

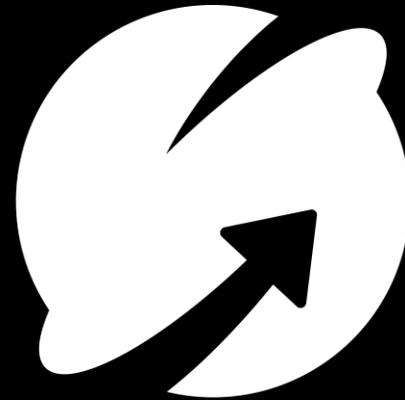
# Improving IMS Availability

*Dynamic Resource Definition and  
IMS HP System Generation Tools*

**Rick Engel**

raengel@us.ibm.com

CTP and Team Leader – IBM NA  
Technical Sales / IMS and Tools



**IBM**

**IMS Tools**

for z/OS

# Topics

- IMS application resources and libraries
- History of IMS resource definition – before DRD
- History of IMS resource definition – DRD and RDDS
- IMS resource definition with IMS Resource Repository
- Use case scenarios for DRD with IMS Resource Repository

## References

- [IMS Fundamentals - Define system resources dynamically](#)
- [Introducing the IMS Repository](#)
- Other DRD presentations:
  - [IMS12 IMS DRD and the Repository - IMS UG June 2013 Sydney](#)
  - [Dynamic Change Management In IMS IMS RUG Jan 2014 e-meeting east](#)



# Terminology for IMS Resources

- “Runtime resource definitions”
  - Resource information (attributes, status, relationship to other resources) stored in an online IMS as *control blocks*:
    - *Database directories = DDIRs*
    - *Program directories = PDIRs*
    - *Fast Path routing codes = RCTEs*
    - *Scheduler message blocks for transactions = SMBs*
  - Controlled/maintained by IMS itself
  - Reside in IMS control region
  
- “Stored resource definitions”
  - Resource information that is stored offline, from which IMS builds runtime resource definitions
    - Reside in system data sets, such as IMS.MODBLKS, resource definition data set (RDDS), or repository
  - Controlled/maintained by user



# IMS Application Resources and Libraries – The Basics

- MODBLKS Library

- Contains the control blocks for databases, programs, transactions, and MFS formats
  - Database control blocks are used by IMS online system to allocate, open, and close database data sets
  - Program control blocks are used by IMS online system to schedule application containers (MPP, BMP, IFP, JMP, or JBP) to run application programs
  - Transaction control blocks are used by IMS online system to associate transaction code with application programs

- ACB Library (ACBLIB)

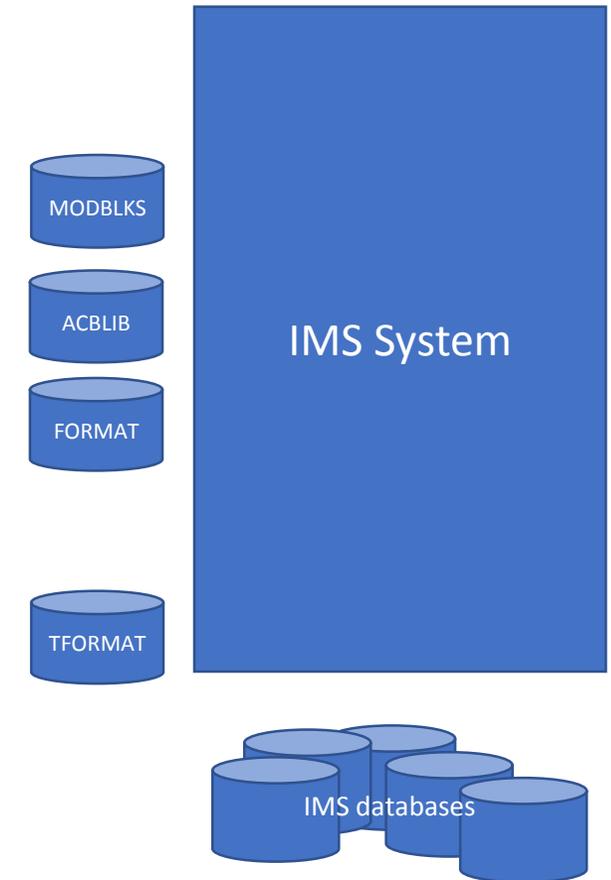
- Contains database descriptors (DBDs) and program descriptors (PSBs)
  - A DBD is used by the IMS online system to create internal control blocks to access database data sets and data structures in the database data sets
  - A PSB is used by IMS online system to provide the application program identified by the PSB with application views for databases that are used by the program

- FORMAT Library

- Contains MFS maps produced by the MFS Language and Service utilities
  - These maps are required by MFS-supported terminals and are to be used by the IMS online system

- TFORMAT Library

- Contains the online MFS maps for test mode online execution



# MODBLKS Runtime Resource Definitions

- In a running IMS system, a *MODBLKS runtime resource definition* is represented by a control block that defines the resource and its attributes
- The following table shows MODBLKS runtime resources and the SYSGEN macro used to generate their respective control blocks:

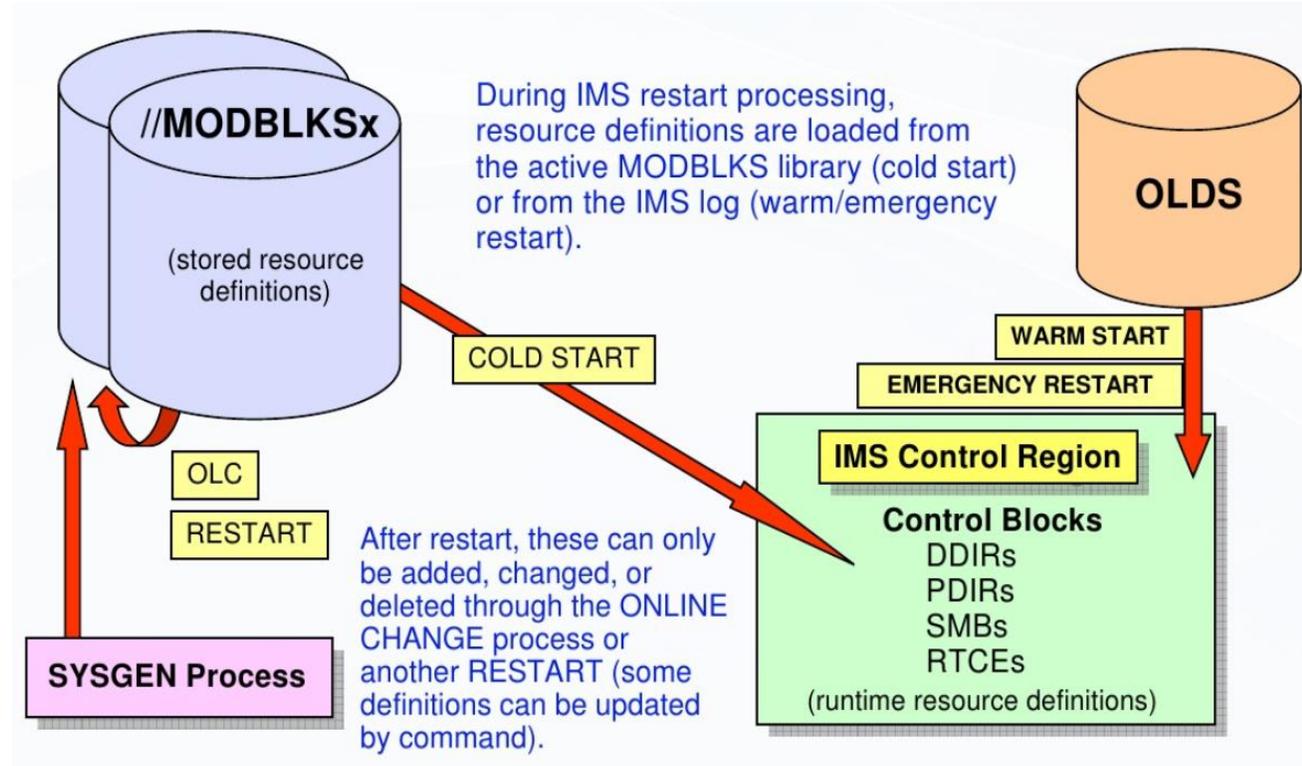
RESOURCE TYPE	SYSGEN MACRO	CONTROL BLOCK
Database	<b>DATABASE</b> (DB/TM, DBCTL)	<b>DDIR</b> (Database Directory)
Application Program	<b>APPLCTN</b> (DB/TM, DBCTL, DCCTL)	<b>PDIR</b> (Program Directory)
Transaction	<b>TRANSACT</b> (DB/TM, DCCTL)	<b>SMB</b> (Scheduler Message Block)
Routing Code	<b>RTCODE</b> (DB/TM, DCCTL)	<b>RTCE</b> (Routing Code Table Entry)



# History of IMS Resource Definition – Before DRD

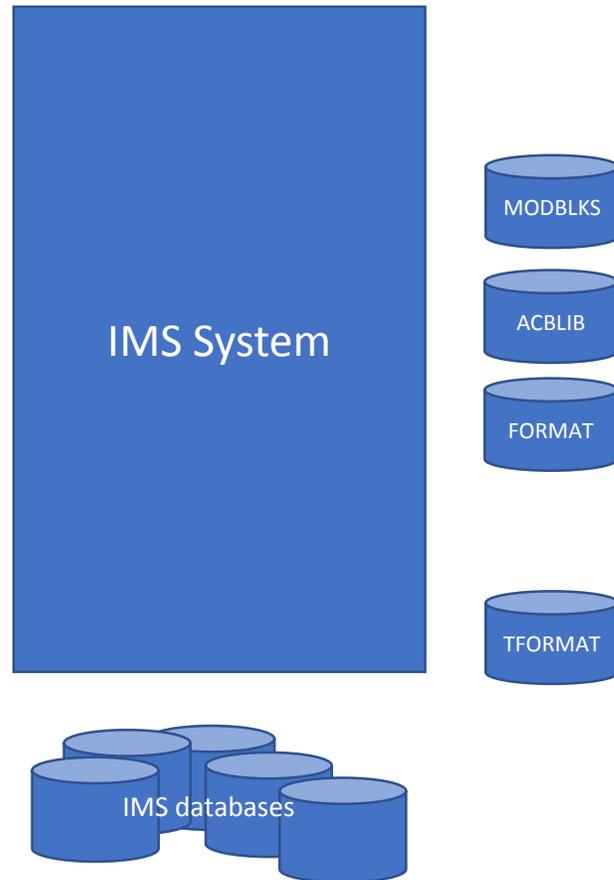
(Source: <https://www.slideshare.net/IBMIMS/introducing-the-ims-repository-66684752>)

- IMS SYSGENs (IMSGEN)  
...since around Day 1 of IMS
  - Resources defined by coding assembler macros and then running jobs (Stage 1 and Stage 2) to assemble and link-edit the macros into IMS RESLIB
  - Cold start of IMS required to take effect
- Online Change (OLC) for MODBLKs, ACBLIB, MFS FORMAT libraries  
...IMS V3.1 or 1.3?
  - MODBLKs for DATABASE, APPLCTN, TRANSACT, and RTCODE resources



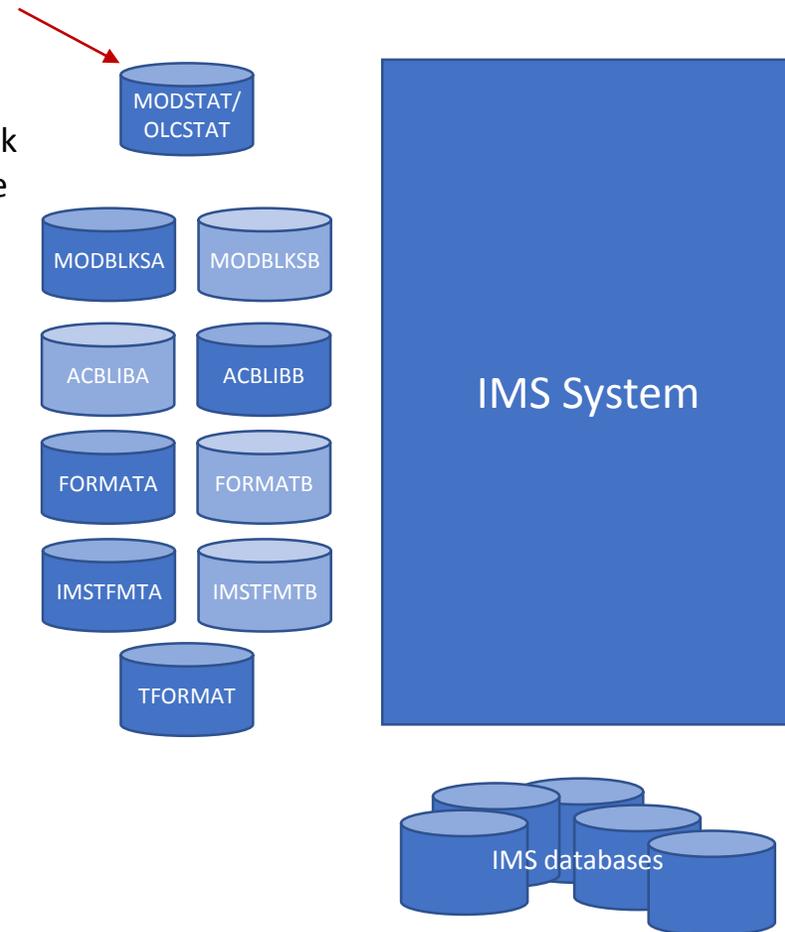
# IMS Resource Libraries and Online Change (OLC)

## An IMS system without OLC or Catalog



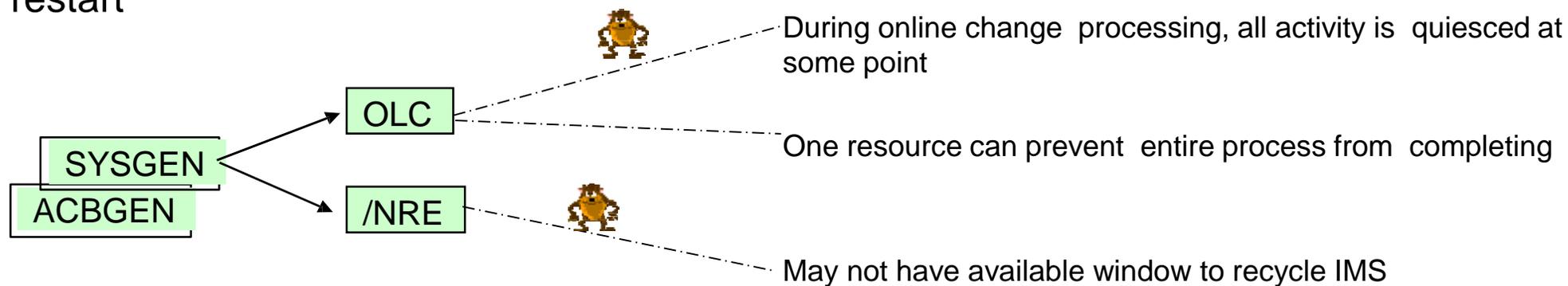
## An IMS system with OLC enabled and without Catalog

- The MODSTAT data set contains online change status for an IMS that has local online change enabled. The data set keeps track of the following libraries that are active: ACBLIBA/ACBLIBB, FORMATA/FORMATB, and MODBLKSA/MODBLKSB.
- The OLCSTAT data set contains online change status for an IMSplex that has global online change enabled. The OLCSTAT data set keeps track of the libraries that are active for each member IMS of the IMSplex.



# Modifying Resource Definitions without DRD

- To add, change, or delete MODBLKS resources in a running IMS system requires:
  - MODBLKS Sysgen
  - ACBGEN - if database (DDIR) or program (PDIR) change
  - Way to bring the new MODBLKS resources online
    - Online change
    - IMS restart

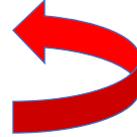


- Process is more complex with multiple IMS's running in an IMSplex
  - Coordinated online change || multiple coordinated system restarts



# Commands to Perform Online Change

- Update 'inactive' datasets
  - SYSGEN
  - ACBGEN
- Initiate online change
  - Issue /MODIFY PREPARE MODBLKS command
- View/Stop resources
  - Issue /DIS MODIFY ALL command
  - Issue /STO xxxx commands
- Commit changes
  - Issue /MODIFY COMMIT
- Start any resources that were stopped
  - Issue /STA xxxx commands



# Tooling to “Enable” Online Change

- Initial tool generally available latter half 1980’s
- Enabled online change in a single step process without required cold start
  - ISPF and batch interfaces
- Functionality to help manage IMS resources:
  - View Resource Definitions
  - See Stage 1 macros
  - Resource Update Lists
    - Define set of incremental changes to sysgen
  - Test proposed changes prior to activating
  - Report on sysgen changes
  - Reverse MODBLKS to recreate IMS sysgen source
  - FASTGEN
  - IMS Command Interface



# Online Change Assist with MODBLKS

- Run Verification
- Issue any 'Before' commands
- Copy Active MODBLKS member to getmained storage
- Update the getmained storage by Resource Update List
- Recreate Inactive MODBLKS member based on the getmained storage
- Stop Resource if needed
- Update IMS incore blocks
- Issue /MODIFY PREPARE MODBLKS command
- Issue /DIS MODIFY ALL command
- Issue /MODIFY COMMIT
- Reload any modules from ACBLIB
- Start Resource if needed
- Issue any 'After' commands
- If any errors, all changes backed out and resources restarted



# Dynamic Resource Definition (DRD)

- OBJECTIVE: Improve availability of the IMS online environment
  - Allow user to dynamically define and enable MODBLKS resource definitions
    - Databases
    - Programs
    - Transactions
    - Routing Codes
  - Allow user to dynamically UPDATE MSC definitions
  - No requirement for IMS Sysgen + IMS restart/online change
  - Result: reduced resource unavailability
  - MODBLKS online change process not required
    - /MODIFY PREPARE MODBLKS
    - /DISPLAY MODBLKS ALL
    - /MODIFY COMMIT



# DRD Components

- Type-2 commands: CREATE, DELETE, UPDATE
- Set of *resource definition data sets* (RDDS)
  - Contains statically (SYSGEN) and dynamically created definitions
    - Resource definitions
    - Model descriptors
- New function to automatically IMPORT and EXPORT resource definitions from/to a resource definition data set
- Enhanced Type-2 command to QUERY the attributes of defined IMS resources and descriptors
- Enhanced DFSINSX0 (Destination Creation) user exit to dynamically create program and transaction resources

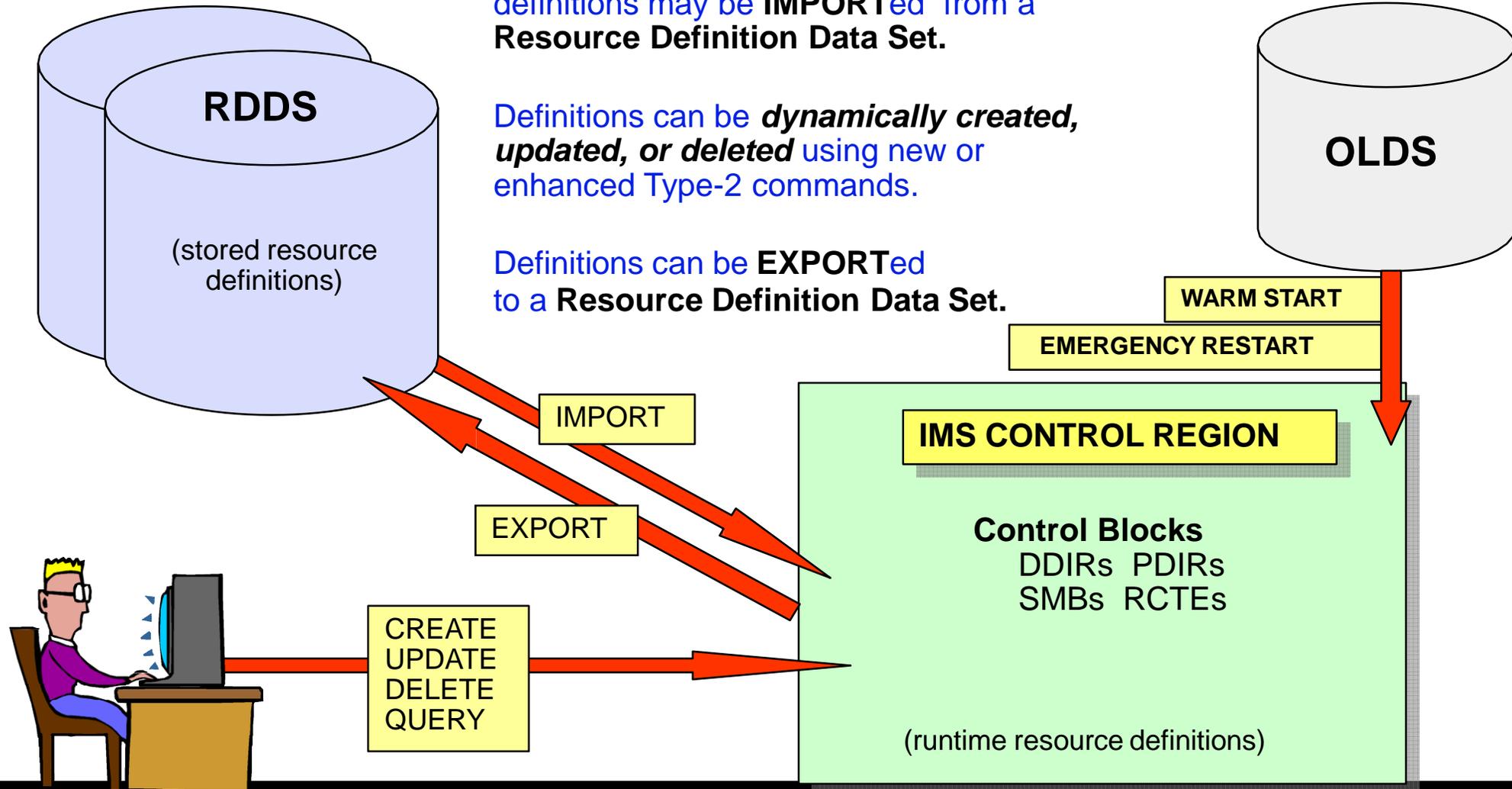


# Modifying MODBLKS Resources with DRD

During IMS cold start processing, resource definitions may be **IMPORT**ed from a **Resource Definition Data Set**.

Definitions can be *dynamically created, updated, or deleted* using new or enhanced Type-2 commands.

Definitions can be **EXPORT**ed to a **Resource Definition Data Set**.



# MODBLKS Resources Loading During IMS Restart

- When DRD is not enabled, control blocks are loaded from:
  - MODBLKS library (cold start)
  - Prior system log (warm or emergency restart)
- When DRD is enabled, control blocks are loaded from:
  - RDDS (cold start)
  - Prior system log (warm or emergency restart)
  - Not at all



# DRD Environment

- DRD requires IMS V10 with:
  - Common Service Layer with Structured Call Interface and Operations Manager (CSL with SCI/OM)
    - Resource Manager is not required
  - IMS PROCLIB members with DRD enablement parameters
    - DFSDFxxx (new in IMS V10)
    - DFSCGxxx (CSL parameters - can be replaced by section of DFSDFxxx)
- Data sets to hold resource definitions (RDDSSs)
- Entry point for DRD commands
  - TSO Single Point Of Control (SPOC) or other OM interface



# DRD Environment

- All IMS online environments supported
  - Including data sharing and shared queues
- No longer required
  - MODBLKS SYSGEN process
  - MODBLKS online change process
  - MODBLKS staging, active, and inactive data sets

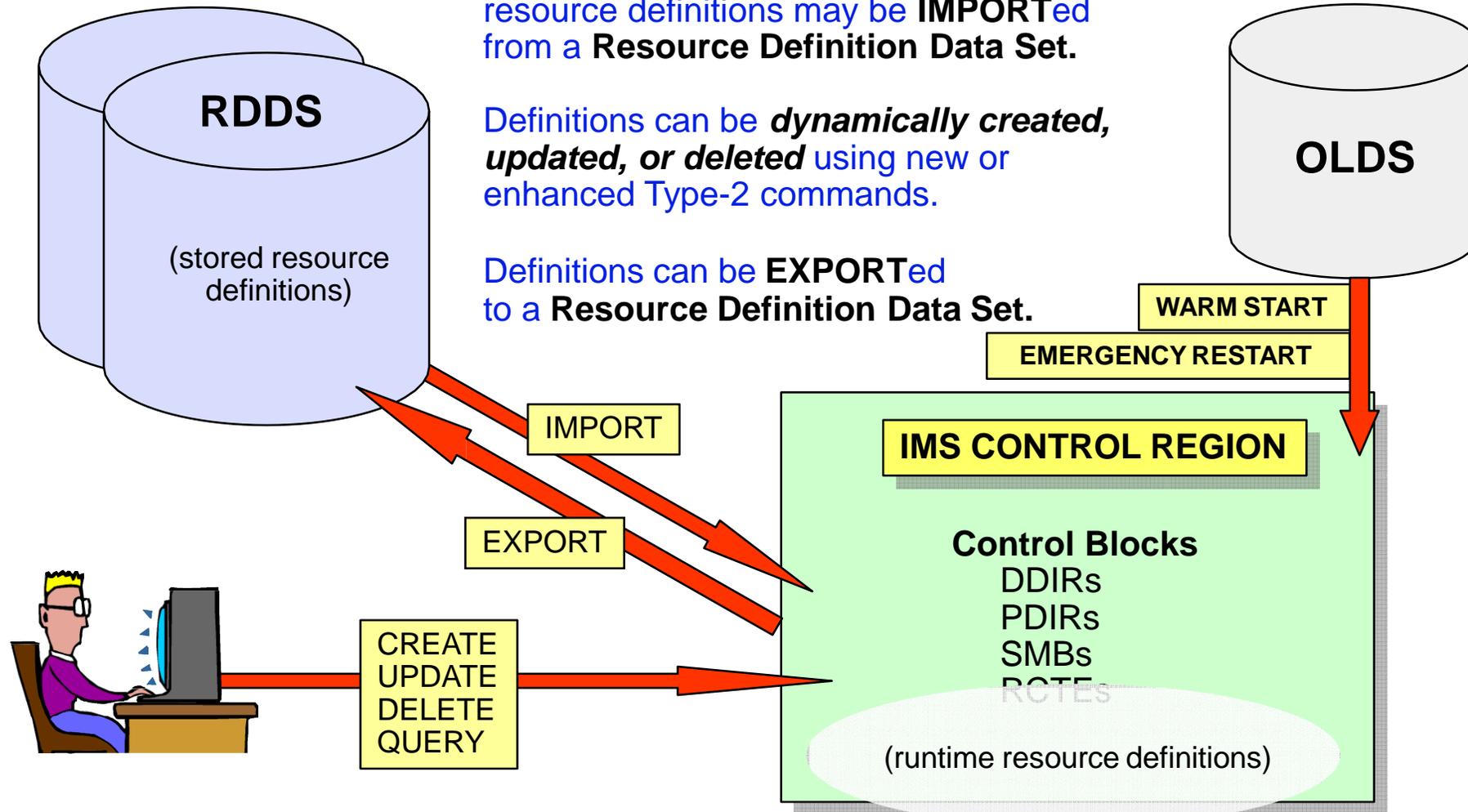


# Modifying MODBLKS Resources with DRD

During IMS cold start processing, resource definitions may be **IMPORTed** from a **Resource Definition Data Set**.

Definitions can be *dynamically created, updated, or deleted* using new or enhanced Type-2 commands.

Definitions can be **EXPORTed** to a **Resource Definition Data Set**.



# History of IMS Resource Definition – DRD and RDDS

(Source: <https://www.slideshare.net/IBMIMS/introducing-the-ims-repository-66684752>)

## • An issue in OLC

- One active resource could prevent OLC from completing

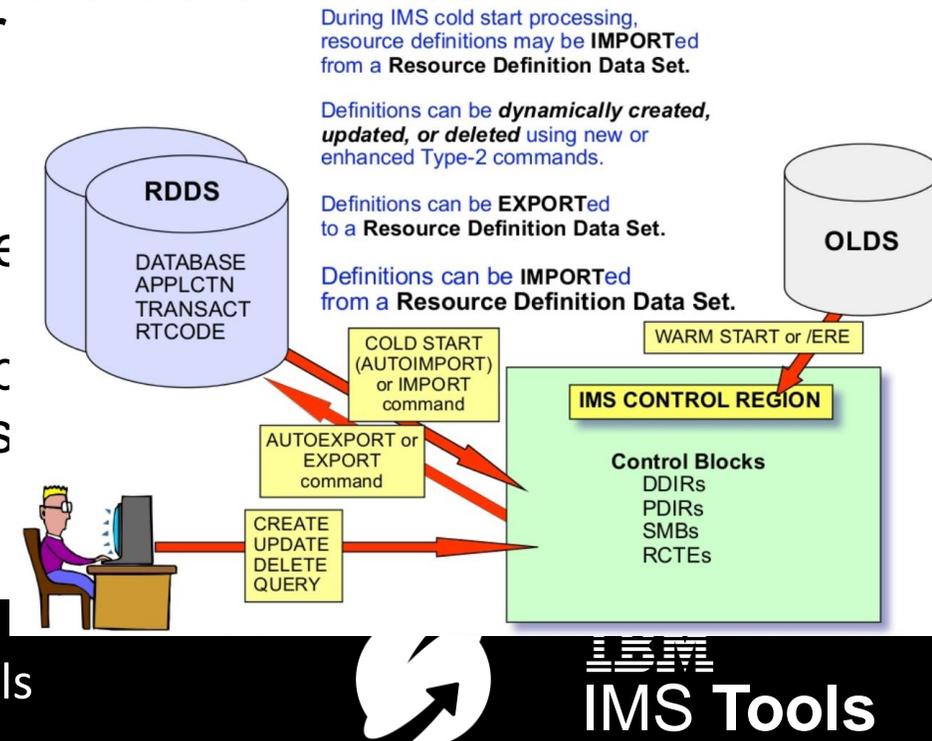
## • **Solution:** Dynamic Resource Definition (DRD) with RDDS

- IMS V10 or later

## • RDDS (Resource Definition Data Set) is a set of data sets that contain MODBLKS resource and descriptor definitions for

## • Objectives of DRD

- Improves availability of the IMS online environment
- No need for IMS SYSGEN or MODBLKS online change
- Suppression of MODBLKS data sets
- Shared resources across all IMS members in a single p
- Reduction of startup time of IMS (up to several minutes)
- Fast execution of dynamic resource change requests



# DRD objectives were not met with RDDS

(Source: <https://slideplayer.com/slide/9014068/>)

- Issue: Synchronization and backout processing in an IMSplex

- Changes to resources are not coordinated IMSplex wide
  - Active on all IMS members or none
  - When error: rollback IMSplex wide
  - An IMS member down in the IMSplex @ DRD request => manual operation
  - Group of related resources as a whole

- Solution: DRD with IMS Resource (IMSRSC) Repository

- Since IMS V12



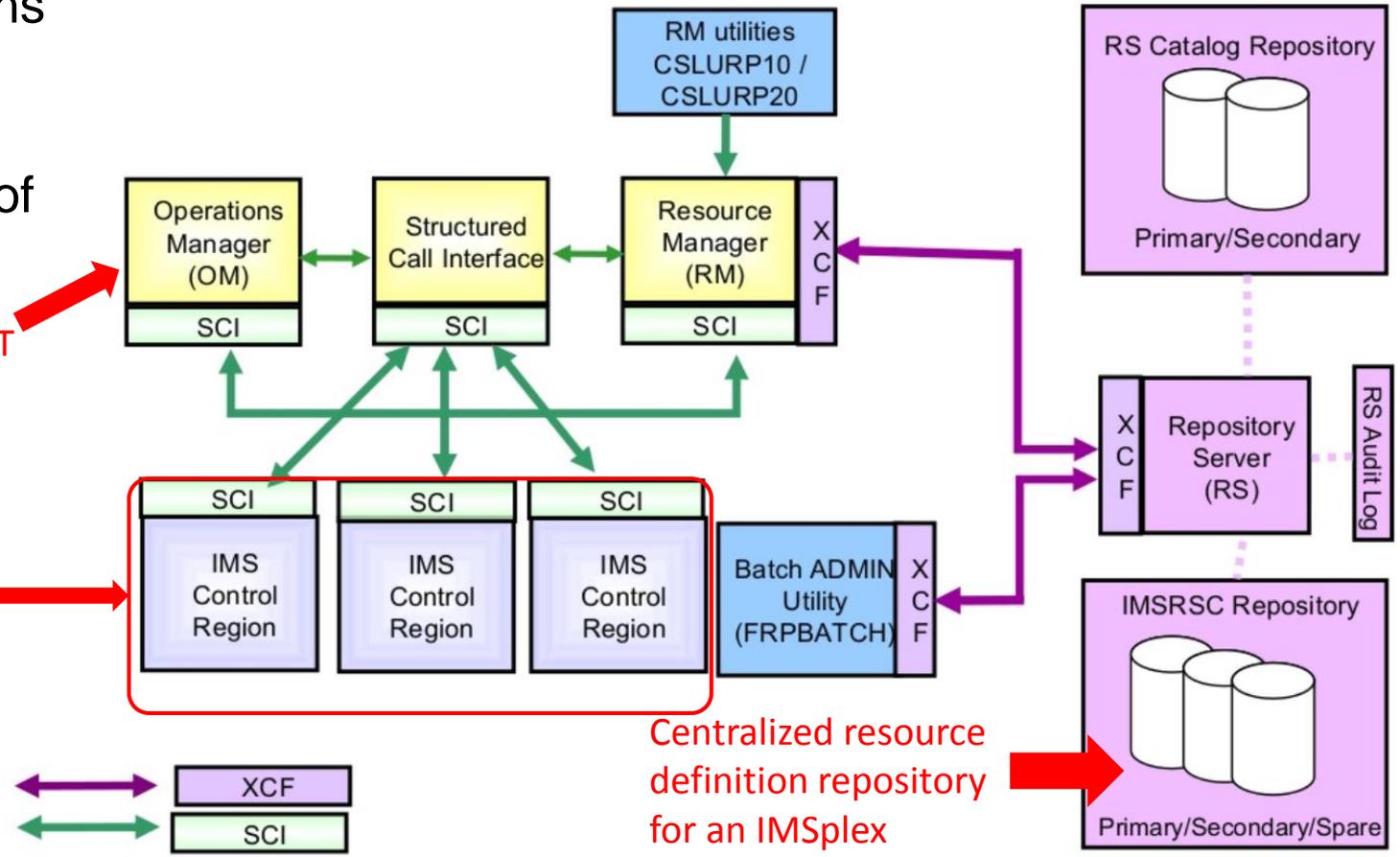
# DRD with IMSRSC Repository

(Ref.: <https://www.slideshare.net/IBMIMS/ims-v14-repository-enhancements-and-repository-overview>)

- The IMSRSC repository function is a centralized method for storing and retrieving resource definitions in an IMSplex
  - Centralizes all resource definitions in one place in an IMSplex
  - Replaces RDDS at each IMS
  - Eliminates manual coordination of multiple RDDSs in an IMSplex

Resource changes and EXPORT/IMPORT to IMSRSC repository with IMS Type-2 commands

IMS control regions in an IMSplex

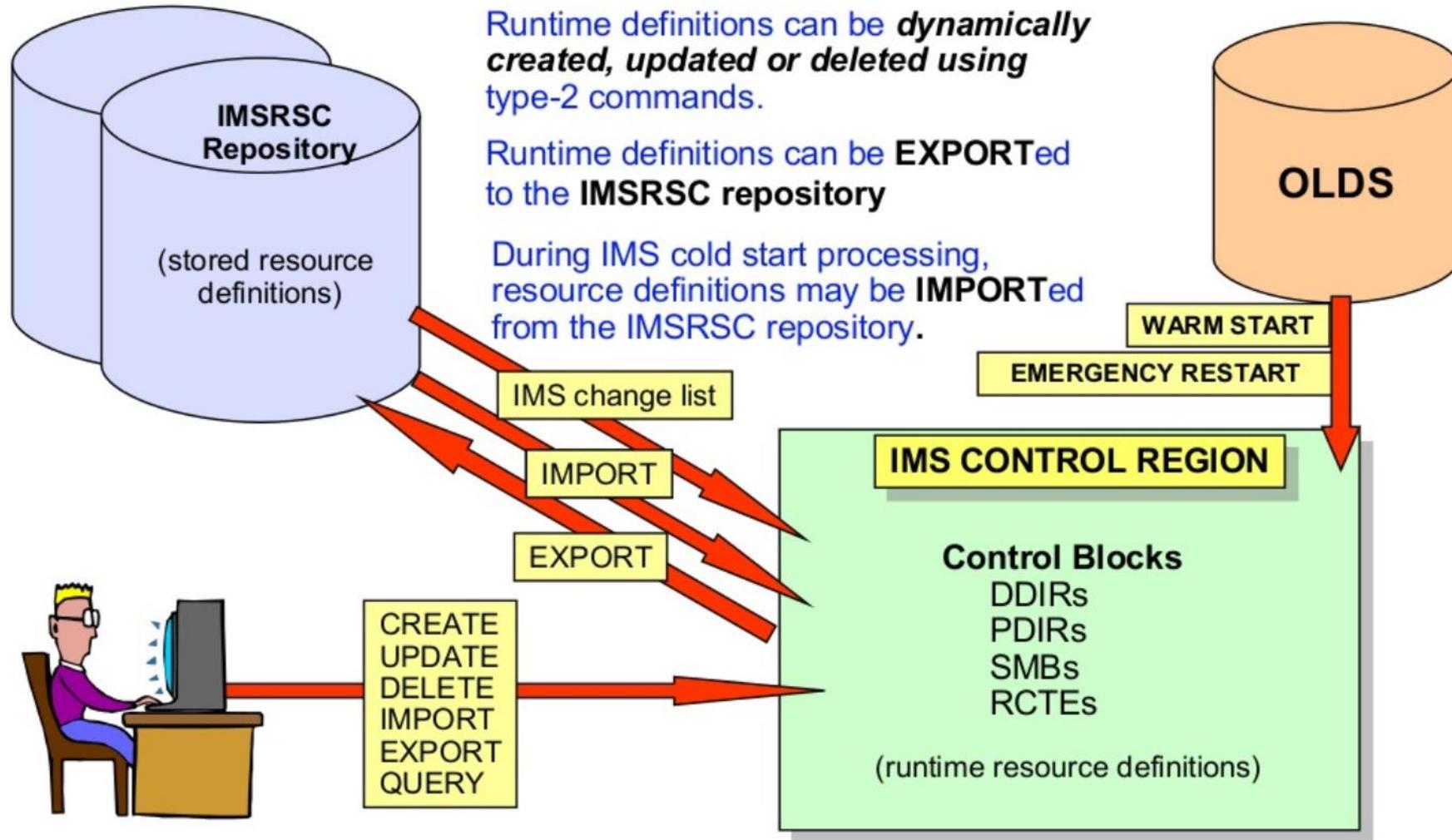


Centralized resource definition repository for an IMSplex

IMS Tools

# IMS Resource Definition with IMSRSC Repository

(Source: <https://www.slideshare.net/IBMIMS/introducing-the-ims-repository-66684752>)



# Descriptors

- A descriptor is a model for defining (creating) a resource or another descriptor
  - Establishes *defaults* for attributes not set in the CREATE command
- IMS-defined descriptors
  - Provided with the IMS product:
    - DFSDSDB1 (database)
    - DFSDSPG1 (program)
    - DFSDSTR1 (transaction)
    - DBFDSRT1 (routing code)
- User-defined descriptors
  - Created or updated by the user
- Current system default descriptor
  - Each resource type will have one *default descriptor*
    - IMS-defined or user-defined with DEFAULT(Y)



# Recoverability

- Resource and descriptor definitions
  - Exist for the life of IMS or until deleted
  - Are recovered from logs across warm and emergency restart
    - Definitions are logged when created, updated, or deleted, and at system checkpoint time
  - Are lost across cold start unless both:
    - Previously exported to RDDS/Repository
    - Imported during next cold start



# Commands Used in Dynamic Resource Definition

- Type-2 commands entered through OM interface

Command	Short Form	Purpose
CREATE	CRE	Create resource or descriptor definition
DELETE	DEL	Delete resource or descriptor definition
UPDATE	UPD	Update attributes of resource or descriptor definition Update status of resource
QUERY	QRY	Query attributes of resource or descriptor definition Query status of resource



# Reference Table for Resources & Keywords

Resource Type	SYSGEN Macro	IMS ControlBlock	Resource Keyword	Descriptor Keyword
Database	DATABASE	DDIR	DB	DBDESC
Program/PSB	APPLCTN	PDIR	PGM	PGMDESC
Transaction	TRANSACT	SMB	TRAN	TRANDESC
Routing Code	RTCODE	RCTE	RTC	RTCDESC

- Examples
  - CREATE PGM...
  - DELETE TRANDESC...
  - UPDATE TRAN...



# CREATE Command - Syntax

```
CREATE rsc-type | desc-type  
NAME (name1 , name2 , ...)  
LIKE (RSC (rsc-name) ) | LIKE (DESC (desc-name) )  
SET (attr1 (val1) , attr2 (val2) , ...)  
DEFAULT (Y) <<< valid for descriptors only
```

- **rsc-type | desc-type** = resource or descriptor type (e.g., PGM, PGMDESC)
- **NAME()** = resource or descriptor names; they will all have the same attributes; wild cards not supported
- **LIKE()** = resource or descriptor name to use as model
- **SET()** = attribute names and values
- **DEFAULT(Y)** – valid for descriptors only
  - – Y cause this descriptor to become current system default descriptor



# CREATE Command - Setting Attributes

- All attribute values are set either explicitly by SET command keyword or by default
- Hierarchy of attribute setting:
  - SET command keyword – takes precedence over all others
  - LIKE command keyword
    - Attributes not SET are obtained from the specified model
    - Can be LIKE DESC or LIKE RSC
    - Takes precedence over current default descriptor
  - If LIKE command keyword not specified
    - Attributes not SET will be gotten from current system default descriptor, which can be either:
      - DFSDSxx1
      - User descriptor created with DEFAULT(Y)



# CREATE Command

- DFSDSTR1 is IMS default descriptor for transactions
- All attributes defined; for example
  - PARLIM = 65535
  - – PLCT = 65535
  - PLCTTIME = 6553500
  - MAXRGN = 0
- Can change these values (and others) in user-defined descriptor
- Examples

```
CRE TRANDESC NAME (ACCTMODL)
      SET (PARLIM (5) , PLCT (20) , PLCTTIME (2) , MAXRGN (5) )

CRE TRAN NAME (ACCTUPDT) LIKE (DESC (ACCTMODL)
      SET (MAXRGN (8) , PGM (ACCTPGM) )

CRE TRAN NAME (ACCTDLET) LIKE (RSC (ACCTUPDT) )
```



# DELETE/UPDATE Commands

- To delete/update a resource
  - Resource cannot be “in use”
  - Recommendation: stop resource before attempting to delete/update
- QUERY command enhanced to display resource work status
  - Show work that might cause DELETE/UPDATE command (or OLC) to fail
- Examples

```
QRY PGM NAME (ACCTPGM) SHOW (WORK | TRAN)
/STO PGM ACCTPGM
DEL PGM NAME (ACCTPGM)

QRY TRAN NAME (ACCTDLET, ACCTADD, ACCTU*) SHOW (WORK)
/STO PGM ALL
DEL TRAN NAME (ACCTDLET, ACCTADD, ACCTU*)
```



# QUERY Command for Runtime Resource Definitions

- Examples

```
QRY DB NAME (ACCTMSTR) SHOW (WORK | PGM)
```

```
QRY PGM NAME (ACCTPGM) SHOW (WORK | TRAN | RTC) QRY
```

```
TRAN NAME (ACCT*) SHOW (WORK)
```

```
QRY RTC NAME (ACCT*) SHOW (WORK | REGION)
```



# Use Case Scenarios: DRD with IMSRSC Repository

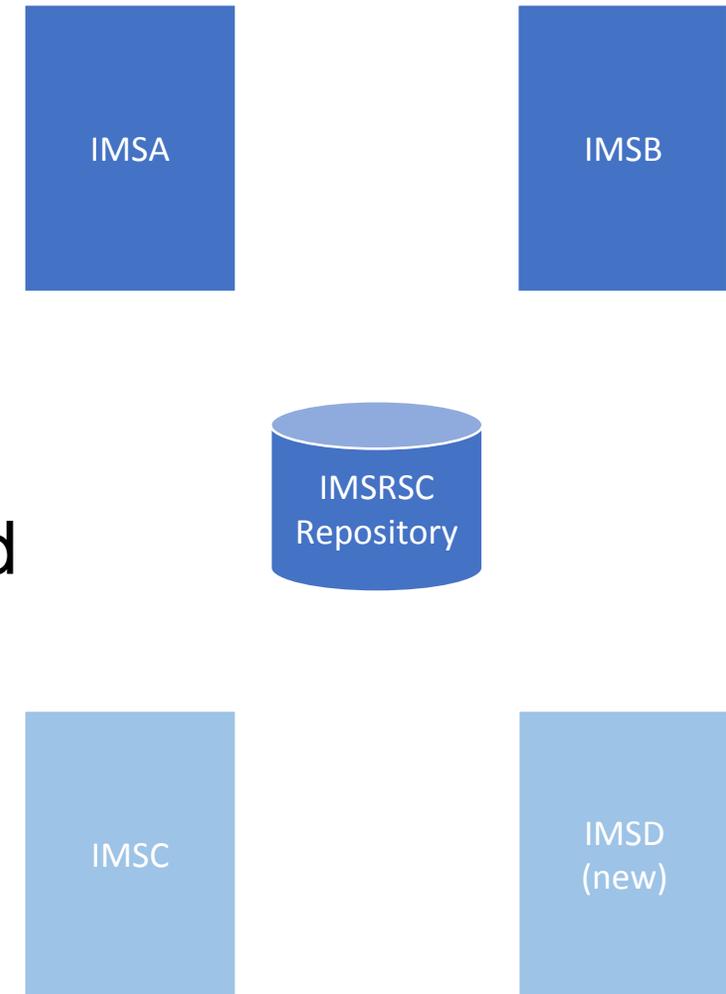
- Scenario #1: Creating/updating resources in a cloned environment
- Scenario #2: Updating resources in a non-cloned environment
- Scenario #3: Deleting resources for IMS systems in an IMSplex

(Source: <https://www.slideshare.net/IBMIMS/ims-v14-repository-enhancements-and-repository-overview>)



# #1: Creating/Updating Resources in a Cloned Environment

- 3-way IMSplex with cloned systems IMSA, IMSB and IMSC is defined to the IMSRSC repository
  - IMSA and IMSB are active
  - IMSC is inactive
  - IMSD will be added in the future
- Program named PGMCAR and transaction named TRANCAR need to be created in the IMSplex



# Scenario #1 (cont.)

- To create the runtime resources at test system IMSA, issue the following commands:

```
CREATE PGM NAME(PGMCAR)          routed to IMSA
CREATE TRAN NAME(TRANCAR) SET(PGM(PGMCAR)) routed to IMSA
```

- To write the resource definitions to the repository for all IMS systems, issue the following command (this is not required if IMS V14+ AUTOEXPORT feature is enabled):

```
EXPORT DEFN TARGET(REPO) OPTION(CHANGESONLY) SET(IMSID(*))
```

- Newly created program PGMCAR and transaction TRANCAR are added to the IMS resource lists for IMSA, IMSB and IMSC and the resource definitions for PGMCAR and TRANCAR are written to the IMSRSC repository

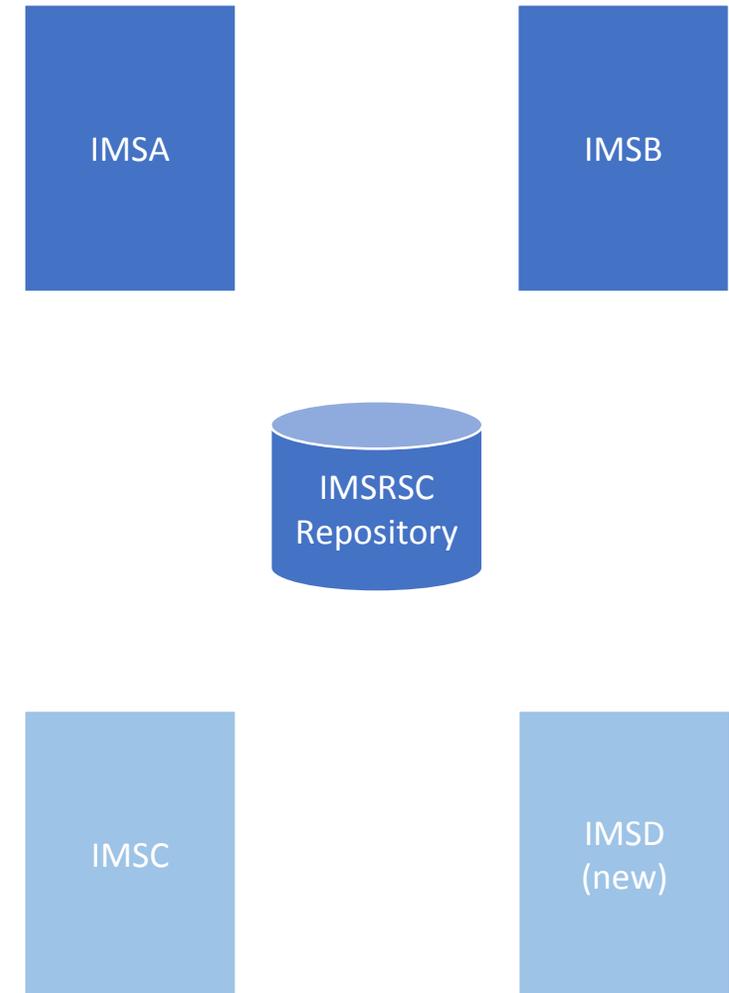
- To create the resource definitions to the repository for a new IMS system IMSD to be defined to the repository, issue:

```
EXPORT DEFN TARGET(REPO) NAME(*) SET(IMSID(IMSD)) ROUTE(IMSA)
```

- To activate the resource definitions in the IMSplex, issue:

```
IMPORT DEFN SOURCE(REPO) NAME(PGMCAR,TRANCAR) SCOPE(ALL) OPTION(UPDATE)
```

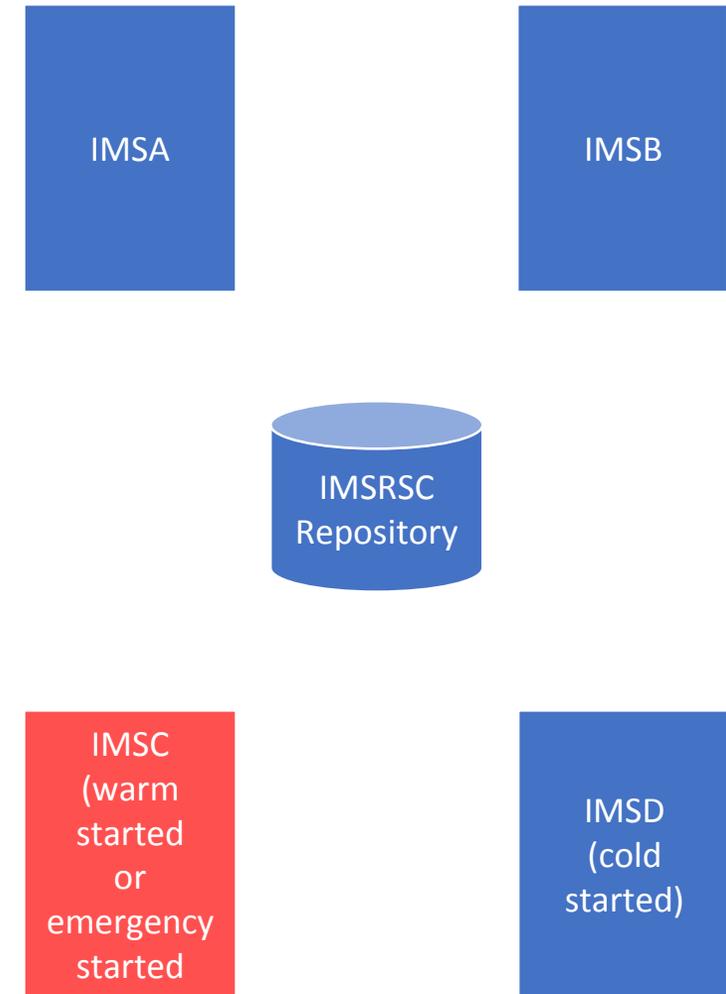
- As IMSA and IMSB are active, the IMPORT command
  - Updates PGMCAR and TRANCAR in IMSA
  - Creates PGMCAR and TRANCAR in IMSB
- How about IMSC and IMSD?



# Scenario #1 (cont.)

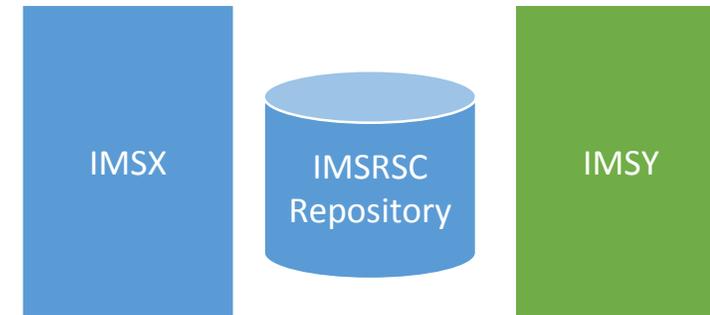
**IMPORT DEFN SOURCE(REPO) NAME(PGMCAR,TRANCAR)  
SCOPE(ALL) OPTION(UPDATE)**

- An **IMS change list** is created in the IMSRSC repository by the command master IMS during IMPORT command processing for IMSC and IMSD with names PGMCAR and TRANCAR
- IMSC at the end of restart (warm or emergency) reads the change list and the resource names PGMCAR and TRANCAR are automatically imported from the repository
  - Its change list is deleted at coldstart
- IMSD as is a new IMS at cold start will read all the resource definitions from the repository



## #2: Updating Resources in a Non-Cloned Environment

- 2-way IMSplex with non-cloned systems IMSX and IMSY is defined to the IMSRSC repository
  - IMSX and IMSY are defined with
    - AUTOEXPORT=AUTO (explicitly specified)
    - AUTOEXPORT\_IMSID=THIS\_IMS as they are non-cloned
  - IMSX and IMSY are active
- An example: Route commands to individual IMS systems
  1. Route the following commands to IMSX only
    - UPDATE PGM NAME(PGMAAA) SET(TRANSTAT(Y))
  2. Route the following commands to IMSY only
    - UPDATE PGM NAME(PGMDDD) SET(SCHDTYPE(PARALLEL))
  3. Route the /CHE command to IMSX and IMSY takes a checkpoint and trigger AUTOEXPORT to IMSX and IMSY:
    - IMSX will write the updated changes for program PGMAAA for IMSX only
    - IMSY will write the updated changes for program PGMDDD for IMSY only



# #3: Deleting resources for IMS systems in an IMSplex

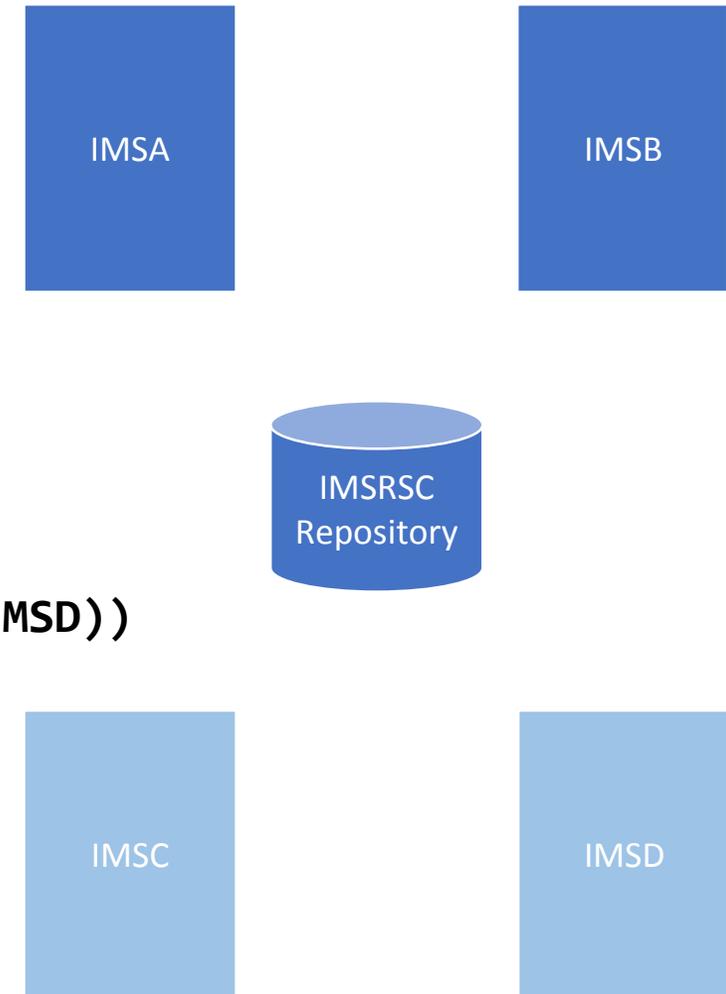
- Assumptions
  - 4-way IMSplex with 4 systems
    - IMSA, IMSB, IMSC, and IMSD
  - They are defined to an IMSRSC repository
  - IMSA and IMSB are active
  - IMSC and IMSD are inactive
- Route the following commands to all IMS systems that are active:

**DELETE PGM NAME(PGMDDD)**

- It will delete PGMDDD if defined

**DELETE DEFN TARGET(REPO) TYPE(PGM) NAME(PGMDDD) FOR(IMSID(IMSD))**

- The command can be routed to any IMS, say IMSA, to be deleted from the repository for another IMS
- The resource PGMDDD does not have to be defined at IMS that processes the DELETE DEFN command



# Tooling that “Enhances” Online Change

- Limit what attributes, by resource type, a user can update
- Ability to set updated default values for resource attributes that apply to a specific set of users.
- View “runtime resources” or “stored resources”
- Same interface(s) to implement changes
  - MODBLKS, RDDS, Repository
  - ACBLIB or IMS Catalog
- Verification of changes before implementing
- History log
  - View who made what changes when
  - Easy to UNDO changes
- Controls for installing packets of changes
  - Group changes are coordinated among all IMS systems.
  - The internal check is completed on all IMS systems before any actual changes are initiated.
  - A resource update list is installed on all IMS systems in a group, or it is backed out on all IMS systems.



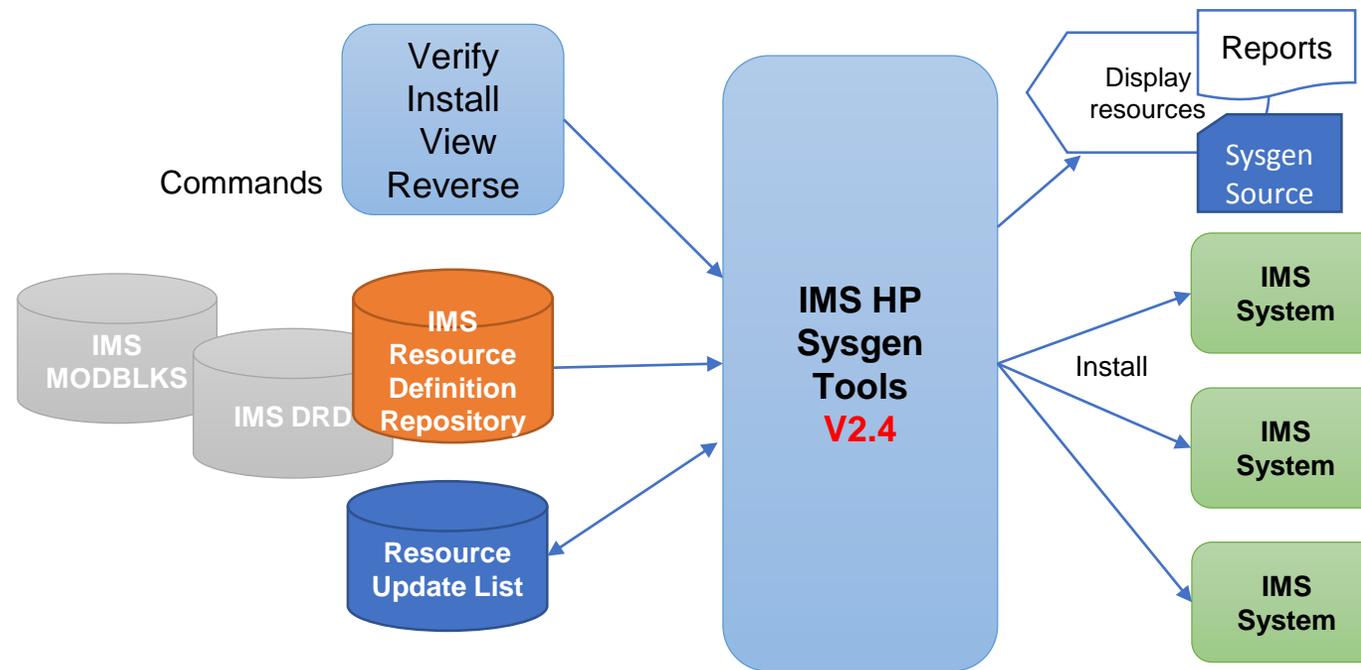
# Online Change Assist with DRD/RDDS

- Run Verification
- Issue any 'Before' commands
- Copy Active RDDS member(s) to getmained storage
- Update the getmained storage by Resource Update List
- Create non-system RDDS from getmained storage
- Stop Resource if needed
- Update IMS incore blocks
- IMPORT DEFN SOURCE(RDDS) RDDSDSN(temp.rddsdsn)
- EXPORT DEFN TARGET(RDDS)
- Reload any modules from ACBLIB
- Start Resource if needed
- Issue any 'After' commands
- If any errors, all changes backed out and resources restarted



# IMS Resource Definition Repository Support

- Displaying current resource location (MODBLKS, DRD: Dynamic Resource Definition or IMSRSC repository)
- Viewing, Editing, Installing IMS resources in the IMSRSC repository
  - IMS Type 2 QUERY / EXPORT command are issued internally
- Generating IMS System Generation Source from the IMSRSC repository



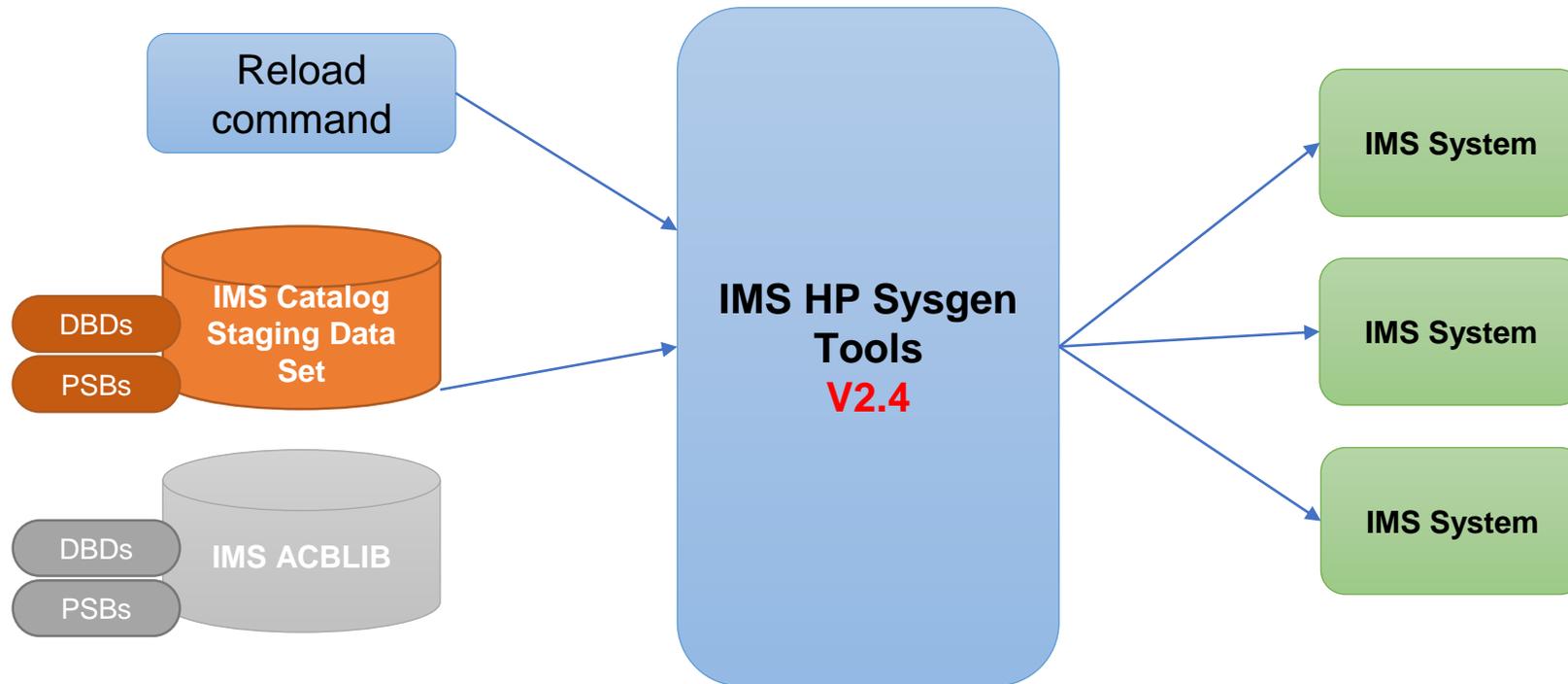
✓ Both ISPF and batch environment are supported  
IMS DRD and HP Sysgen Tools



IBM  
IMS Tools

# IMS Managed ACBs (IMS Catalog) Support

- Reloading DBD and PSB definition in IMS Catalog Staging Data Set to IMS Online System
  - IMS Type 2 IMPORT command are issued internally



✓ Both ISPF and batch environment are supported

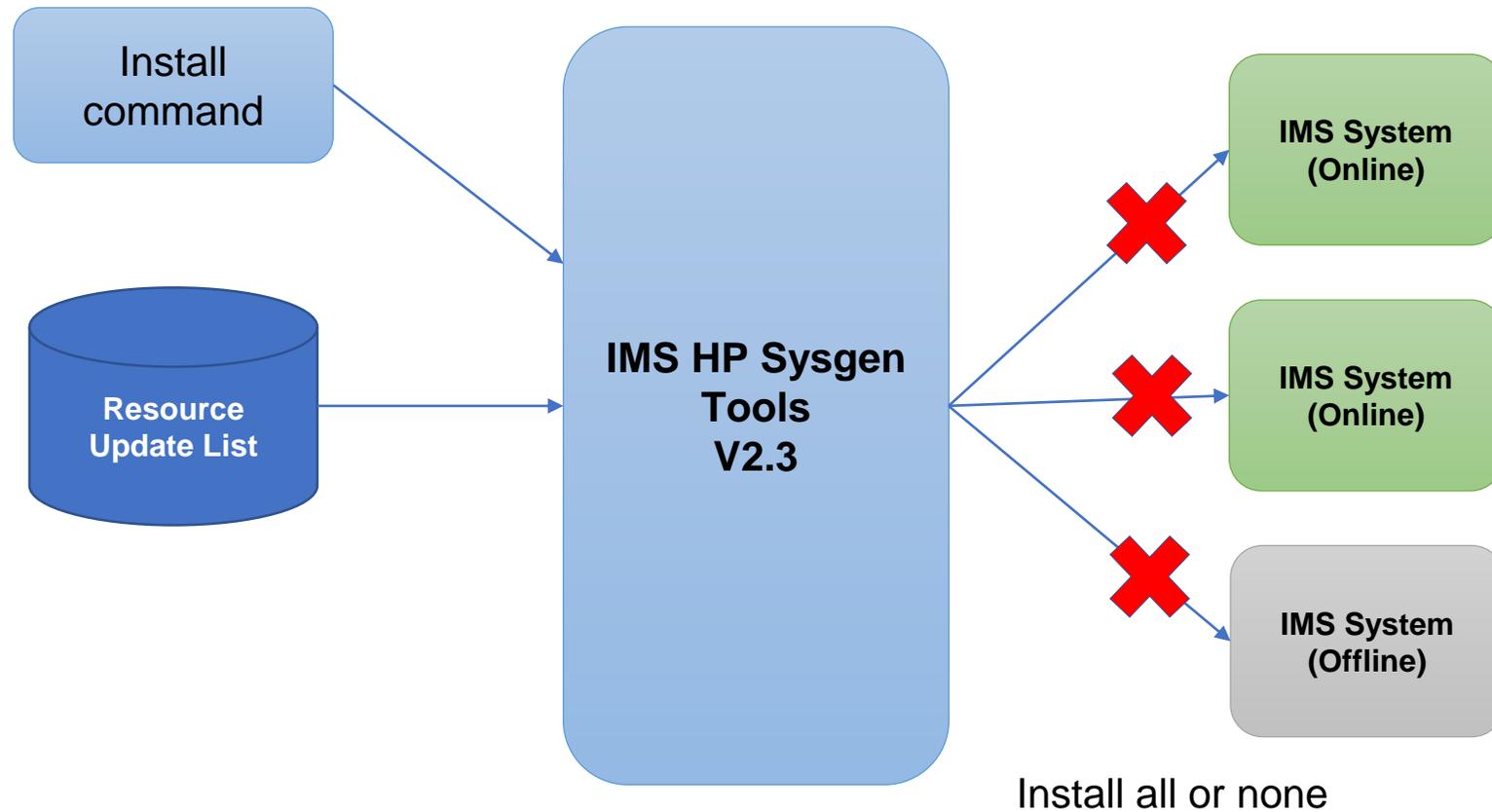
IMS DRD and HP Sysgen Tools



IBM  
IMS Tools

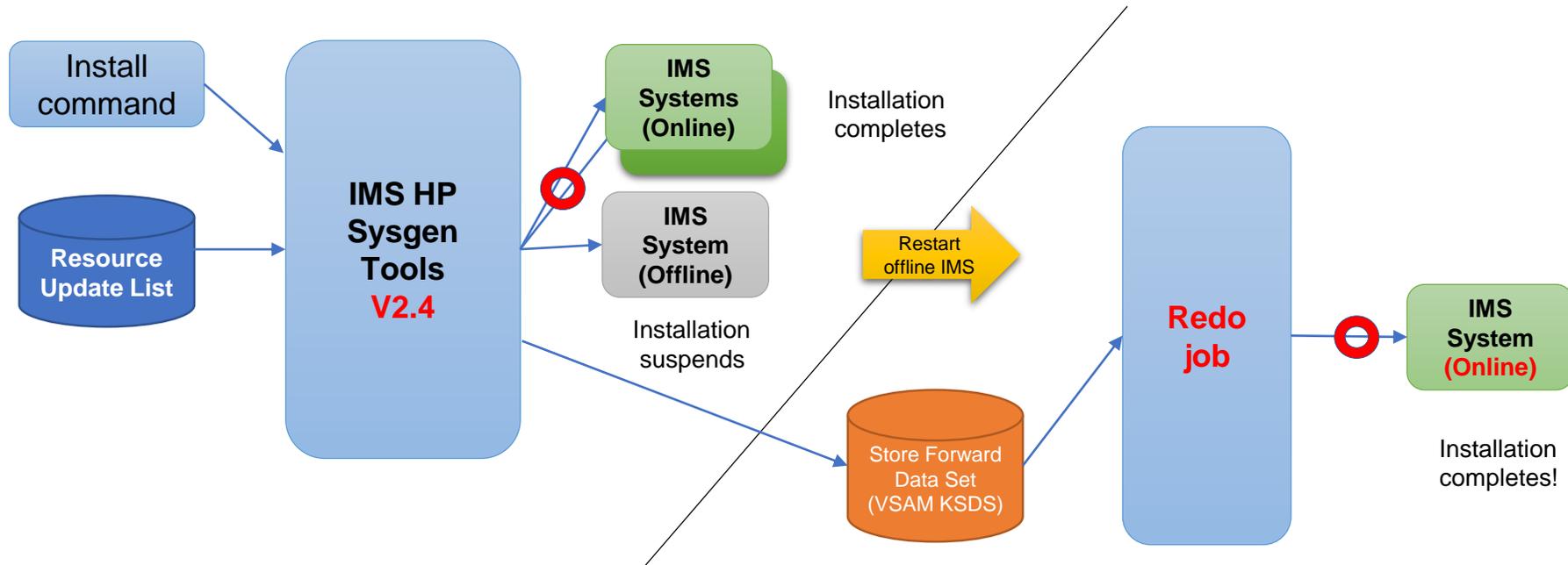
# Installation store/forward functionality

- Handling changes as a packet.
  - Install changes on all IMS systems or none of them



# Installation store/forward functionality

- With HP Sysgen V2.4:
  - Installation for online IMS Systems completes successfully
  - Installation for offline IMS Systems are suspended until next restart
  - When the offline IMS is starting up, “Redo” job automatically starts and suspended resource installation is performed



✓ Both ISPF and batch environment are supported



# Summary

- Dynamic Resource Definition (DRD) improves on the availability of IMS systems when resources need to be managed
- Implementing DRD requires process changes
- Implementing the IMS Catalog requires process changes
- Tooling allows for users to implement DRD and the IMS catalog without imposing process changes



# For More Information

- IMS Tools website  
[www.ibm.com/it-infrastructure/z/ims/tools](http://www.ibm.com/it-infrastructure/z/ims/tools)
- IMS Tools new functions  
[www.ibm.com/support/docview.wss?uid=swg22015506](http://www.ibm.com/support/docview.wss?uid=swg22015506)
- IMS Tools Product Documentation  
[www.ibm.com/support/docview.wss?uid=swg27020942](http://www.ibm.com/support/docview.wss?uid=swg27020942)
- IMS Tools Youtube Playlist  
[www.youtube.com/playlist?list=PLezLS0Tuqb-5DSdF1Locnq5lhTgcX02vf](http://www.youtube.com/playlist?list=PLezLS0Tuqb-5DSdF1Locnq5lhTgcX02vf)
- IMS new functions  
[www.ibm.com/support/knowledgecenter/en/SSEPH2\\_15.1.0/com.ibm.ims15.doc.rpg/ims\\_cd\\_functions.htm](http://www.ibm.com/support/knowledgecenter/en/SSEPH2_15.1.0/com.ibm.ims15.doc.rpg/ims_cd_functions.htm)
- IBM zITSM newsletter (email every 2 months with summary articles and links to more information)  
<http://ibm.biz/zITSMNewsletterSubscribe>
- IMS Tools support for IMS V15  
[www.ibm.com/support/docview.wss?uid=swg22009341](http://www.ibm.com/support/docview.wss?uid=swg22009341)
- IMS Tools support for Managed ACBs  
[www.ibm.com/support/docview.wss?uid=ibm10731745](http://www.ibm.com/support/docview.wss?uid=ibm10731745)
- IMS Tools support for Data Set Encryption  
[www.ibm.com/support/docview.wss?uid=ibm107333513](http://www.ibm.com/support/docview.wss?uid=ibm107333513)



धन्यवाद

Hindi

多謝

Traditional

감사합니다

Korean

Спасибо

Russian

Ndzi khense ngopfu

Tsonga

Gracias

Spanish

Thank You

English

Obrigado

Brazilian Portuguese

شكراً

Arabic

Grazie

Italian

Danke

German

多谢

Simplified Chinese

Merci

French

Ke a leboha

Tswana

நன்றி

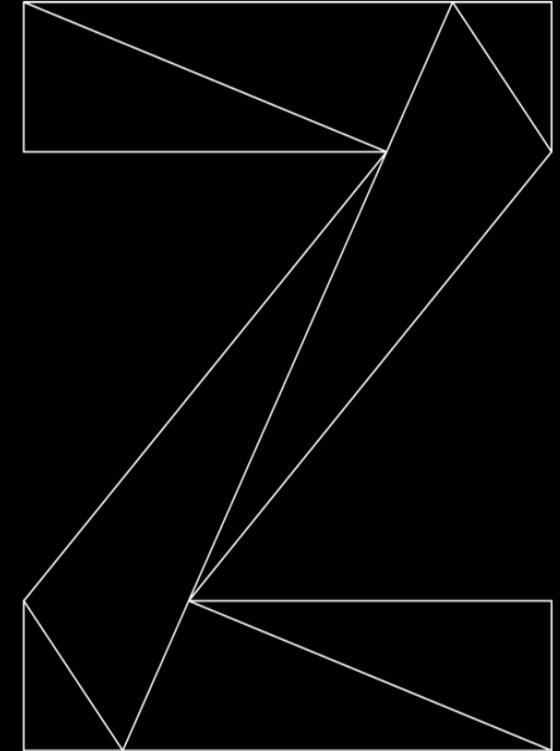
Tamil

ありがとうございました

Japanese

ขอขอบคุณ

Thai



IBM  
IMS Tools