda**

  - Automatic Restart Management (ARM) Overview
    - What is ARM?

  - IMS's Support for ARM
    - What has been added to IMS for ARM?

  - Implementing ARM for IMS
    - What needs to be done to use ARM with IMS?

ARM Overview

What is ARM?

ARM Purpose
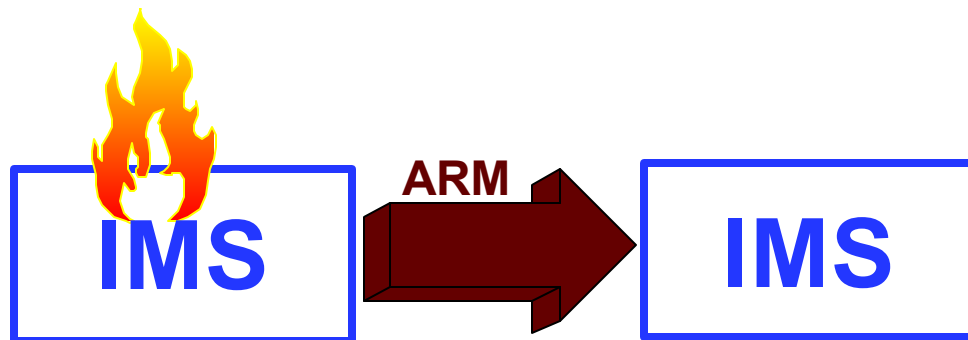
- **What ARM is:**
  - Automated job and started task restart
  - Managed by MVS
  - Sysplex wide
- **ARM benefits:**
  - Minimized outage times
  - Multiple processors exploitation
  - Increased availability

IMS → **ARM** → IMS

Configuration Requirement

- **Any supported release of OS/390 or z/OS**

- **Sysplex (monoplex or multiplex)**
  - Restarts within same sysplex

- **Restarts within same JES2 MAS or JES3 complex**

- **ARM Couple Data Set (CDS)**

ARM with IMS   6

Software Support

- **Jobs/started tasks must register with ARM**
  - Software must include code to invoke the register function

- **Some users of ARM:**
  - IMS
    - Includes TM, DBCTL, DCCTL, XRF, CQS, and FDBR
  - IRLM
  - DB2
  - CICS
  - CICSPlex SM
  - VTAM
  - TCP/IP
  - More than these!

Registration

- **Job or started task requirements:**
  - Must register with ARM as an element
  - May indicate dependency on other registrants
  - Indicates when ready to accept work
  - Deregisters when complete or restart not required
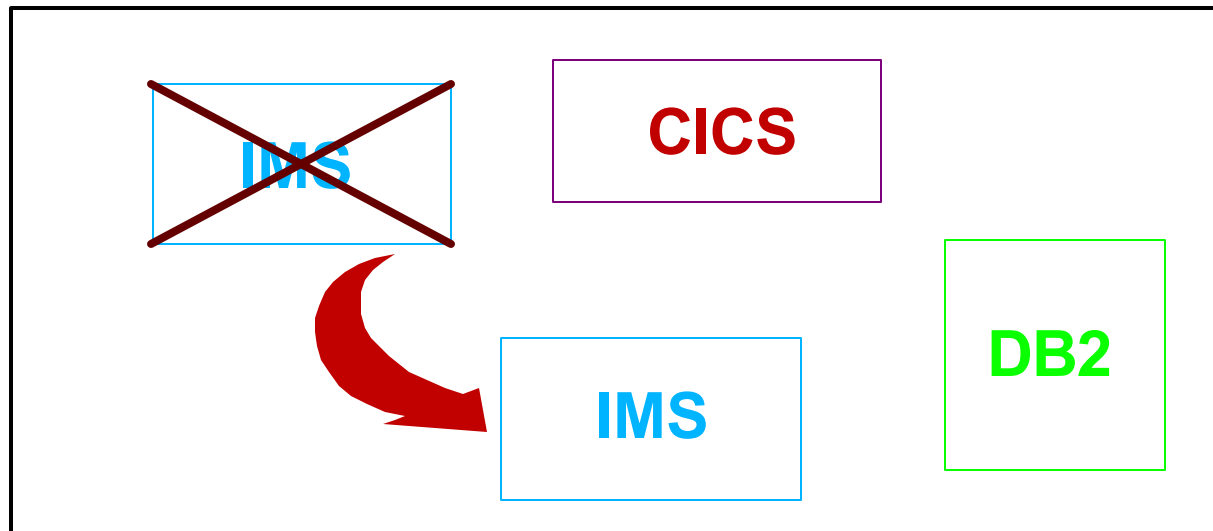
- **Only registered elements are restarted**
  - Registration is required for ARM to keep track of an element
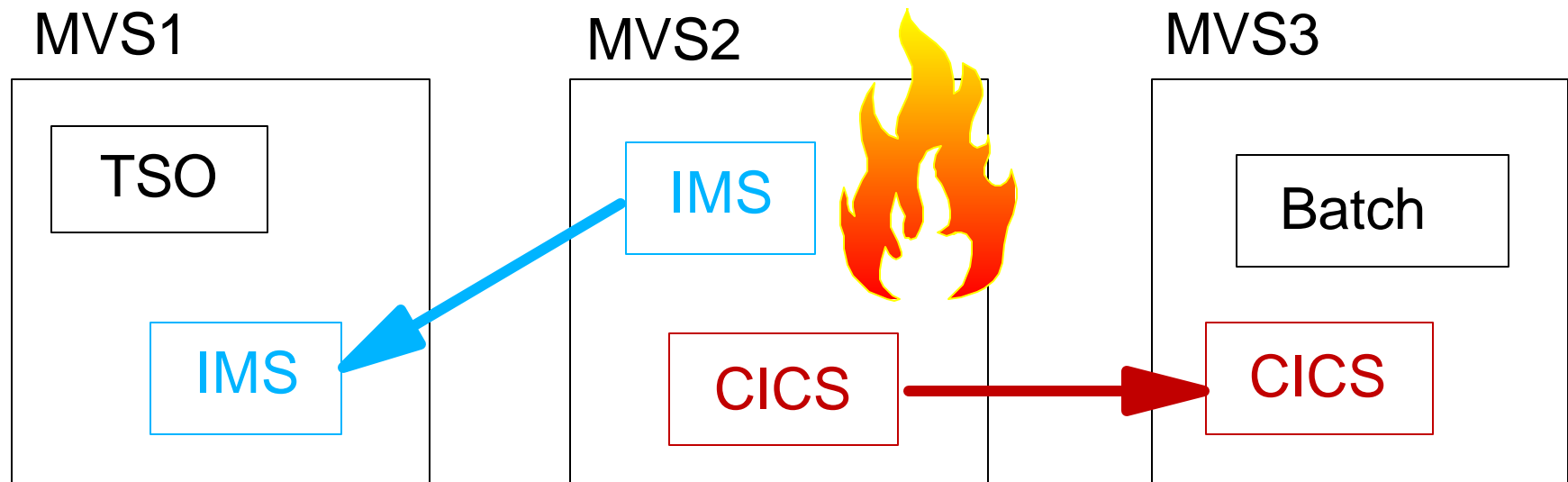
ABENDs

- **If registered element ABENDs (EOJ or EOM), ARM restarts element on same MVS**
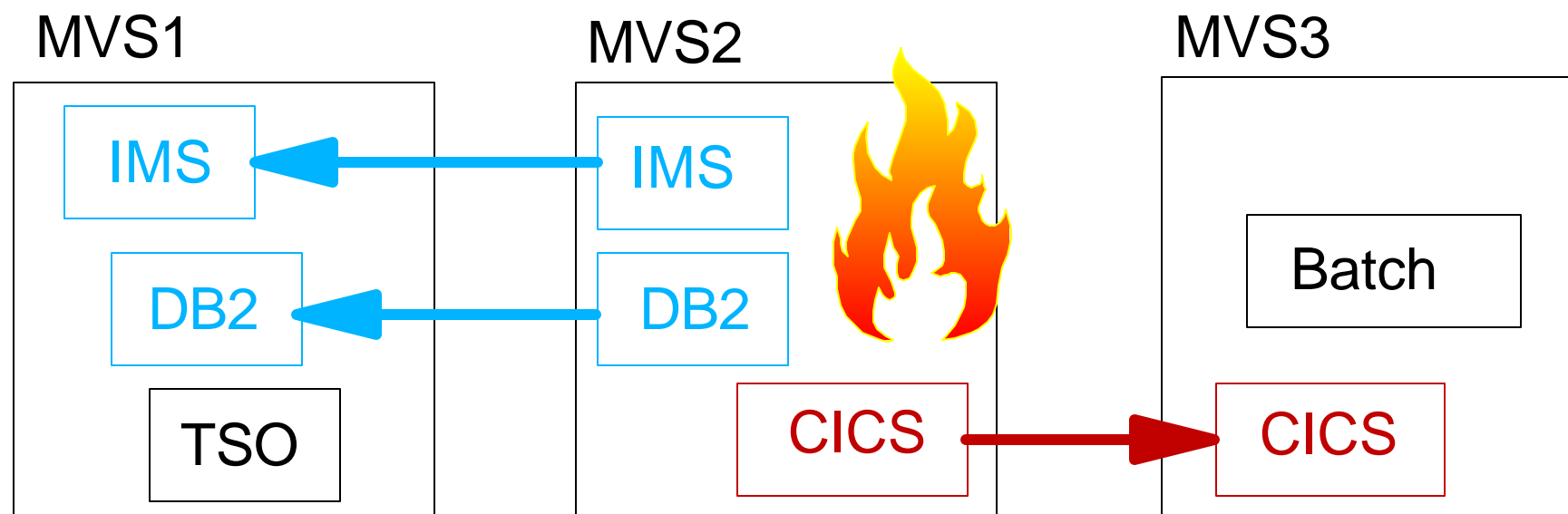
MVS

System Failures

- **If MVS fails, ARM restarts elements elsewhere in sysplex.**

MVS1

TSO

IMS

MVS2

IMS

CICS

MVS3

Batch

CICS

Restart Groups

- **Elements maybe <u>grouped</u> for restart on the same MVS**
  - Example: IMS and DB2

MVS1

IMS

DB2

TSO

MVS2

IMS

DB2

CICS

MVS3

Batch

CICS

Restart Groups

- **Elements in group may be restarted in specified order**
  - Example: DBCTL before CICS

- **Elements in group may be restarted at pacing intervals**
  - Example: At 20 second intervals

- **Minimum CSA/ECSA may be specified for target MVS system**

- **Target MVS selected by available capacity**
  - If using WLM goal mode

Restart Conditions

- **Conditions for restart may be specified**
- **Restart only on ABENDs**
  - Example: Application development system
- **Restart on ABEND or MVS system failure**
  - Example: Production system

MVS A

| Job A1 | ← ABEND — Job A1 | System failure → | Job A1 |

MVS B

Job B1

Job A2 ← ABEND — Job A2

Job B2
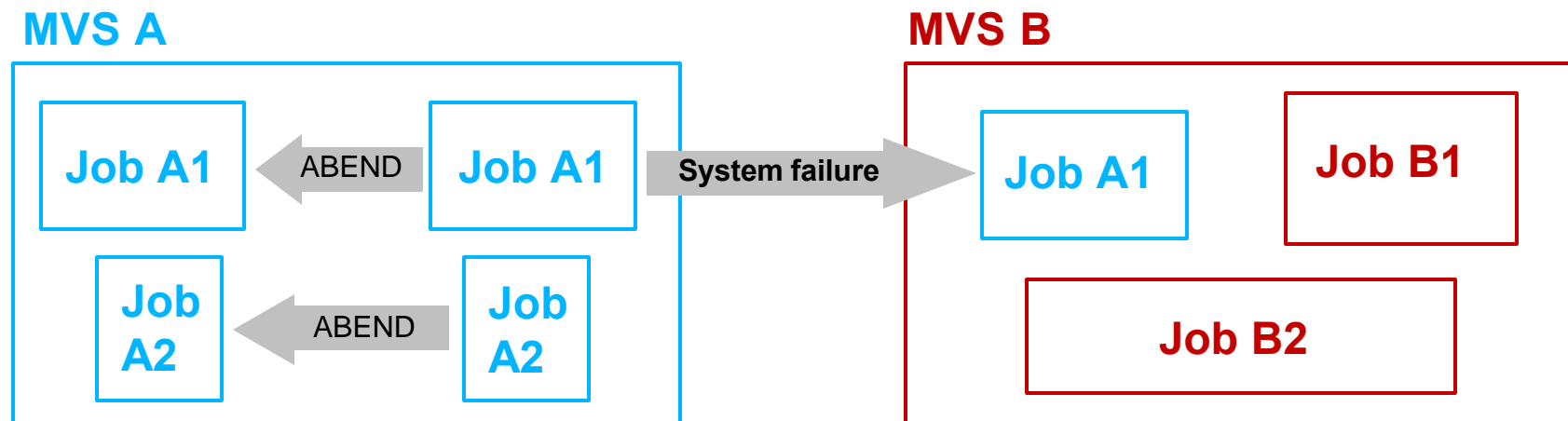
Restart Method

- **Three choices:**
  - *Persistent*
    - Same JCL or command text as before
  - *JOB*
    - Specified data set (or member) containing JCL
  - *STC*
    - Specified command text

- **Choice may be different for ABENDs and system failures**

ARM Policy

- **ARM policy must be active for ARM to restart elements**

- **Policy activated by command**
  - `SETXCF START,POLICY,TYPE=ARM,POLNAME=policy_name`

- **System default policy or customized policy may be used**

- **Policy defined with Administrative Data Utility**

ARM Policy (cont.)

- **For *elements* the policy defines:**
  - Number of restarts in an interval
    - Zero indicates that the element should not be restarted

  - Whether to restart on system failures and ABENDs
    - Or only on ABENDs

  - Method of restart
    - Persistent, JOB, or STC

  - Maximum wait times on restart
    - Registration - time to register
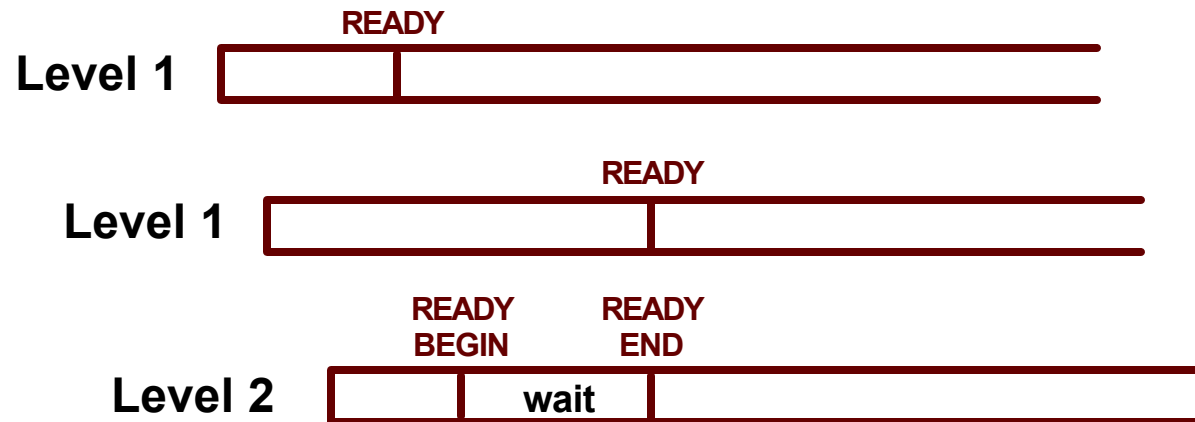   - Ready - time to become ready

ARM Policy (cont.)

- **For *restart groups* the policy defines:**

  - Elements in group

  - Candidate systems for restarts
    - Controls where the group will be restarted

  - Levels for elements
    - Controls the order in which elements are restarted

  - Pacing intervals between restarts of elements

  - Minimum CSA/ECSA required on systems for restarts

Levels

- **Levels**
  - Elements are assigned a level
  - Levels control order of cross-system restart activities
    - Restarts after system failures
  - Elements are restarted in level order
  - Elements become ready in level order

```
                    READY
Level 1   ┌────────────┬──────────────────────────────┐
          └────────────┴──────────────────────────────┘

                              READY
Level 1   ┌──────────────────────┬──────────────────────┐
          └──────────────────────┴──────────────────────┘

                    READY         READY
                    BEGIN         END
Level 2   ┌─────────┬──────wait──────┬──────────────────┐
          └─────────┴────────────────┴──────────────────┘
```

Program Interface to ARM

- ***Register***
  - Eligible for restart

- ***Ready***
  - Ready to do work
  - Used to coordinate restarts of multiple elements
  - Applies only to cross-system restarts

- ***Wait for Predecessor***
  - Wait for other element to become ready
  - Applies only to cross-system restarts
  - IMS does not issue this

- ***Associate***
  - Disable restart of another registrant
  - IMS issues this for XRF and FDBR

- ***Deregister***
  - Ineligible for restart

Policy Example

```
DATA TYPE(ARM)

DEFINE POLICY NAME(MYARMPOL)

RESTART_ORDER

  LEVEL(1)                     /* Start these elements first and  */
                               /* make them ready first           */
    ELEMENT_TYPE(SYSIMS)       /* Include elements of type SYSIMS  */
    ELEMENT_NAME(DB2$DB2P)     /* Include element named DB2$DB2P   */

  LEVEL(2)                     /* Start these elements second and  */
                               /* make them ready second           */
    ELEMENT_NAME(CICSP*)       /* Include elements with names      */
                               /* beginning with CICSP             */
```

Policy Example (CONT.)

```
RESTART_GROUP(GRP1)
TARGET_SYSTEM(SYS1,SYS2,SYS3) /* Restart on these systems        */
   FREE_CSA(200,1000)               /* Must have 200K of CSA and      */
                                    /* 1000K of ECSA to restart group */
   RESTART_PACING(30)               /* Delay 30 sec. between restarts */

ELEMENT(IMSP)                    /* For IMSP                          */
   RESTART_ATTEMPTS(3,300) /* Restart 3 times in 300 sec.         */
   RESTART_TIMEOUT(90)      /* 90 sec. to reregister after restart*/
   READY_TIMEOUT(1000)      /* 1000 sec. to issue READY after     */
                            /* restart                            */
   TERMTYPE(ALLTERM)        /* Restart on ABEND or system failure */
   RESTART_METHOD(ELEMTERM,PERSIST)     /* Same JCL after ABEND  */
   RESTART_METHOD(SYSTEM,JOB,'IMS.JCL(IMSPSYS)')
                 /* Special JCL for restart on another system  */

ELEMENT(CICSP*) ...

ELEMENT(DB2$DB2) ...
```

ARM Exits

- **Workload restart exit**
  - Used to prepare a system for cross-system restart
    - Invoked before restarts
    - Exit routine is passed which system failed and elements to be restarted
    - Could cancel lower priority work on this system

- **Element restart exit**
  - Used to modify restarts of elements
    - Invoked for each element to be restarted on this system
    - Exit routine is passed the element name, reason for restart, and restart method
    - May cancel restart, change to different restart method, modify the restart method, or allow restart to proceed

ARMWRAP

- **ARMWRAP provides ARM support for programs which have not provided their own ARM support**

  - ARMWRAP is a program
    - Registers and unregisters an address space with ARM
    - Runs unauthorized, therefore, must be SAF authorized to the element resource

  - ARMWRAP is executed as steps in a job or proc
    - First step executes ARMWRAP to register with ARM
    - Last step executes ARMWRAP to unregister from ARM

  - ARMWRAP is available from the Web

ARMWRAP (CONT.)

- **Example JCL to provide ARM support for APPLPGM**
  - Step REGSTEP registers with ARM
  - Step UNREGSTP unregisters from ARM

```
//PROCNAME PROC ...
//*  Register element 'EXAMPLE' using element type of
//*   'XAMP'with ARM.  Restart on all types of terminations.
//REGSTEP  EXEC PGM=ARMWRAP,
//       PARM=(REQUEST=REGISTER,READYBYMSG=N,
//       TERMTYPE=ALLTERM,ELEMENT=EXAMPLE,ELEMTYPE=XAMP)
//APPLSTEP EXEC PGM=APPLPGM
                 .
                 .
                 .
//UNREGSTP EXEC PGM=ARMWRAP,PARM=(REQUEST=UNREGISTER)
```

IMS's Support for ARM

What has been added to IMS for ARM?

IMS's Support for ARM

- **IMS TM, DBCTL, DCCTL, XRF, and FDBR are supported**
  - ARMRST=Y|N execution parameter
  - ARMRST=Y causes IMS to register
- **Control region**
  - Registers with ARM
  - Control region starts DLI SAS and DBRC which do not register
- **Dependent regions, DLI and DBB batch, utilities**
  - Do not register
  - Are not restarted
- **XRF and FDBR**
  - Register with ARM
  - Special treatment for tracked control regions

IMS Interface to ARM

- **Element name is the IMSID**
  - May be used in `ELEMENT(imsid)` and `ELEMENT_NAME(imsid)` in ARM policy
- **Element type is SYSIMS**
  - May be used in `ELEMENT_TYPE(SYSIMS)` in ARM policy
- *REGISTER* **at initialization**
  - Eligible for restart
- *READY* **at end of restart**
  - Higher level elements in group cannot become ready until this occurs
  - Used only on cross-system restarts
- *DEREGISTER* **at normal termination**
  - Not eligible for restart

Special Treatment

- **Restarts by ARM ignore AUTO=N**
  - /ERE OVERRIDE set if not XRF alternate
  - /ERE BACKUP set for XRF alternate

- **ABEND before restart completes causes *DEREGISTER***
  - Avoids restart that is likely to fail

- **When XRF alternate or FDBR begins tracking, restarting of tracked IMS is disabled.**
  - XRF alternate will be restarted if it fails
  - FDBR will be restarted if it fails

Restarts Not Attempted

- **IMS *DEREGISTERs* for these ABENDS:**

  - U0020 - `MODIFY` command
  - U0028 - `/CHE ABDUMP`
  - U0604 - `/SWITCH SYSTEM` (XRF)
  - U0758 - Message queue full
  - U0759 - Message queue I/O error
  - U2476 - CICS XRF takeover

- **Restart is inappropriate or would fail**

ARM with IMS

IMS Support Summary

● **What has been added to IMS for ARM?**

  ■ ARM handling: register, ready, deregister

  ■ Automatic handling of appropriate restarts
    • AUTO=Y, /ERE OVERRIDE, ...

  ■ Deregistration when restart would fail

  ■ Parameter to turn off ARM registration
    • ARMRST=N

It's automatic!

Implementing ARM for IMS

Implementing ARM for IMS

- **ARM restarts IMS by default if an ARM policy is active**
  - `SETXCF START,POLICY,TYPE=ARM`

- **Ways to turn off restarts:**
  - `ARMRST=N` in JCL

  or

  - `RESTART_ATTEMPTS(0`) in ARM policy

What to Restart

- **Typical restart decisions**

  - Production IMS systems
    - ARM restart for ABENDs and system failures

  - Development IMS systems
    - ARM restart for ABENDs
    - Do not ARM restart for system failures

  - Test IMS systems
    - Never ARM restart, unless testing ARM functions

Restart Groups with IMS

- **DEFAULT group**

  - All elements register in some group
  - If not specified in policy, element is in `DEFAULT` group

- **Reasonable scheme:**

  - Do not restart any element in `DEFAULT` group
    - `RESTART_GROUP(DEFAULT)RESTART_ATTEMPTS(0)`

    ➡ This prevents accidental restarts of IMS, CICS, DB2, etc.

  - Assign restartable elements to defined groups

    ➡ This allows intentional restarts of IMS, CICS, or DB2

Restart Groups

- ## CICS
  - ### Place all CICSs using DBCTL in same restart group with IMS
    - IMS could be either DBCTL or IMS TM providing DBCTL services
    - CICS AORs using DBCTL must be in the group with IMS
    - CICS TORs and FORs do not need to be in group with IMS and AORs
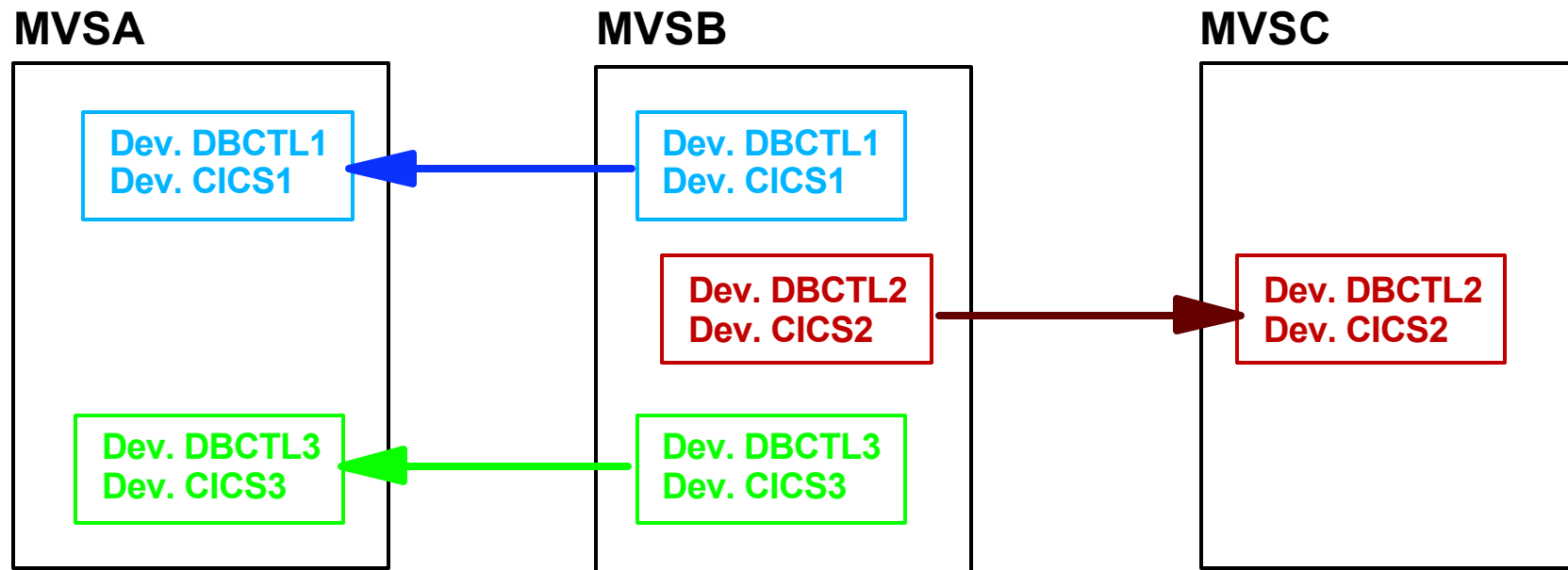   - ‣ They do not connect to IMS

- ## DB2
  - ### Place DB2 used by IMS in same restart group with IMS
  - ### Place DB2 used by CICS AORs in same restart group with AORs
    - CICS TORs and FORs do not need to be in same group with DB2
      - ‣ They do not connect to DB2

Restart Groups (cont.)

- ● **Make groups small**
  - ▪ Minimize number of elements in a group
  - ▪ Do not force ARM to keep multiple subsystems in one MVS unnecessarily
  - ▪ If elements can run on separate MVSs, place them in separate groups
  - ▪ Control candidate MVSs for restart with `TARGET_SYSTEM` parameter

MVSA

MVSB

MVSC

Dev. DBCTL1
Dev. CICS1

Dev. DBCTL1
Dev. CICS1

Dev. DBCTL2
Dev. CICS2

Dev. DBCTL2
Dev. CICS2

Dev. DBCTL3
Dev. CICS3

Dev. DBCTL3
Dev. CICS3

ARM with IMS

How to Restart

- **Restart Method**
  - PERSIST is usually appropriate
    - Same JCL as before
   - IMS chooses AUTO=Y and OVERRIDE (if appropriate)

- **Restart order and pacing**
  - If many CICSs are in the group, pacing may be desirable
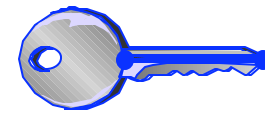 - It is reasonable to restart DB2 and IMS concurrently

Data Sharing

- **When block level data sharing IMS subsystem fails**
  - Locks protecting in-flight updates are held
  - Requests for the locks result in lock rejects
  - Lock rejects generally cause application ABENDs (U3303)
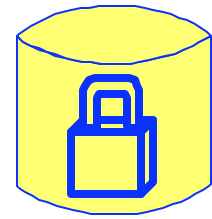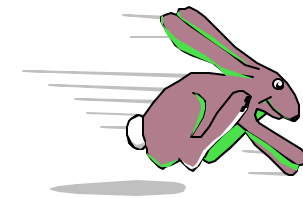
Database records and blocks

- **Held locks are released by emergency restart backout**

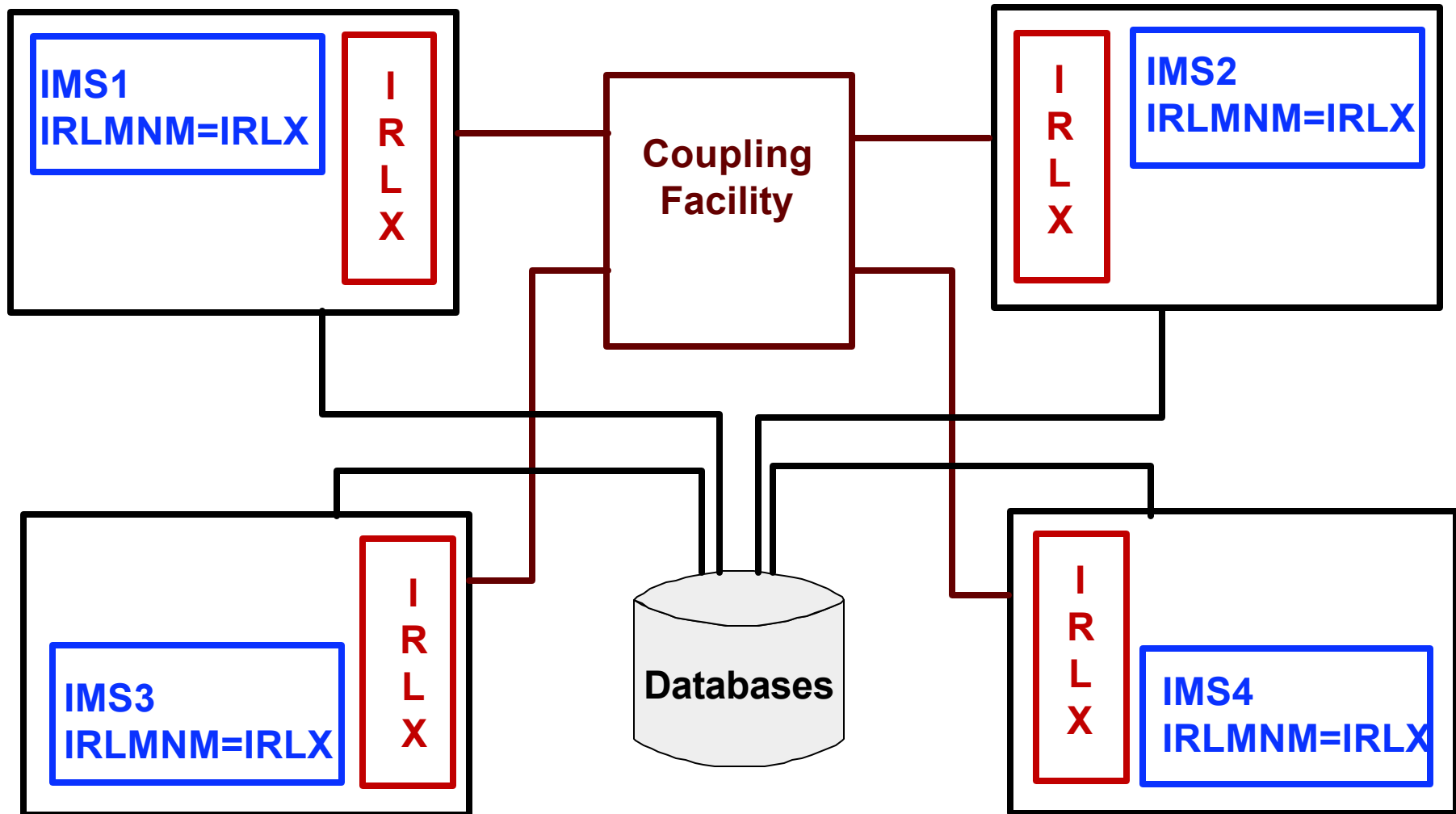- **ARM should be used to automatically restart failed IMS subsystems quickly**

IRLM

- **ARM support added by APAR PQ06465**
  - IRLM *registers* to ARM
- **IMS specifies IRLM name (`IRLMNM=`)**
  - This IRLM must reside on MVS where IMS runs
  - If IMS moves to another MVS either
    - IRLM name must be changed (different IRLMNM=)

    or
    - IRLM with same name must reside on new MVS
- **Solution!**
  - Use the same name for all IRLMs in data sharing group
  - Start an IRLM with this name on each candidate MVS system
  - Use ARM to restart IRLM only after IRLM abends

Data Sharing Example

All IRLMNMs are IRLX

IMS1
IRLMNM=IRLX

IRLX

Coupling
Facility

IRLX

IMS2
IRLMNM=IRLX

IMS3
IRLMNM=IRLX

IRLX

Databases

IRLX

IMS4
IRLMNM=IRLX

VTAM

- **VTAM applid (acbname)**

  - Must be active on system
    - Cannot be active on multiple systems

  OR

  - Must match active model application program definition

VTAM

- **Model Application Program Definition**
  - VTAM APPL statement includes wildcard character(s) in label
    - Wildcards are * and ?
    - ACBNAME operand cannot be coded
    - Example statement
      ```
      IMSP*    APPL    AUTH=(PASS,ACQ,SPO),PARSESS=YES
      ```

  - May be active on multiple systems concurrently
    - Matching application program (acbname) may be opened on any of these systems

  - Recommended for use with Parallel Sysplex
    - Application (IMS, CICS, etc.) may execute on any system

Additional IMS Support

- **Common Queue Server (CQS)**
  - ARM support included
  - ARMRST=Y|N parameter
  - Include CQS in restart group with its IMS

- **Fast Database Recovery (FDBR)**
  - ARM support included
  - ARMRST=Y|N parameter for FDBR region
    - Determines if FDBR will be restarted if it fails during tracking
  - When FDBR is active, restarting of the tracked IMS is disabled
  - Do not include FDBR in restart group with its IMS
    - IMS and FDBR will usually run on different systems

IMS Connect

- ● **IMS Connect has no explicit ARM support**
  - IMS Connect provides TCP/IP connectivity for IMS
  - ARMWRAP may be used to provide ARM support

```
//PROCNAME PROC ...
//*  Register IMS Connect with ARM using element type of
//*  SYSIMS.  Restart on all types of terminations.
//REGSTEP  EXEC PGM=ARMWRAP,
//       PARM=(REQUEST=REGISTER,READYBYMSG=N,
//       TERMTYPE=ALLTERM,ELEMENT=IMSC1,ELEMTYPE=SYSIMS)
//IMSCONN  EXEC PGM=HWSHWS00,REGION=4M,TIME=1440,
//              PARM='BPECFG=BPECFGHT,HWSCFG=HWSCFG00'
//STEPLIB  DD   DSN=IMS.SDFSRESL,DISP=SHR
//         DD   DSN=IMS.HWS.RESLIB,DISP=SHR
//PROCLIB  DD   DSN=IMS.PROCLIB,DISP=SHR
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&SOUT
//UNREGSTP EXEC PGM=ARMWRAP,PARM=(REQUEST=UNREGISTER)
```

DB2

- **When ARM is active, DB2 always *registers***
  - No parameter to turn it off
  - To avoid restarts, you must assign DB2 to a restart group with

    `RESTART_ATTEMPTS(0)`

- **DB2 element name**
  - If not data sharing,

    name is `DB2$` concatenated with member name
  - If data sharing,

    name is DB2 group name concatenated with member name

DB2 (cont.)

- **DB2 element type**
  - Element type is `SYSDB2`
  - Element type may be used to assign an element to a level

- **IRLM with DB2**
  - May use ARM or AUTO START to restart IRLM after system failure
  - Use ARM to restart IRLM after IRLM ABEND

CICS

- **When ARM is active, CICS always registers**
  - No parameter to turn it off
  - To avoid restarts, you must assign CICS to a restart group with

    ```
    RESTART_ATTEMPTS(0)
    ```

- **CICS element name**
  - Element name is `SYSCICS_` concatenated with the applid
  - May be used to assign an element to a group or level

- **CICS element type**
  - Element type is `SYSCICS`
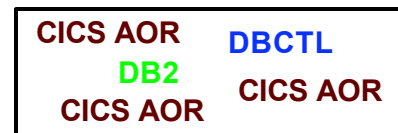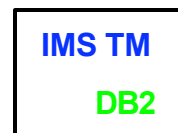 - May be used to assign an element to a level

Implementation Summary

IMS
THE WORLD DEPENDS ON IT

● **What needs to be done to use ARM with IMS?**

✓ **Place subsystems in restart groups**

IMS TM		CICS AOR	DBCTL		Production
DB2		DB2	CICS AOR		Development
		CICS AOR		Test	

✓ **Decide on restart method and timings**

Persistent, JOB, or STC   Restart order, Levels, Time-outs, ...

✓ **Create VTAM model application program definitions**

```
IMSP*   APPL AUTH=(PASS,ACQ),...
```

✓ **Add ARMWRAP steps to IMS Connect (ITOC) JCL**

✓ **Create a policy**

```
RESTART_GROUP(GRP1)
TARGET_SYSTEM...
```

✓ **Start the policy**

```
SETXCF START,POLICY,TYPE=ARM,POLNAME=MYARMPOL
```

Summary and Benefits

- **What is ARM?**
  - Automated restart management
  - Managed by MVS
  - Sysplex wide
  - Works for ABENDs and MVS failures

It's automatic!

- **Benefits**
  - Increased availability
  - Minimized restart times
  - Simple to implement
  - Supported by IMS, IRLM, DB2, CICS, CPSM, VTAM, TCP/IP,...
  - ARMWRAP is available to support products lacking explicit support

It's quick!