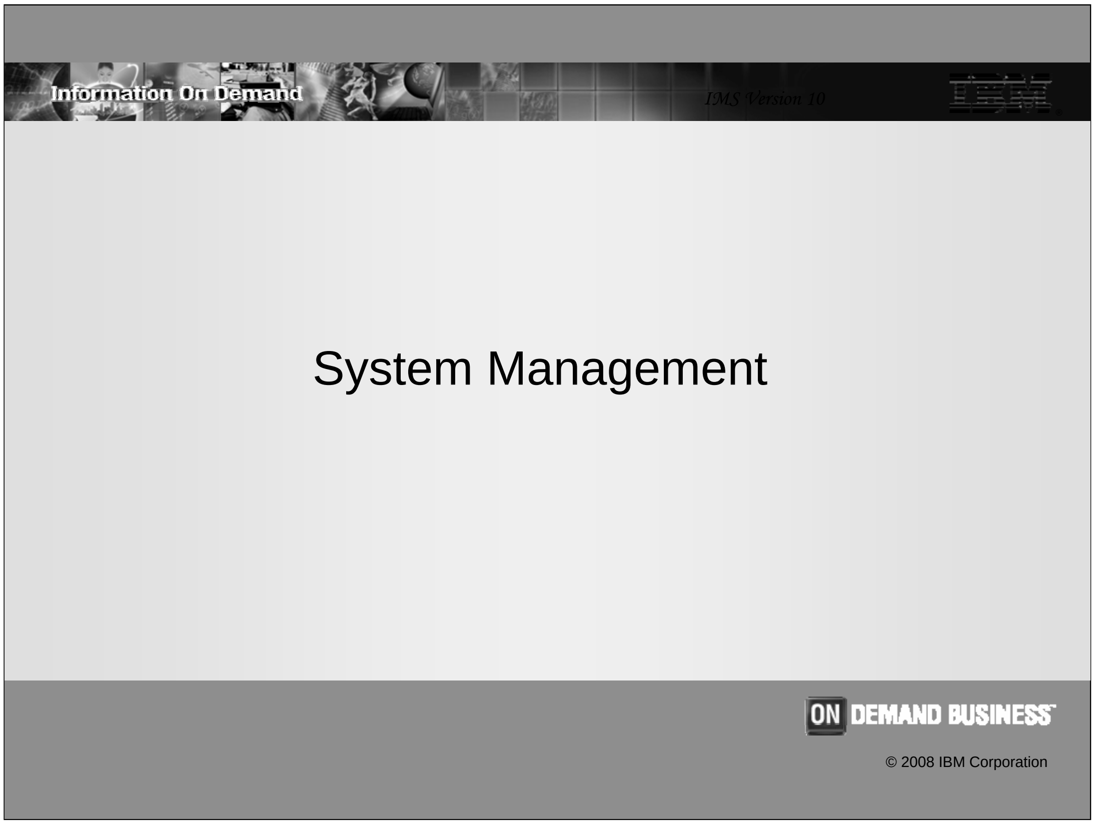
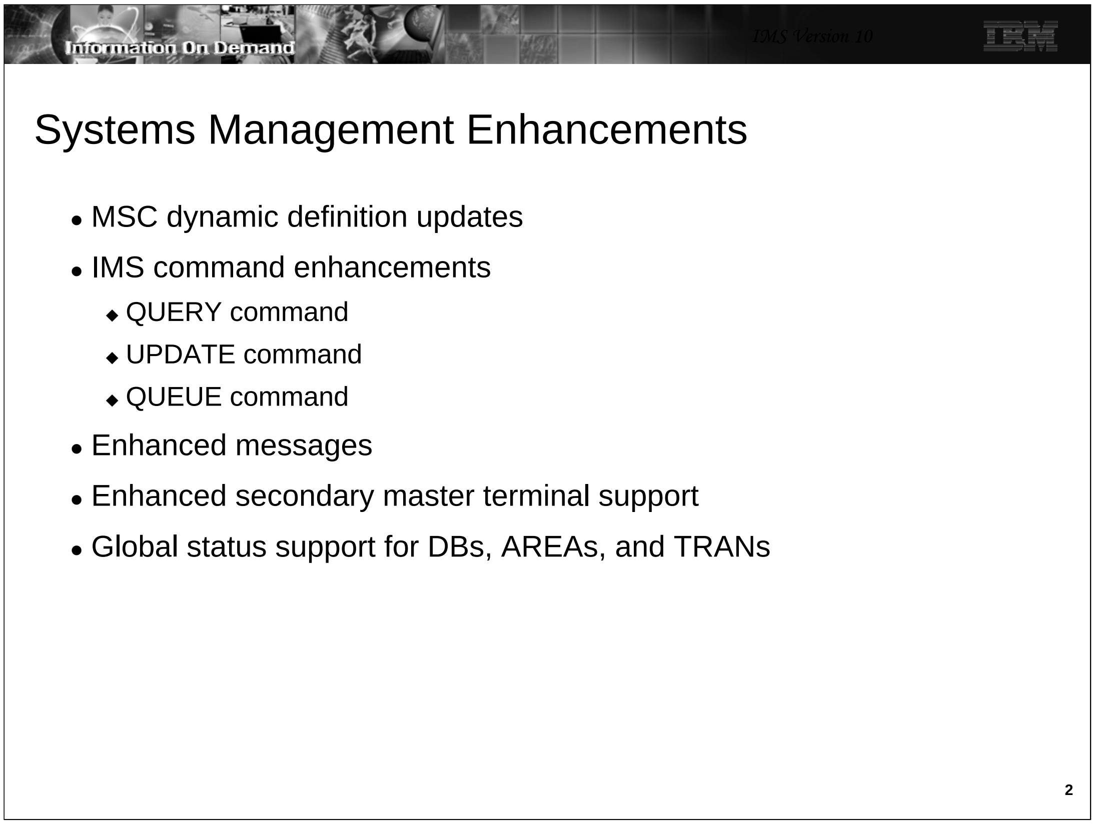


Information On Demand

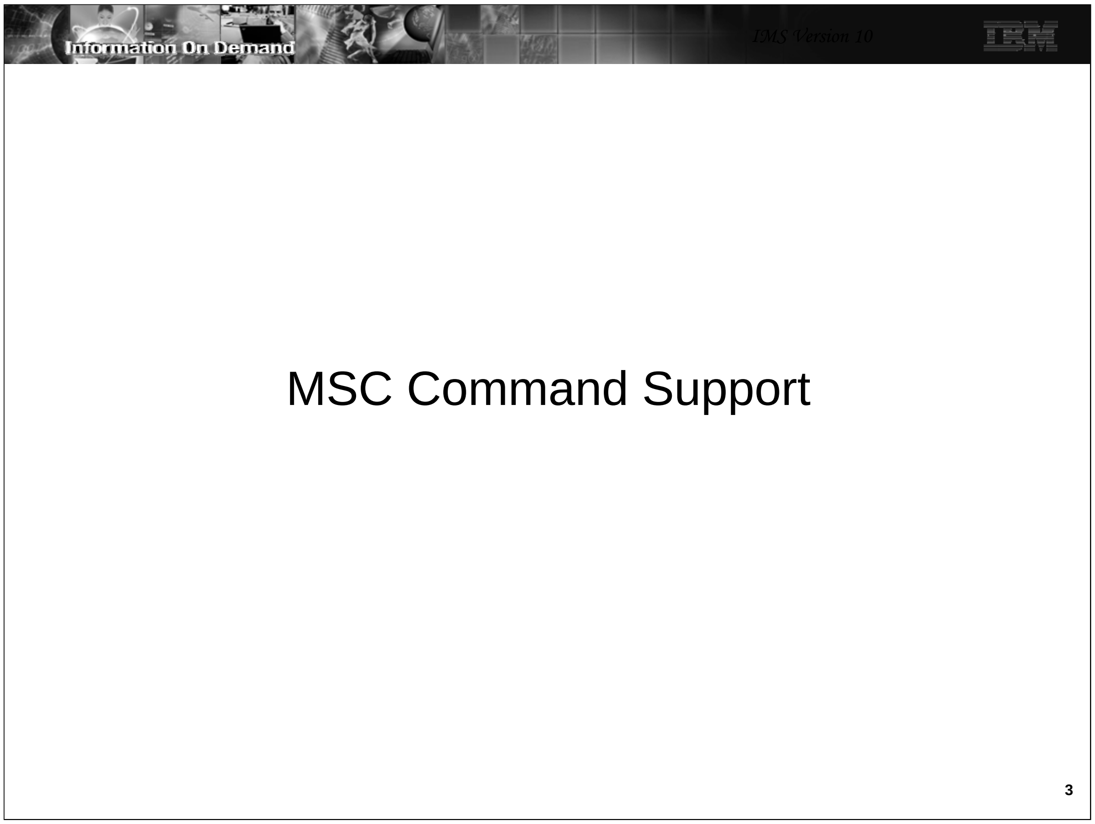
IMS Version 10

System Management



Systems Management Enhancements

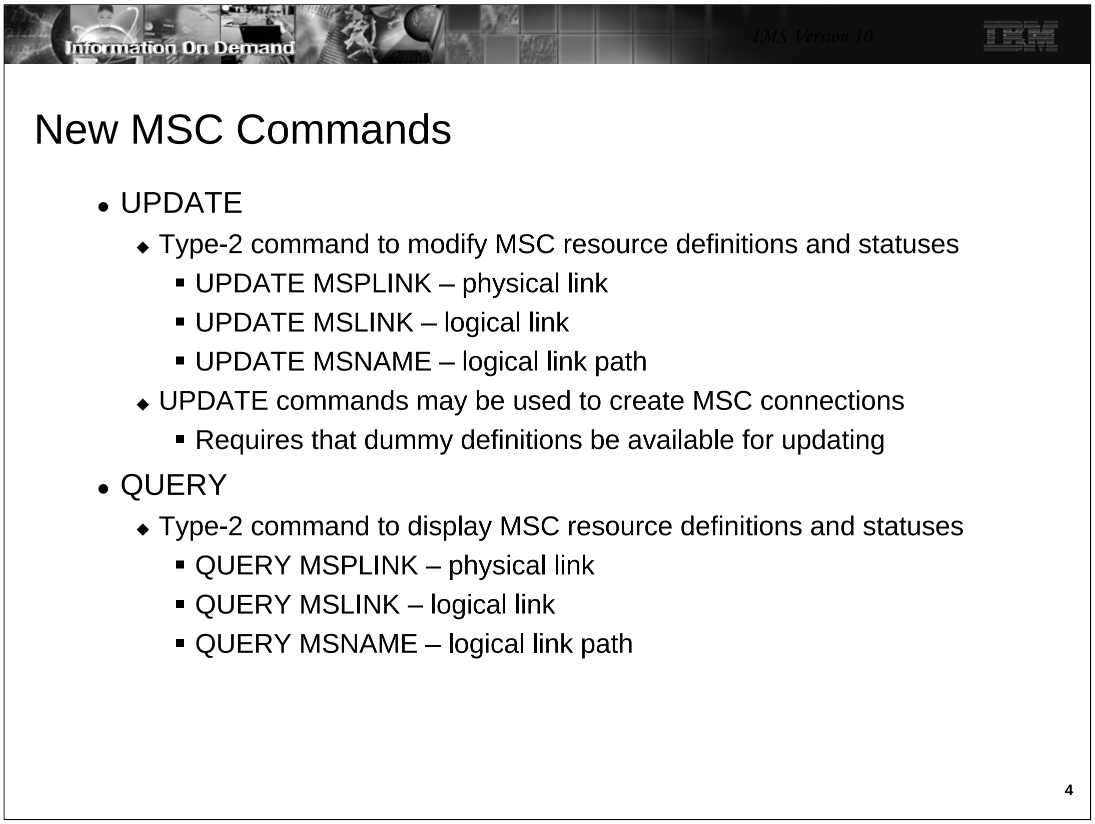
- MSC dynamic definition updates
- IMS command enhancements
  - QUERY command
  - UPDATE command
  - QUEUE command
- Enhanced messages
- Enhanced secondary master terminal support
- Global status support for DBs, AREAs, and TRANs



MSC Command Support

3

Although the DRD CREATE and DELETE process does not support MSC resources, there is some enhanced support for MSC in V10.


Information On Demand
IMS Version 10
IBM

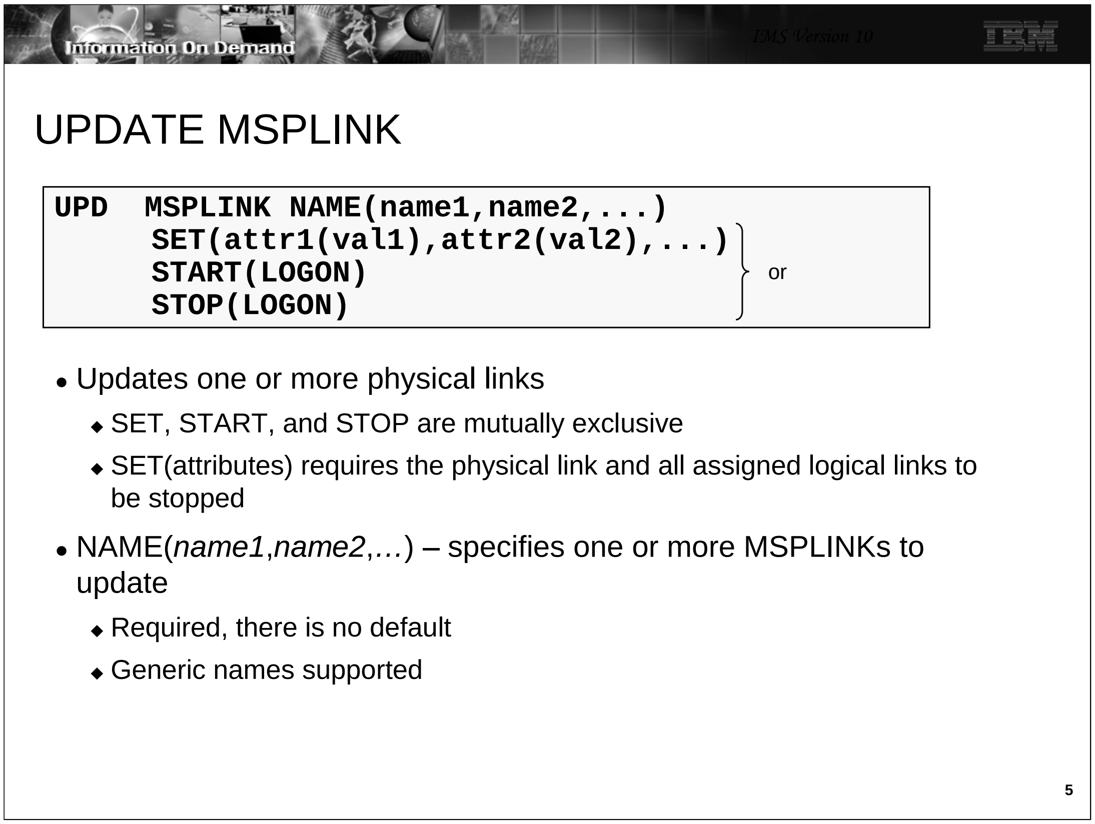
New MSC Commands

- UPDATE
  - Type-2 command to modify MSC resource definitions and statuses
    - UPDATE MSPLINK – physical link
    - UPDATE MSLINK – logical link
    - UPDATE MSNAME – logical link path
  - UPDATE commands may be used to create MSC connections
    - Requires that dummy definitions be available for updating
- QUERY
  - Type-2 command to display MSC resource definitions and statuses
    - QUERY MSPLINK – physical link
    - QUERY MSLINK – logical link
    - QUERY MSNAME – logical link path

4

IMS V10 adds support for MSC resources with the type-2 UPDATE and QUERY commands.

There is a new performance parameter for MSC logical links called BANDWIDTH. You will see this reference in the following slides but a discussion of what this means is in Transaction Manager Enhancements section of the class.



Information On Demand
IMS Version 10

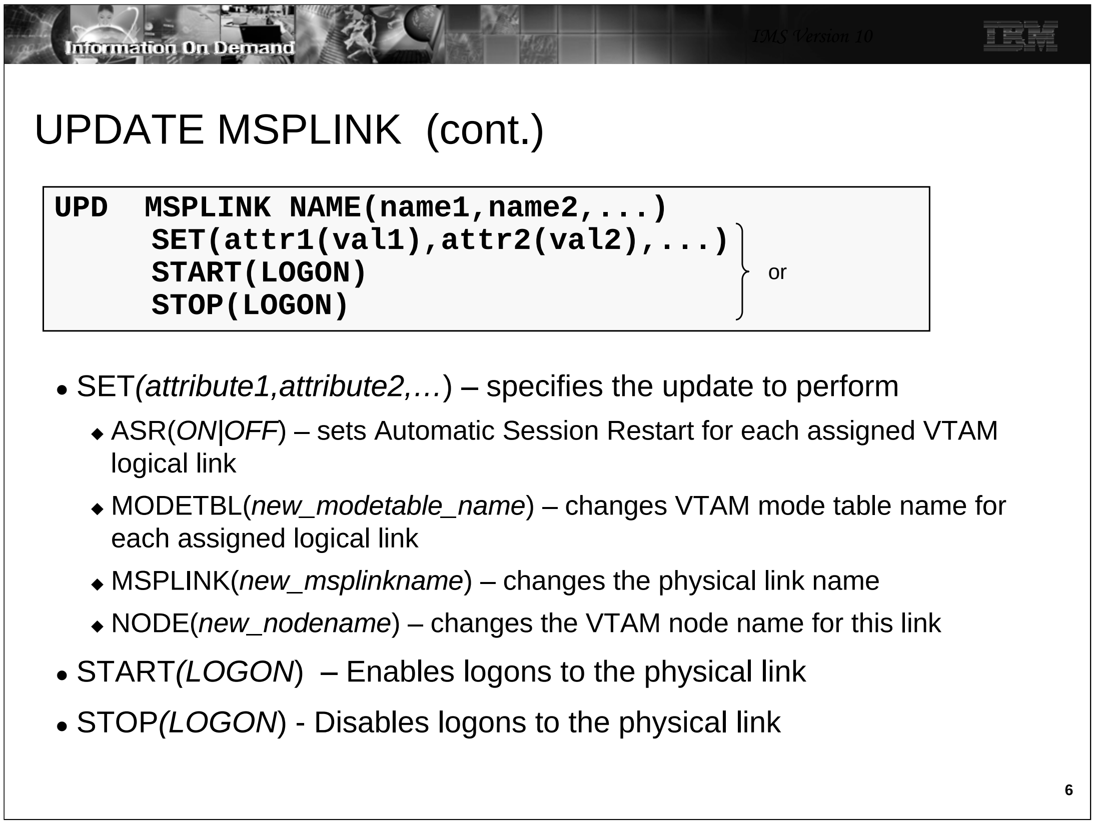
UPDATE MSPLINK

```
UPD  MSPLINK NAME(name1,name2,...)
     SET(attr1(val1),attr2(val2),...) }
     START(LOGON)                      } or
     STOP(LOGON)                       }
```

- Updates one or more physical links
  - SET, START, and STOP are mutually exclusive
  - SET(attributes) requires the physical link and all assigned logical links to be stopped
- NAME(*name1*,*name2*,*…*) – specifies one or more MSPLINKs to update
  - Required, there is no default
  - Generic names supported

5

This slide shows the format of the UPDATE MSPLINK command.


Information On Demand
IMS Version 10

UPDATE MSPLINK (cont.)

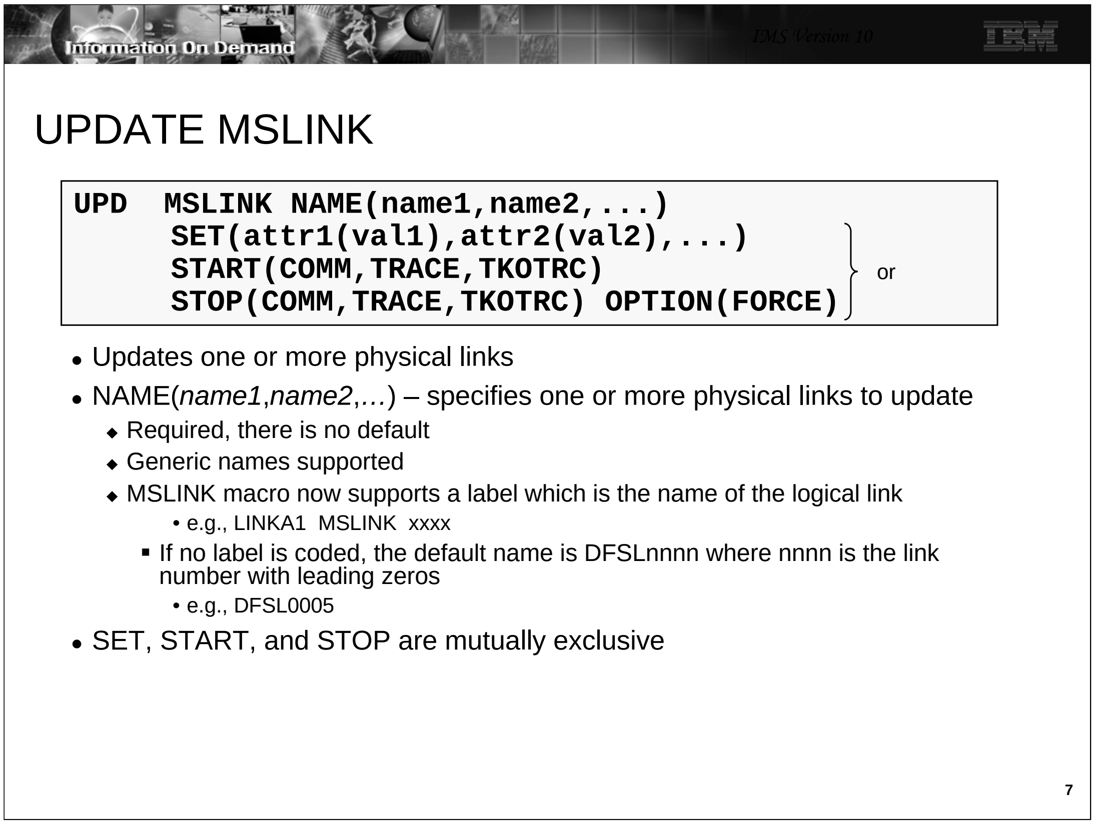
```
UPD  MSPLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...) ⎫
      START(LOGON)                     ⎬ or
      STOP(LOGON)                      ⎭
```

- SET*(attribute1,attribute2,…)* – specifies the update to perform
  - ASR(*ON*|*OFF*) – sets Automatic Session Restart for each assigned VTAM logical link
  - MODETBL(*new_modetable_name*) – changes VTAM mode table name for each assigned logical link
  - MSPLINK(*new_msplinkname*) – changes the physical link name
  - NODE(*new_nodename*) – changes the VTAM node name for this link
- START*(LOGON*) – Enables logons to the physical link
- STOP*(LOGON*) - Disables logons to the physical link

6

The UPDATE MSPLINK command can be used to change any of the four attributes identified on the slide. Except for changing the MSPLINK name, they apply only to VTAM links. Before the attributes can be changed, the physical link and all assigned logical links must be stopped.

START and STOP LOGON enables or stops logons to the physical link. STOP does not affect existing sessions.



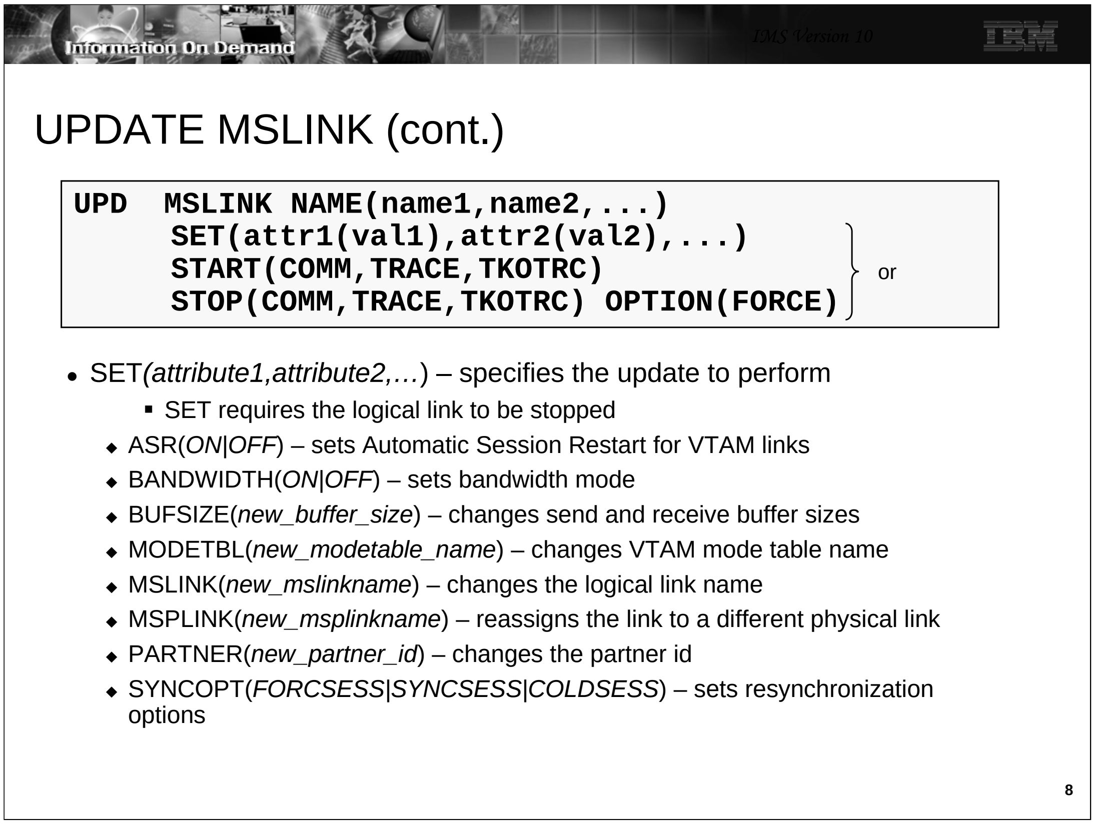
UPDATE MSLINK

```
UPD  MSLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...)         
      START(COMM,TRACE,TKOTRC)                   } or
      STOP(COMM,TRACE,TKOTRC) OPTION(FORCE)
```

- Updates one or more physical links
- NAME(*name1*,*name2*,*…*) – specifies one or more physical links to update
  - Required, there is no default
  - Generic names supported
  - MSLINK macro now supports a label which is the name of the logical link
    - e.g., LINKA1 MSLINK xxxx
    - If no label is coded, the default name is DFSLnnnn where nnnn is the link number with leading zeros
      - e.g., DFSL0005
- SET, START, and STOP are mutually exclusive

7

This slide shows the format of the UPDATE MSLINK command. As with the other UPD commands, you cannot change attributes and the status in the same command. The NAME of the MSLINK is either the label on the MSLINK macro or, if no label exists, a default name of DFSLnnnn where nnnn is the MSLink number with leading zeros. The command does not support MSLink numbers. You must specify a NAME.

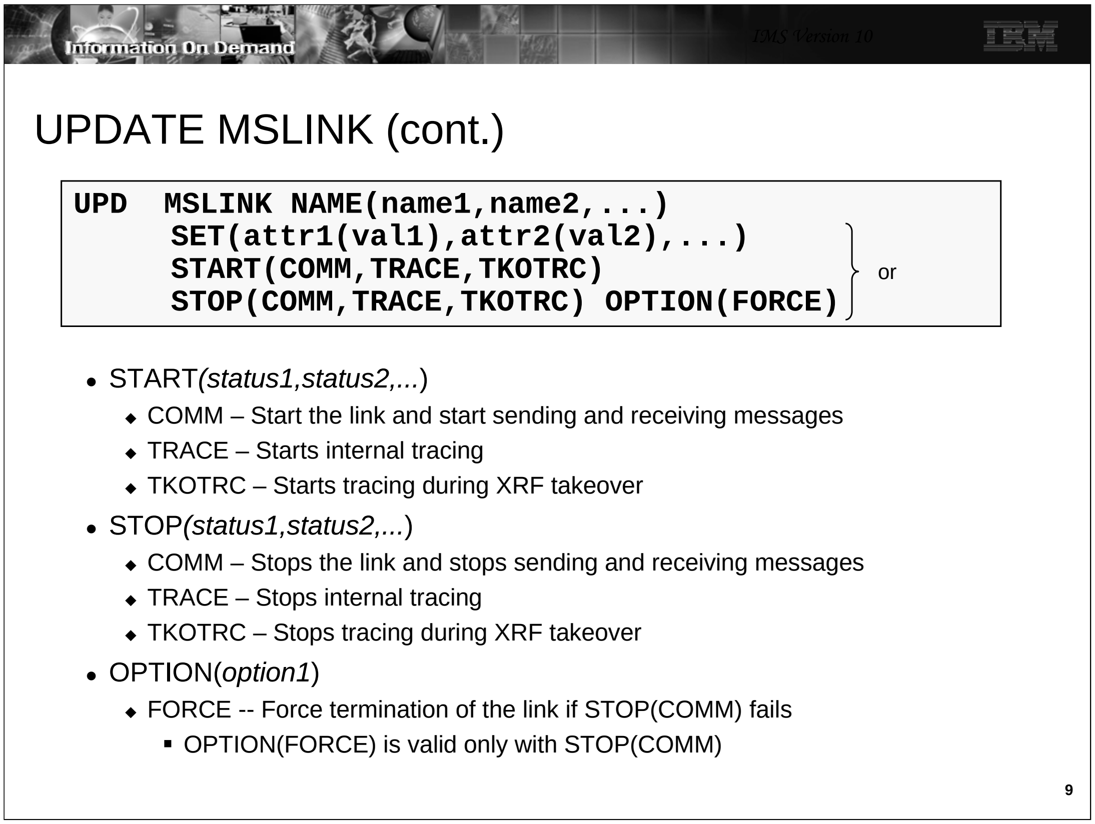

Information On Demand
IMS Version 10

UPDATE MSLINK (cont.)

```
UPD  MSLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...)
      START(COMM,TRACE,TKOTRC)                 or
      STOP(COMM,TRACE,TKOTRC) OPTION(FORCE)
```

- SET*(attribute1,attribute2,…*) – specifies the update to perform
  - SET requires the logical link to be stopped
  - ASR(*ON*|*OFF*) – sets Automatic Session Restart for VTAM links
  - BANDWIDTH(*ON*|*OFF*) – sets bandwidth mode
  - BUFSIZE(*new_buffer_size*) – changes send and receive buffer sizes
  - MODETBL(*new_modetable_name*) – changes VTAM mode table name
  - MSLINK(*new_mslinkname*) – changes the logical link name
  - MSPLINK(*new_msplinkname*) – reassigns the link to a different physical link
  - PARTNER(*new_partner_id*) – changes the partner id
  - SYNCOPT(*FORCSESS*|*SYNCSESS*|*COLDSESS*) – sets resynchronization options

Any of the attributes can be changed using this command. The link must be stopped before changing the attributes.

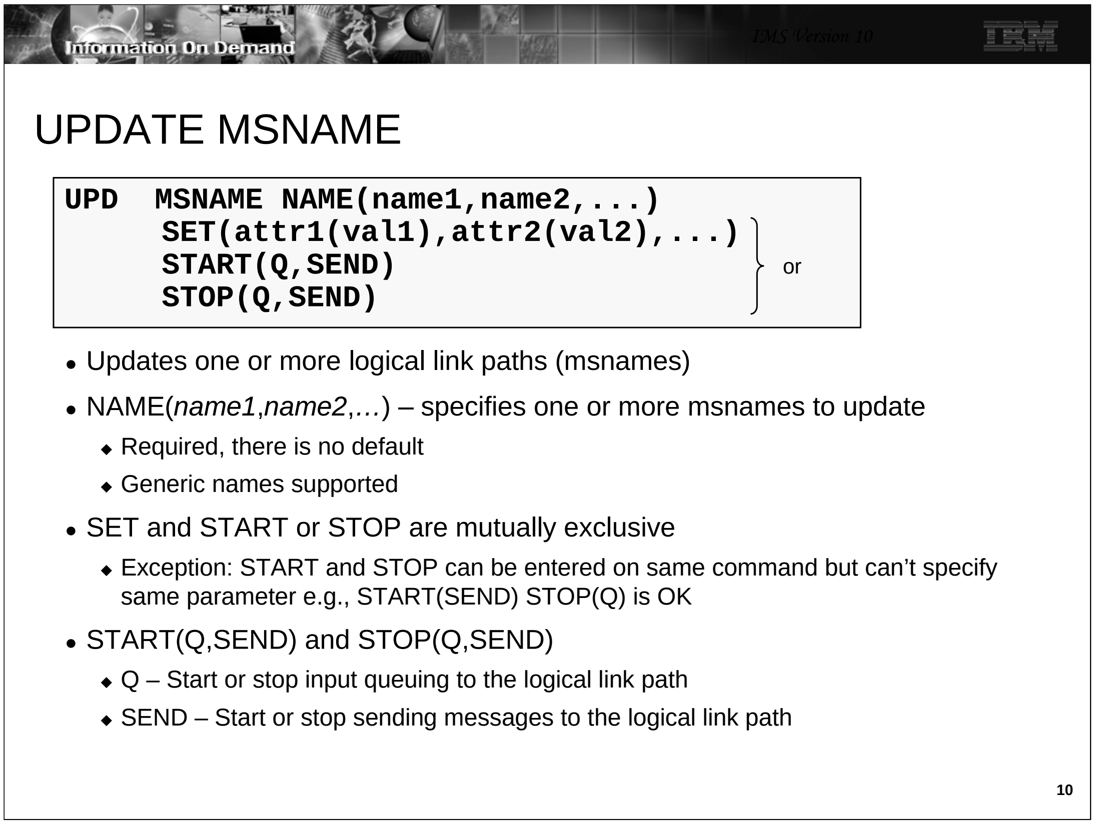


UPDATE MSLINK (cont.)

```
UPD  MSLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...)    }
      START(COMM,TRACE,TKOTRC)            }  or
      STOP(COMM,TRACE,TKOTRC) OPTION(FORCE)}
```

- START*(status1,status2,...*)
  - COMM – Start the link and start sending and receiving messages
  - TRACE – Starts internal tracing
  - TKOTRC – Starts tracing during XRF takeover
- STOP*(status1,status2,...*)
  - COMM – Stops the link and stops sending and receiving messages
  - TRACE – Stops internal tracing
  - TKOTRC – Stops tracing during XRF takeover
- OPTION(*option1*)
  - FORCE -- Force termination of the link if STOP(COMM) fails
    - OPTION(FORCE) is valid only with STOP(COMM)

The START and STOP keywords can be entered on the same command, but you cannot start and stop the same status. OPTION(FORCE) is valid only for VTAM and CTC sessions, and only when used in conjunction with STOP(COMM).



Information On Demand
IMS Version 10
IBM

UPDATE MSNAME

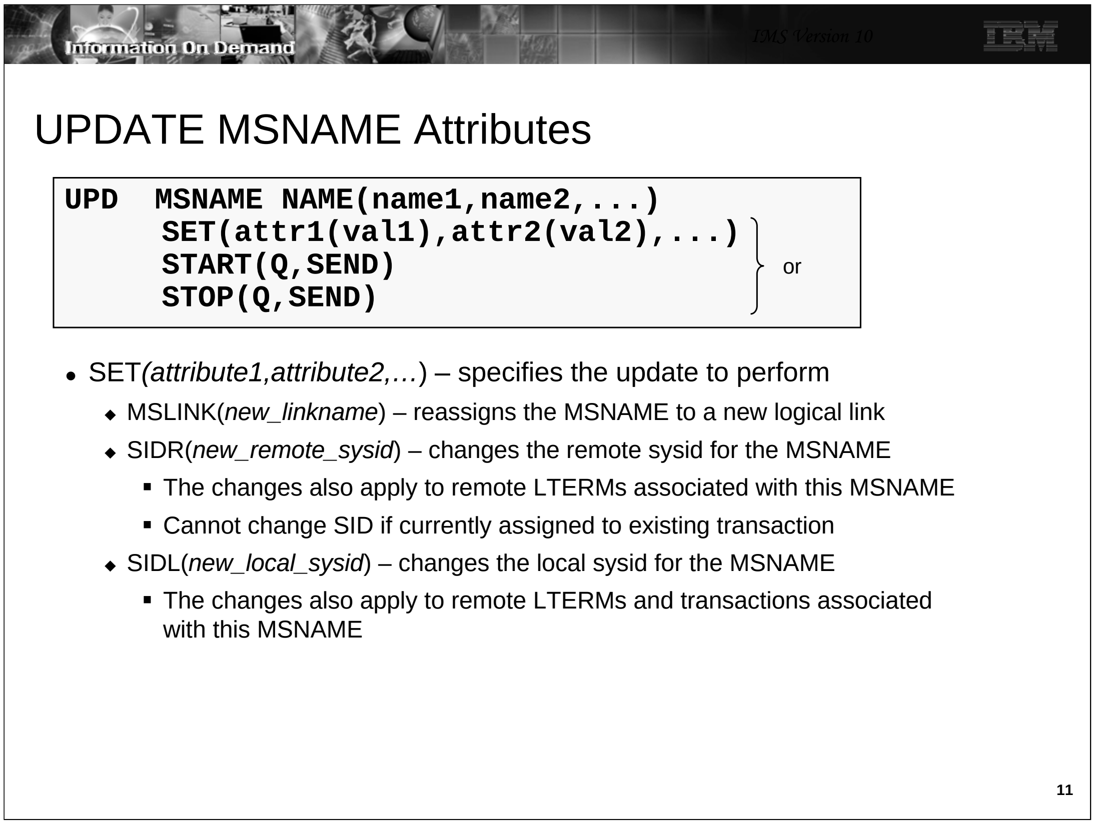
```
UPD  MSNAME NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...) ⎫
      START(Q,SEND)                     ⎬  or
      STOP(Q,SEND)                      ⎭
```

- Updates one or more logical link paths (msnames)
- NAME(*name1*,*name2*,*…*) – specifies one or more msnames to update
  - Required, there is no default
  - Generic names supported
- SET and START or STOP are mutually exclusive
  - Exception: START and STOP can be entered on same command but can’t specify same parameter e.g., START(SEND) STOP(Q) is OK
- START(Q,SEND) and STOP(Q,SEND)
  - Q – Start or stop input queuing to the logical link path
  - SEND – Start or stop sending messages to the logical link path

10

The slide shows the format of the UPDATE MSNAME command. As with the others, you cannot issue a command with SET and with START or STOP.

UPD MSNAME can be used to start or stop queuing and sending messages across that logical link path. You can use START and STOP in the same command, but not with the same status.



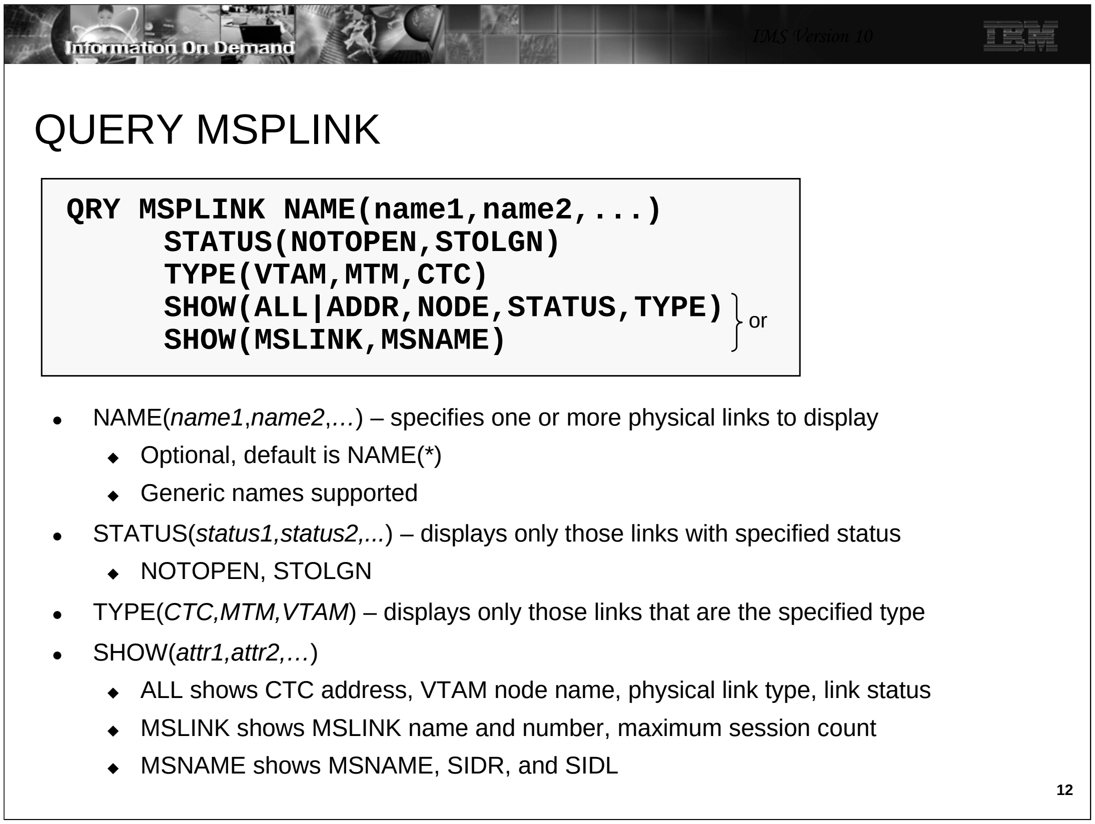
UPDATE MSNAME Attributes

```
UPD  MSNAME NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...) ⎫
      START(Q,SEND)                    ⎬ or
      STOP(Q,SEND)                     ⎭
```

- SET(*attribute1*,*attribute2,…*) – specifies the update to perform
  - MSLINK(*new_linkname*) – reassigns the MSNAME to a new logical link
  - SIDR(*new_remote_sysid*) – changes the remote sysid for the MSNAME
    - The changes also apply to remote LTERMs associated with this MSNAME
    - Cannot change SID if currently assigned to existing transaction
  - SIDL(*new_local_sysid*) – changes the local sysid for the MSNAME
    - The changes also apply to remote LTERMs and transactions associated with this MSNAME

To change the logical link (MSLINK) with which this logical link path (MSNAME) is associated, the logical link must be stopped. To change the remote and local sysids, both the logical link and the logical link path must be stopped.

If the SYSIDs are currently assigned to an existing transaction, you must first use the UPDATE TRAN SET(SIDL(),SIDR()) command to reassign that transaction to another valid SYSID. Then you can update SIDR and SIDL. Then you can UPDATE TRAN again to reassign the remote and local sysids for the transaction back to the new sysids.


Information On Demand
IMS Version 10
IBM

QUERY MSPLINK

```
QRY MSPLINK NAME(name1,name2,...)
     STATUS(NOTOPEN,STOLGN)
     TYPE(VTAM,MTM,CTC)
     SHOW(ALL|ADDR,NODE,STATUS,TYPE) } or
     SHOW(MSLINK,MSNAME)             }
```

- NAME(*name1*,*name2*,*…*) – specifies one or more physical links to display
  - Optional, default is NAME(*)
  - Generic names supported
- STATUS(*status1,status2,...*) – displays only those links with specified status
  - NOTOPEN, STOLGN
- TYPE(*CTC,MTM,VTAM*) – displays only those links that are the specified type
- SHOW(*attr1,attr2,…*)
  - ALL shows CTC address, VTAM node name, physical link type, link status
  - MSLINK shows MSLINK name and number, maximum session count
  - MSNAME shows MSNAME, SIDR, and SIDL

12

This slide shows the format of the QRY MSPLINK command.

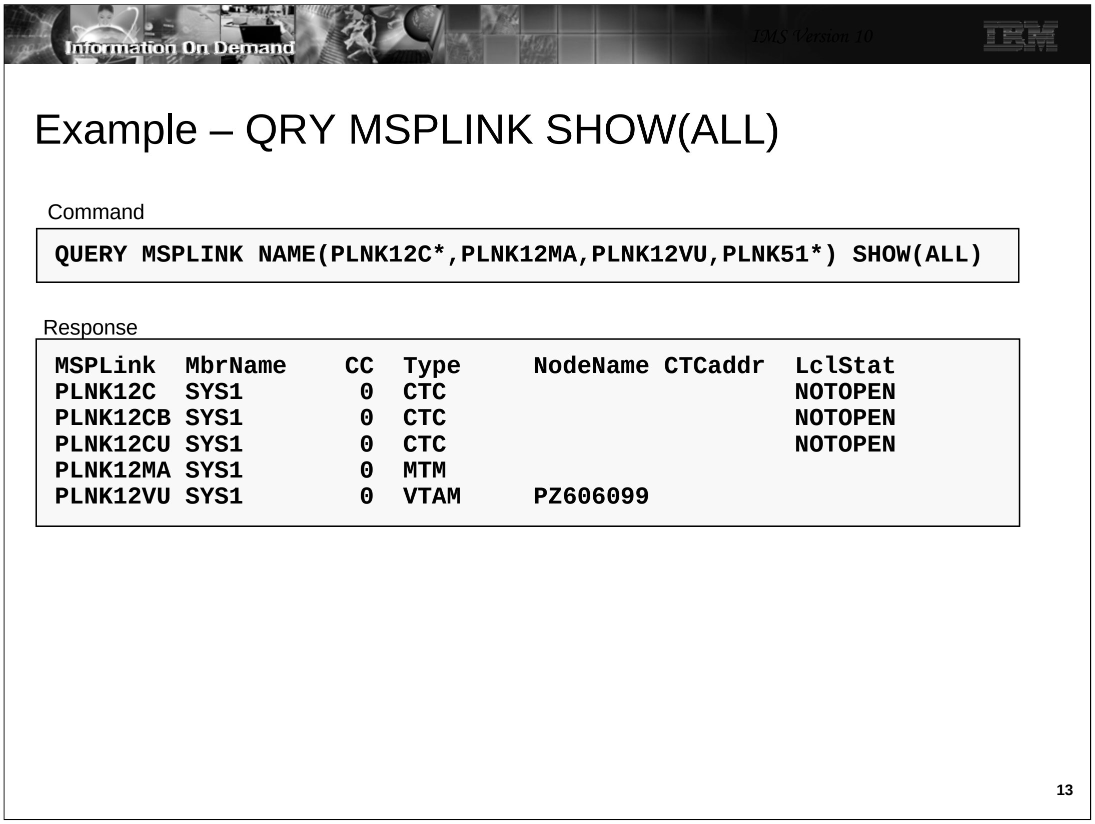
NAME is the name of the physical link. You can specify one or more names, and wild cards are allowed (* = multi-character substitution and % = single character substitution). The default is * which is to display all physical links.

STATUS is a filter to limit the responses to those MSPLINKs with the specified status. Multiple statuses are ORed. For example, if STATUS(STOLGN) is specified, only those physical links with the STOP LOGON status will be displayed. The only other valid status is NOTOPEN.

TYPE is a filter for the link type – VTAM, Memory-to-memory, or Channel-to-channel.

SHOW determines what information is returned by IMS and what the TSO SPOC will show on the screen. SHOW(ALL) includes CRC address. VTAM node name of the remote IMS, the physical link type, and the link status. Each individual attribute may be selected - e.g., SHOW(NODE,STATUS). You can also select SHOW(MSLINK and/or MSNAME). You cannot specify the first and second type of SHOW together on the same command - e.g. SHOW(NODE,STATUS,MSNAME,MSLINK) is not allowed. MSNAME displays the maximum session count defined on the MSPLINK macro as well as the link name, link number.

Note: V10 permits a label on the MSLINK macro. This label is the MSLINK name. If not specified, the default name is DFSLnnnn, where nnnn is the link number.



Example – QRY MSPLINK SHOW(ALL)

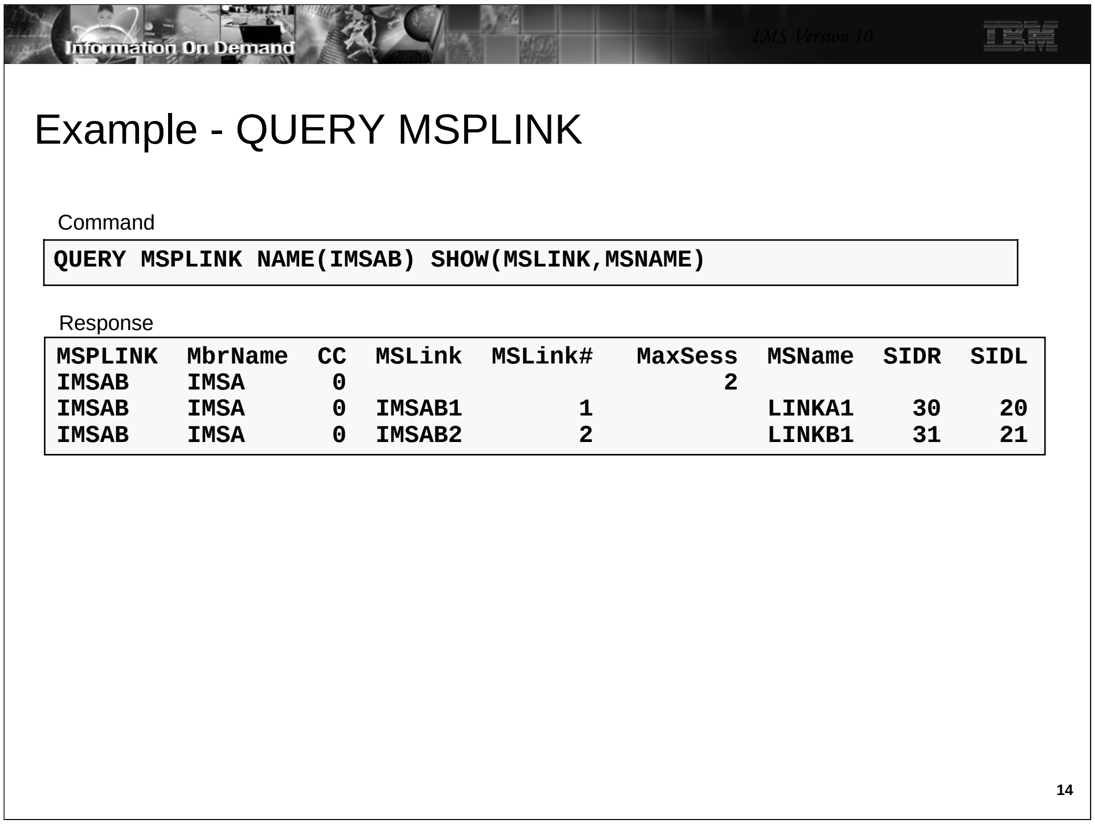
Command

```
QUERY MSPLINK NAME(PLNK12C*,PLNK12MA,PLNK12VU,PLNK51*) SHOW(ALL)
```

Response

```
MSPLink  MbrName    CC  Type     NodeName CTCaddr  LclStat
PLNK12C  SYS1        0  CTC                        NOTOPEN
PLNK12CB SYS1        0  CTC                        NOTOPEN
PLNK12CU SYS1        0  CTC                        NOTOPEN
PLNK12MA SYS1        0  MTM
PLNK12VU SYS1        0  VTAM     PZ606099
```

This is an example of a QRY MSPLINK NAME(xxx) SHOW(ALL) command. All QRY MSPLINK commands show the MSPLINK name, the member name responding, the condition code, and, if CC is not zero, the CC text explaining why the command failed.



Example - QUERY MSPLINK

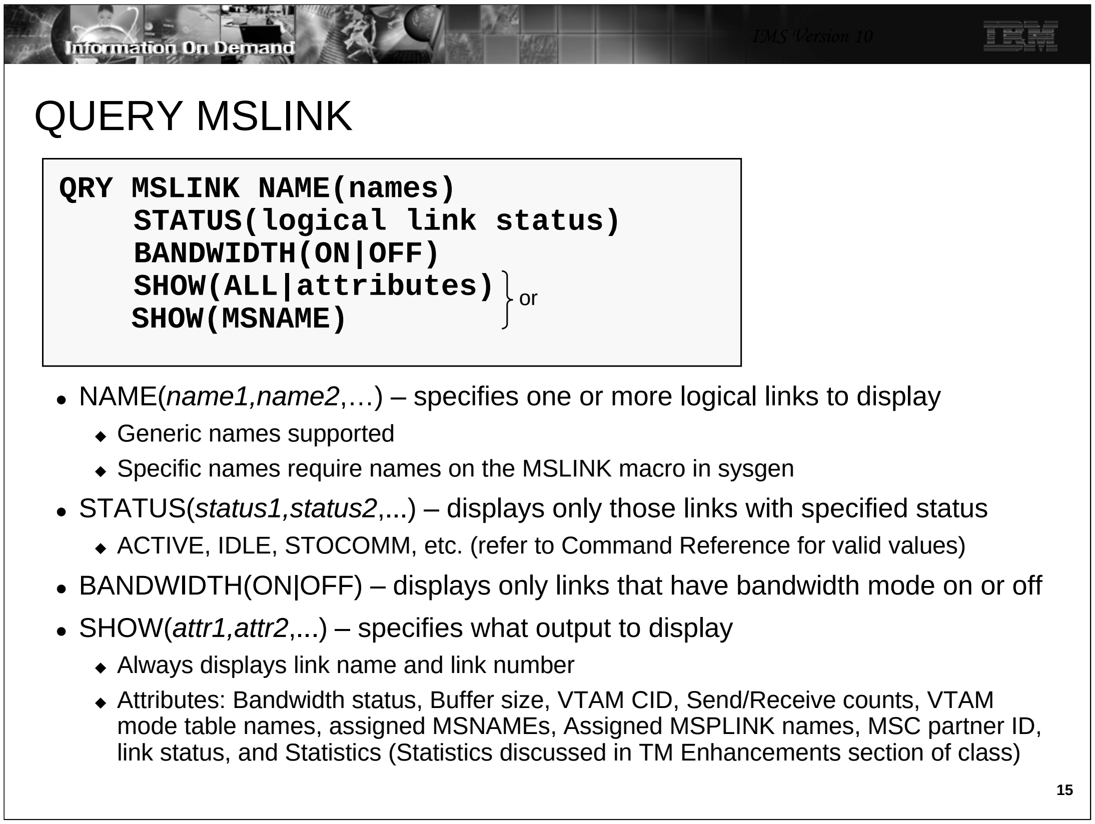
Command

```
QUERY MSPLINK NAME(IMSAB) SHOW(MSLINK,MSNAME)
```

Response

```
MSPLINK  MbrName  CC  MSLink  MSLink#   MaxSess  MSName  SIDR  SIDL
IMSAB    IMSA      0                          2
IMSAB    IMSA      0  IMSAB1        1            LINKA1    30    20
IMSAB    IMSA      0  IMSAB2        2            LINKB1    31    21
```

This slide shows an example of QRY MSPLINK showing MSLINK and MSNAME. For MSLINK, the MSLink name and number and the MSPLINK maximum sessions are shown. For MSNAME, the MSName and remote and local sysids are shown.



Information On Demand

IMS Version 10

QUERY MSLINK

```
QRY MSLINK NAME(names)
    STATUS(logical link status)
    BANDWIDTH(ON|OFF)
    SHOW(ALL|attributes) } or
    SHOW(MSNAME)         }
```

- NAME(*name1,name2*,…) – specifies one or more logical links to display
  - Generic names supported
  - Specific names require names on the MSLINK macro in sysgen
- STATUS(*status1,status2*,...) – displays only those links with specified status
  - ACTIVE, IDLE, STOCOMM, etc. (refer to Command Reference for valid values)
- BANDWIDTH(ON|OFF) – displays only links that have bandwidth mode on or off
- SHOW(*attr1,attr2*,...) – specifies what output to display
  - Always displays link name and link number
  - Attributes: Bandwidth status, Buffer size, VTAM CID, Send/Receive counts, VTAM mode table names, assigned MSNAMEs, Assigned MSPLINK names, MSC partner ID, link status, and Statistics (Statistics discussed in TM Enhancements section of class)

15

This is the format of the QRY MSLINK command.

NAME is like other query commands. The name of the MSLINK is either the label on the macro (new in V10), or the default name DFSLnnnn, where nnnn is the MSLINK number with leading zeros. The command does not support MSLink numbers. You must specify a NAME.

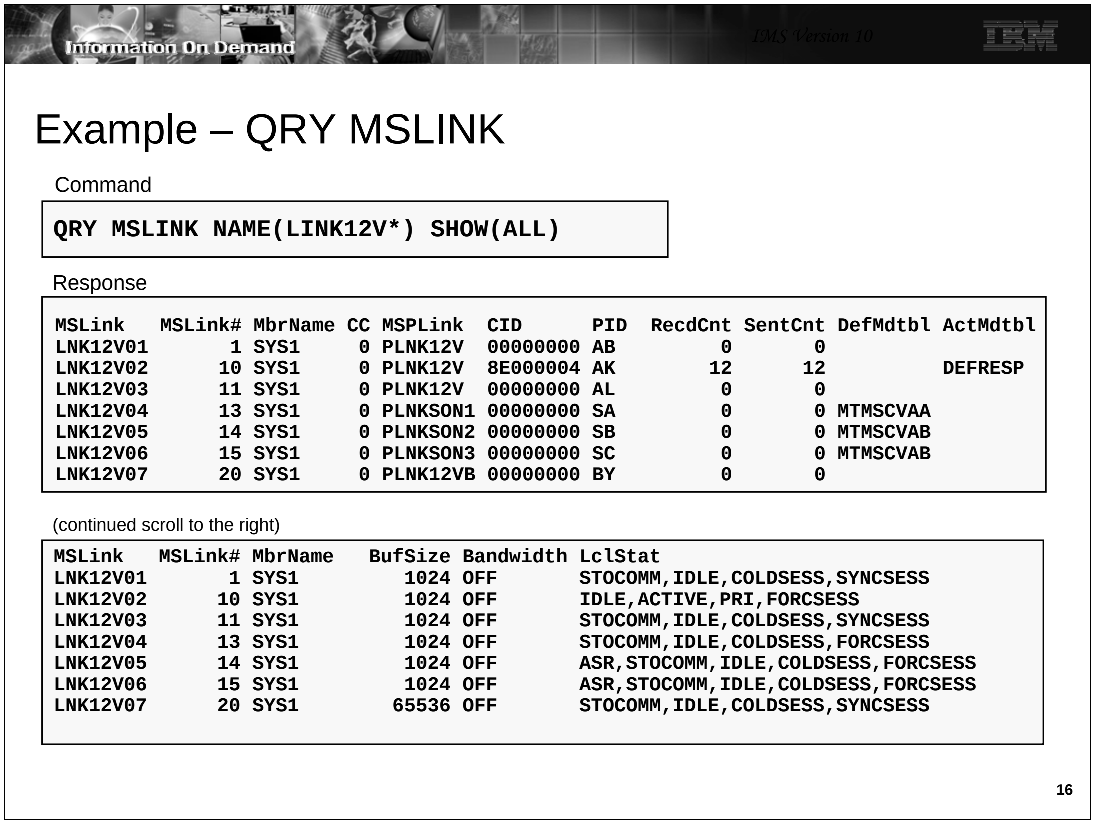
STATUS is a filter to limit the responses to those MSLINKs with the specified status. Multiple statuses are ORed. These statuses may be found in the Command Reference publication.

BANDWIDTH is a filter for whether the new bandwidth performance option is turned OFF or ON.

SHOW always shows the MSNAME name (the label) and the link number to which this MSNAME is assigned.

SHOW(ALL) shows all the definitional attributes and the status of the logical link. The attributes that may be specified can found in the Command Reference publication. You can request that just some of them be shown. Statistics may be requested. This is discussed in the TM Enhancements section of this class. SHOW(ALL) does not return statistics. They must be explicitly requested.

SHOW(MSNAME) cannot be specified with SHOW(ALL) or any of the attributes.


Information On Demand
IMS Version 10

Example – QRY MSLINK

Command

```
QRY MSLINK NAME(LINK12V*) SHOW(ALL)
```

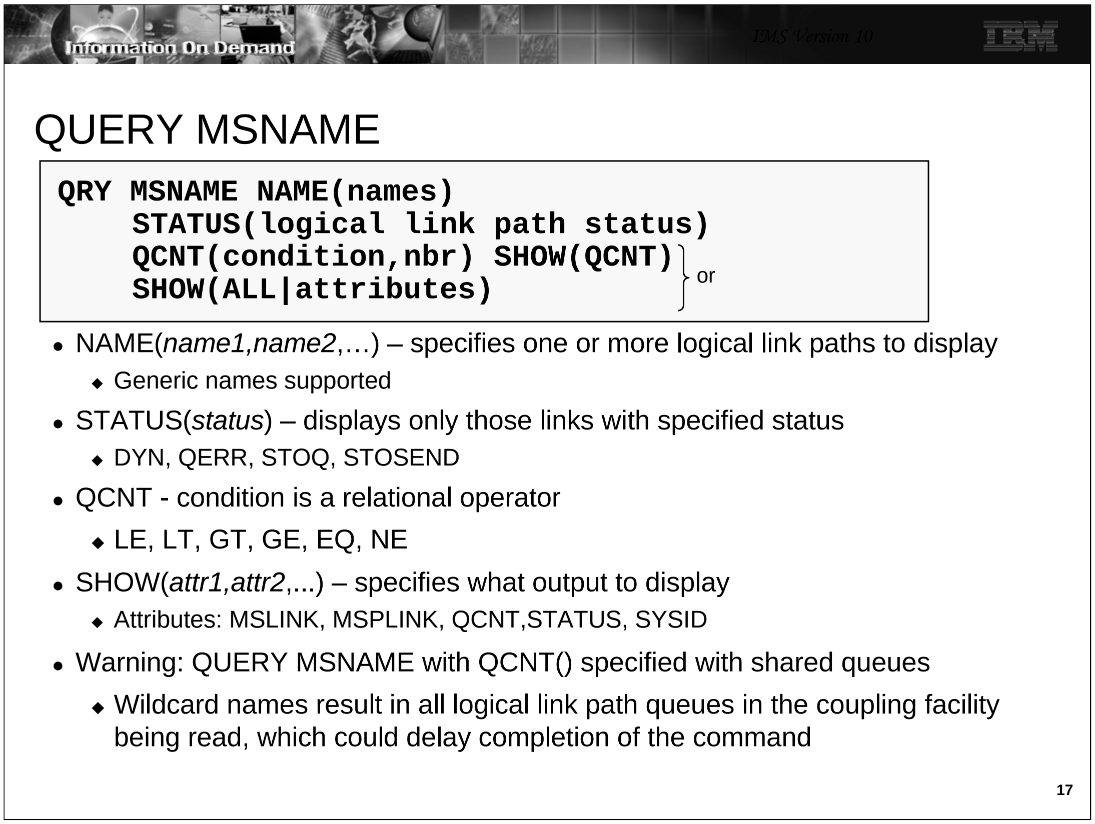
Response

```
MSLink   MSLink# MbrName CC MSPLink  CID      PID  RecdCnt SentCnt DefMdtbl ActMdtbl
LNK12V01       1 SYS1     0 PLNK12V  00000000 AB         0       0
LNK12V02      10 SYS1     0 PLNK12V  8E000004 AK        12      12          DEFRESP
LNK12V03      11 SYS1     0 PLNK12V  00000000 AL         0       0
LNK12V04      13 SYS1     0 PLNKSON1 00000000 SA         0       0 MTMSCVAA
LNK12V05      14 SYS1     0 PLNKSON2 00000000 SB         0       0 MTMSCVAB
LNK12V06      15 SYS1     0 PLNKSON3 00000000 SC         0       0 MTMSCVAB
LNK12V07      20 SYS1     0 PLNK12VB 00000000 BY         0       0
```

(continued scroll to the right)

```
MSLink   MSLink# MbrName   BufSize Bandwidth LclStat
LNK12V01       1 SYS1         1024 OFF       STOCOMM,IDLE,COLDSESS,SYNCSESS
LNK12V02      10 SYS1         1024 OFF       IDLE,ACTIVE,PRI,FORCSESS
LNK12V03      11 SYS1         1024 OFF       STOCOMM,IDLE,COLDSESS,SYNCSESS
LNK12V04      13 SYS1         1024 OFF       STOCOMM,IDLE,COLDSESS,FORCSESS
LNK12V05      14 SYS1         1024 OFF       ASR,STOCOMM,IDLE,COLDSESS,FORCSESS
LNK12V06      15 SYS1         1024 OFF       ASR,STOCOMM,IDLE,COLDSESS,FORCSESS
LNK12V07      20 SYS1        65536 OFF       STOCOMM,IDLE,COLDSESS,SYNCSESS
```

This slide shows an example of the QRY MSLINK xxx SHOW(ALL) command. The output is too wide for a single screen, so the TSO SPOC will let you scroll right and left, or up and down. The lower part if this screen assumes the user as scrolled right.



QUERY MSNAME

```
QRY MSNAME NAME(names)
    STATUS(logical link path status)
    QCNT(condition,nbr) SHOW(QCNT) ⎫
    SHOW(ALL|attributes)           ⎭ or
```

- NAME(*name1*,*name2*,…) – specifies one or more logical link paths to display
  - Generic names supported
- STATUS(*status*) – displays only those links with specified status
  - DYN, QERR, STOQ, STOSEND
- QCNT - condition is a relational operator
  - LE, LT, GT, GE, EQ, NE
- SHOW(*attr1*,*attr2*,...) – specifies what output to display
  - Attributes: MSLINK, MSPLINK, QCNT,STATUS, SYSID
- Warning: QUERY MSNAME with QCNT() specified with shared queues
  - Wildcard names result in all logical link path queues in the coupling facility being read, which could delay completion of the command

The logical link path status may be:

DYN - dynamically created in a shared queues environment

QERR - queue error

STOQ - stopped queuing

STOSEND - stopped sending

The "condition" value in the QCNT parameter is a relational operator. It can be:

LE – less than or equal

LT – less than

GE – greater than or equal

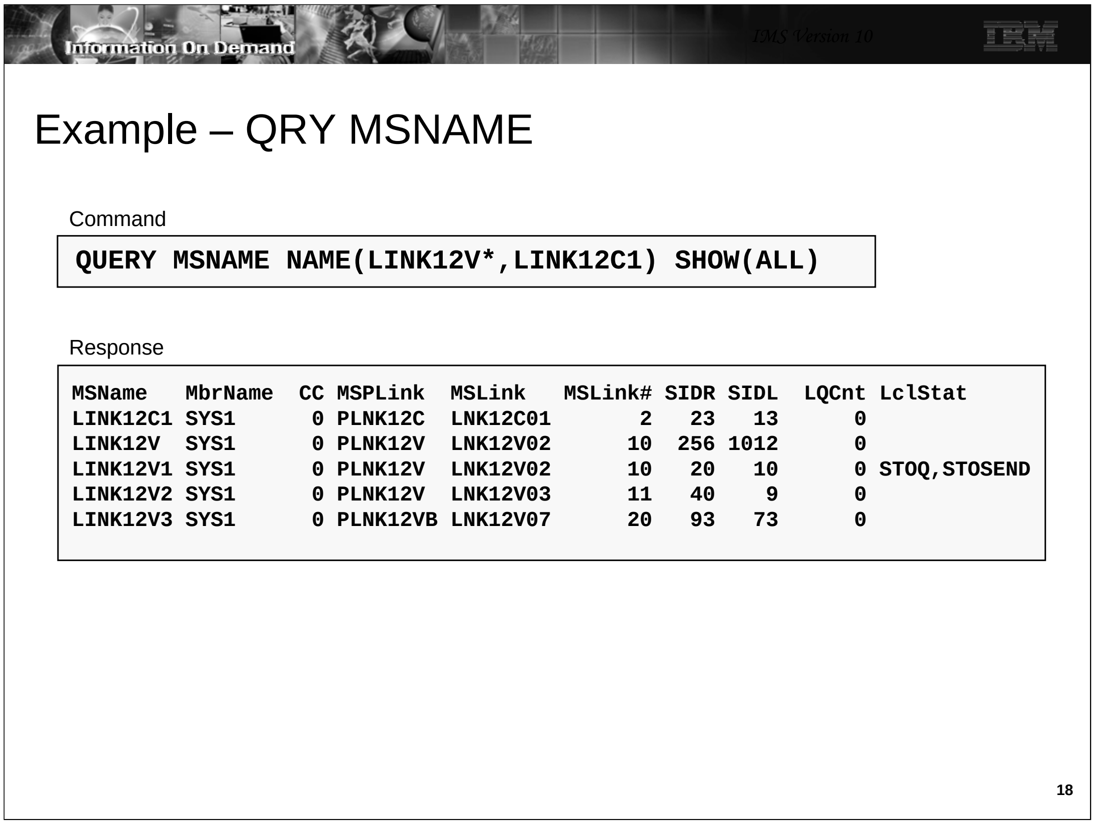
GT – greater than

EQ – equal

NE – not equal

QRY MSNAME NAME(*) QCNT(NE,0) SHOW(QCNT) would show all MSNAMEs with a queue count that is not zero.

Be careful when issuing the QUERY MSNAME ... QCNT command in a shared queues environment. The use of wildcard names will result in reading every remote queue “list” to determine whether it satisfies the wildcard name (XES list services does not support wildcards).



Example – QRY MSNAME

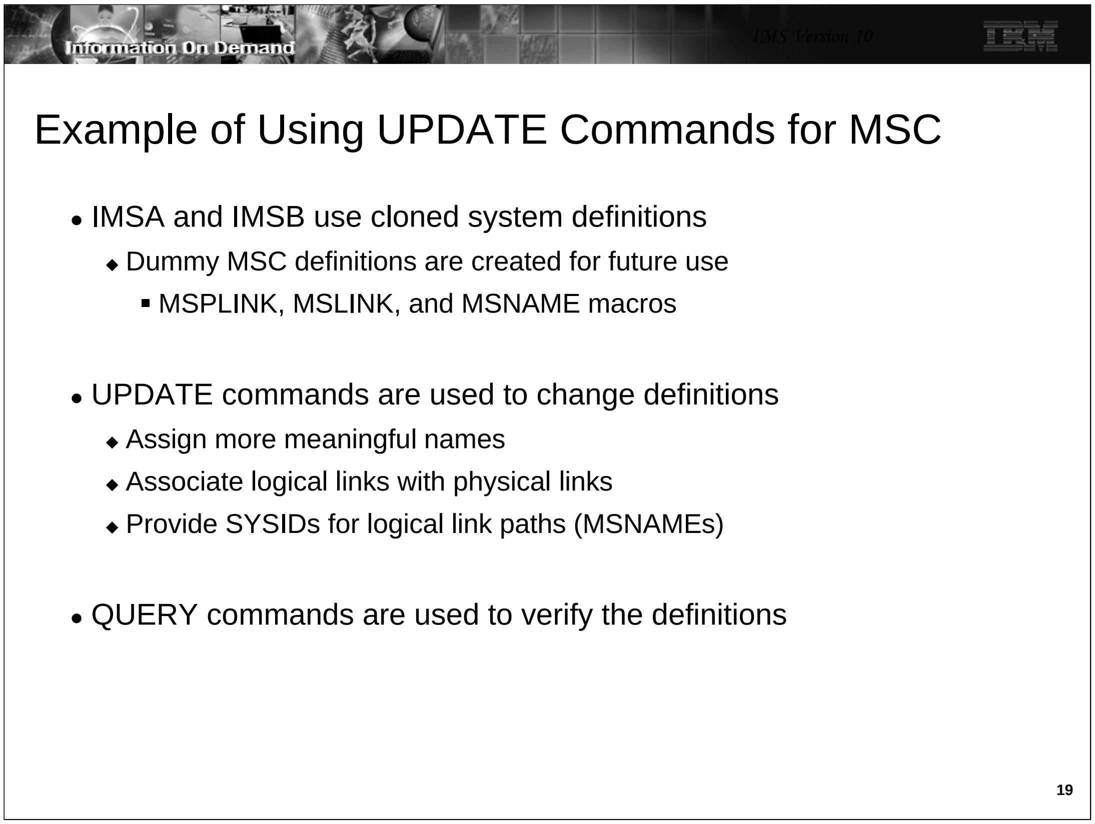
Command

```
QUERY MSNAME NAME(LINK12V*,LINK12C1) SHOW(ALL)
```

Response

```
MSName   MbrName  CC MSPLink  MSLink   MSLink# SIDR SIDL  LQCnt LclStat
LINK12C1 SYS1      0 PLNK12C  LNK12C01       2   23   13      0
LINK12V  SYS1      0 PLNK12V  LNK12V02      10  256 1012      0
LINK12V1 SYS1      0 PLNK12V  LNK12V02      10   20   10      0 STOQ,STOSEND
LINK12V2 SYS1      0 PLNK12V  LNK12V03      11   40    9      0
LINK12V3 SYS1      0 PLNK12VB LNK12V07      20   93   73      0
```

This is an example of a QRY MSNAME ... SHOW(ALL) command.   In a shared queues environment, there would be two lines for each MSName, with the first line showing the global QCNT and the second line showing the local attributes, queue count, and status as you see in the slide.



Information On Demand
IMS Version 10
IBM

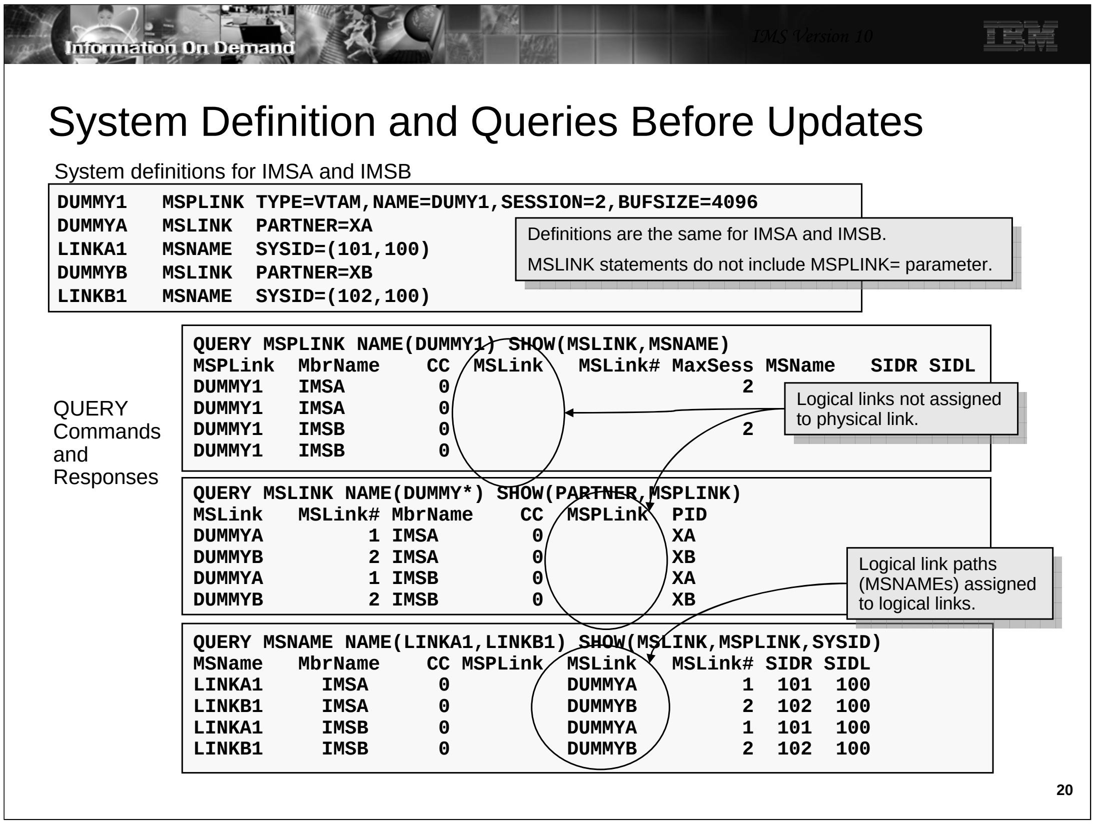
Example of Using UPDATE Commands for MSC

- IMSA and IMSB use cloned system definitions
  - Dummy MSC definitions are created for future use
    - MSPLINK, MSLINK, and MSNAME macros
- UPDATE commands are used to change definitions
  - Assign more meaningful names
  - Associate logical links with physical links
  - Provide SYSIDs for logical link paths (MSNAMEs)
- QUERY commands are used to verify the definitions

19

Although CREATE does not support the addition of new MSC links, with careful planning, you can define dummy physical links, logical links, and logical link paths in each IMS between which you might want to create links. Then when they are needed, you can use the UPDATE MSPLINK, MSLINK, and MSNAME commands to activate those links.

The following shows an example of this technique. First dummy MSC definitions are included in the system definitions for IMSA and IMSB. The two systems have cloned definitions for future use by UPDATE commands. Second, QUERY commands are used to show these definitions. Third, UPDATE commands are used to assign more meaningful names to the physical links and logical links, to associate logical links with specific physical links, and to provide appropriate SYSIDs for the logical link paths. Finally, QUERY commands are used to show the changed definitions.


Information On Demand
IMS Version 10

System Definition and Queries Before Updates

System definitions for IMSA and IMSB

```
DUMMY1   MSPLINK TYPE=VTAM,NAME=DUMY1,SESSION=2,BUFSIZE=4096
DUMMYA   MSLINK  PARTNER=XA
LINKA1   MSNAME  SYSID=(101,100)
DUMMYB   MSLINK  PARTNER=XB
LINKB1   MSNAME  SYSID=(102,100)
```

Definitions are the same for IMSA and IMSB.

MSLINK statements do not include MSPLINK= parameter.

QUERY Commands and Responses

```
QUERY MSPLINK NAME(DUMMY1) SHOW(MSLINK,MSNAME)
MSPLink  MbrName    CC  MSLink   MSLink# MaxSess MSName   SIDR SIDL
DUMMY1   IMSA        0                         2
DUMMY1   IMSA        0
DUMMY1   IMSB        0                         2
DUMMY1   IMSB        0
```

Logical links not assigned to physical link.

```
QUERY MSLINK NAME(DUMMY*) SHOW(PARTNER,MSPLINK)
MSLink   MSLink# MbrName    CC  MSPLink   PID
DUMMYA         1 IMSA        0            XA
DUMMYB         2 IMSA        0            XB
DUMMYA         1 IMSB        0            XA
DUMMYB         2 IMSB        0            XB
```

Logical link paths (MSNAMEs) assigned to logical links.

```
QUERY MSNAME NAME(LINKA1,LINKB1) SHOW(MSLINK,MSPLINK,SYSID)
MSName   MbrName    CC MSPLink  MSLink   MSLink# SIDR SIDL
LINKA1     IMSA      0          DUMMYA         1  101  100
LINKB1     IMSA      0          DUMMYB         2  102  100
LINKA1     IMSB      0          DUMMYA         1  101  100
LINKB1     IMSB      0          DUMMYB         2  102  100
```

20

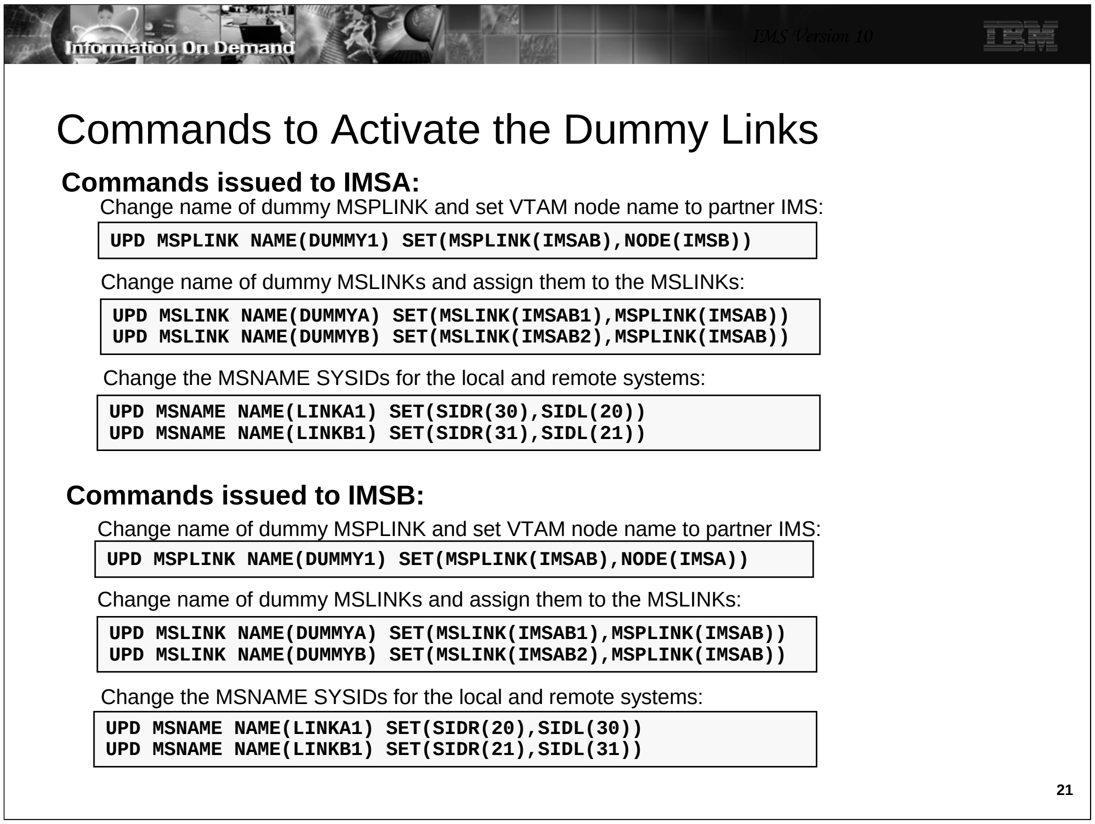
This shows the beginning of a scenario for defining dummy links and then activating them. Here we have defined in the System Definition with one MSPLINK, two MSLINKs, and two MSNAMEs.

The MSPLINK statement uses dummy names for the label and for the node name (NAME=).

The MSLINK statements use dummy names for the label and partner. They do not have a MSPLINK= parameter, so they are not assigned to a physical link.

The MSNAME statements use dummy numbers for the SYSIDs.

The QRY commands show the definitions before any changes are made. The MSNAMEs are assigned to MSLINKs, but since the MSLINKs are not defined to a MSPLINK, neither are the MSNAMEs.



Commands to Activate the Dummy Links

Commands issued to IMSA:

Change name of dummy MSPLINK and set VTAM node name to partner IMS:

```
UPD MSPLINK NAME(DUMMY1) SET(MSPLINK(IMSAB),NODE(IMSB))
```

Change name of dummy MSLINKs and assign them to the MSLINKs:

```
UPD MSLINK NAME(DUMMYA) SET(MSLINK(IMSAB1),MSPLINK(IMSAB))
UPD MSLINK NAME(DUMMYB) SET(MSLINK(IMSAB2),MSPLINK(IMSAB))
```

Change the MSNAME SYSIDs for the local and remote systems:

```
UPD MSNAME NAME(LINKA1) SET(SIDR(30),SIDL(20))
UPD MSNAME NAME(LINKB1) SET(SIDR(31),SIDL(21))
```

Commands issued to IMSB:

Change name of dummy MSPLINK and set VTAM node name to partner IMS:

```
UPD MSPLINK NAME(DUMMY1) SET(MSPLINK(IMSAB),NODE(IMSA))
```

Change name of dummy MSLINKs and assign them to the MSLINKs:

```
UPD MSLINK NAME(DUMMYA) SET(MSLINK(IMSAB1),MSPLINK(IMSAB))
UPD MSLINK NAME(DUMMYB) SET(MSLINK(IMSAB2),MSPLINK(IMSAB))
```

Change the MSNAME SYSIDs for the local and remote systems:

```
UPD MSNAME NAME(LINKA1) SET(SIDR(20),SIDL(30))
UPD MSNAME NAME(LINKB1) SET(SIDR(21),SIDL(31))
```

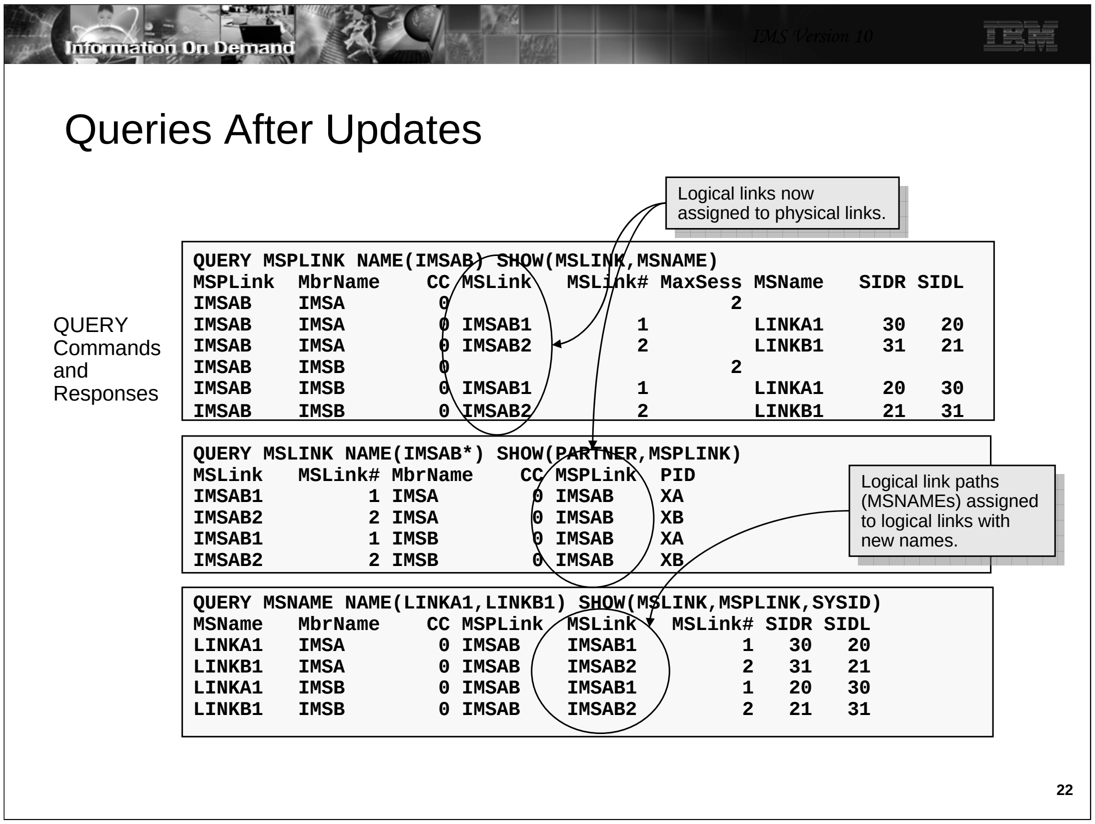
On this slide, we show how to “create” the MSC links between IMSA and IMSB. Similar commands are issued on both IMS systems. The responses to these commands are not shown.

For both systems the MSPLINK is updated to give it a name that is more descriptive and to identify the remote VTAM node. The command in IMSA specifies IMSB as the NODE name. The command in IMSB specifies IMSA as the NODE name.

The MSLINKs are updated to assign them to the renamed MSPLINK. The commands are the same for IMSA and IMSB.

MSNAMEs are updated to specify remote and local sysids. Of course, the remote sysids in IMSA are local sysids in IMSB.

After this we can CREATE remote transactions in IMSA and IMSB with the corresponding sysids.



Queries After Updates

Logical links now assigned to physical links.

QUERY Commands and Responses

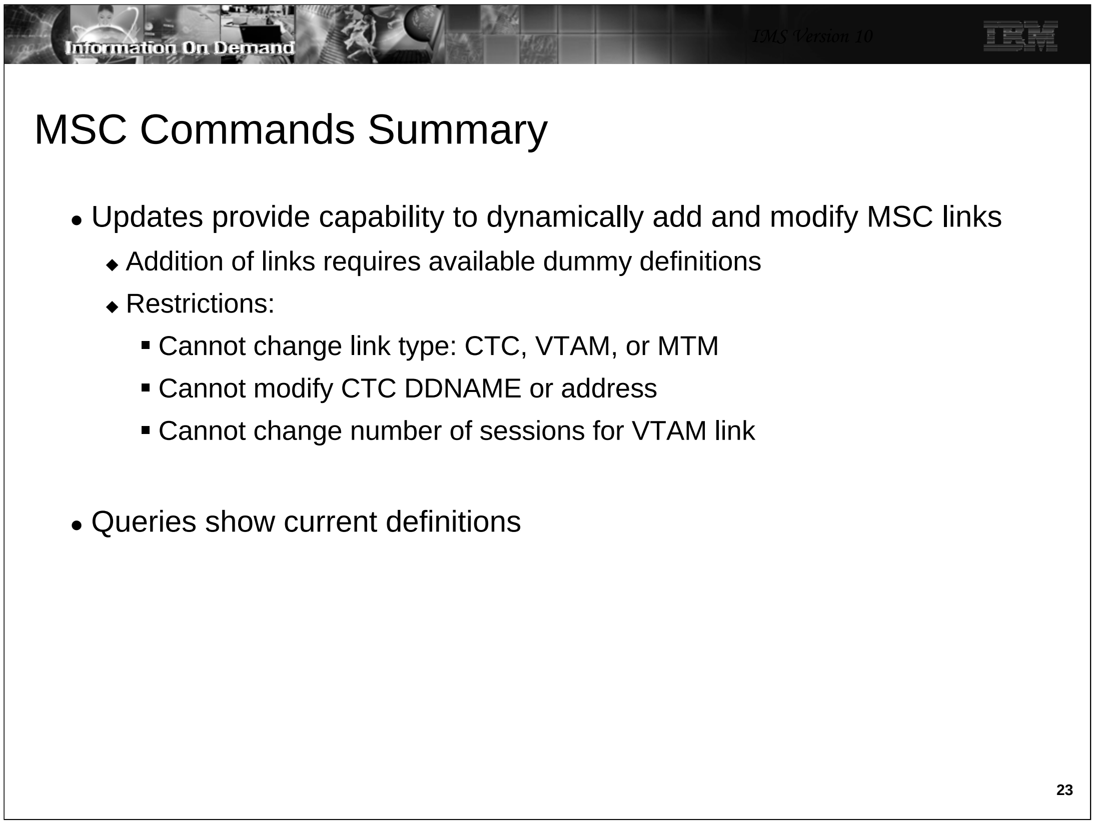
```
QUERY MSPLINK NAME(IMSAB) SHOW(MSLINK,MSNAME)
MSPLink  MbrName    CC MSLink   MSLink# MaxSess MSName   SIDR SIDL
IMSAB    IMSA        0                        2
IMSAB    IMSA        0 IMSAB1         1         LINKA1     30   20
IMSAB    IMSA        0 IMSAB2         2         LINKB1     31   21
IMSAB    IMSB        0                        2
IMSAB    IMSB        0 IMSAB1         1         LINKA1     20   30
IMSAB    IMSB        0 IMSAB2         2         LINKB1     21   31
```

```
QUERY MSLINK NAME(IMSAB*) SHOW(PARTNER,MSPLINK)
MSLink   MSLink# MbrName    CC MSPLink  PID
IMSAB1         1 IMSA        0 IMSAB    XA
IMSAB2         2 IMSA        0 IMSAB    XB
IMSAB1         1 IMSB        0 IMSAB    XA
IMSAB2         2 IMSB        0 IMSAB    XB
```

Logical link paths (MSNAMEs) assigned to logical links with new names.

```
QUERY MSNAME NAME(LINKA1,LINKB1) SHOW(MSLINK,MSPLINK,SYSID)
MSName   MbrName    CC MSPLink  MSLink   MSLink# SIDR SIDL
LINKA1   IMSA        0 IMSAB    IMSAB1         1   30   20
LINKB1   IMSA        0 IMSAB    IMSAB2         2   31   21
LINKA1   IMSB        0 IMSAB    IMSAB1         1   20   30
LINKB1   IMSB        0 IMSAB    IMSAB2         2   21   31
```

The QUERY commands show the results of the updates.



Information On Demand

IMS Version 10

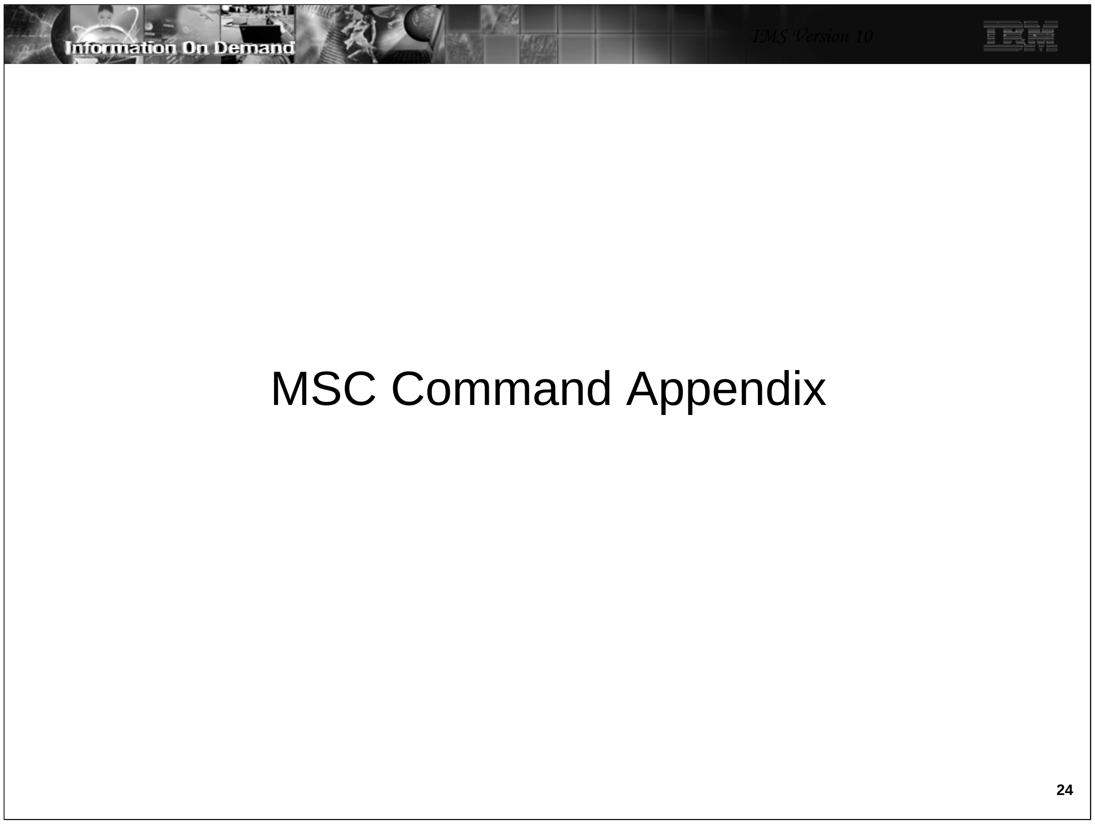
MSC Commands Summary

- Updates provide capability to dynamically add and modify MSC links
  - Addition of links requires available dummy definitions
  - Restrictions:
    - Cannot change link type: CTC, VTAM, or MTM
    - Cannot modify CTC DDNAME or address
    - Cannot change number of sessions for VTAM link
- Queries show current definitions

23

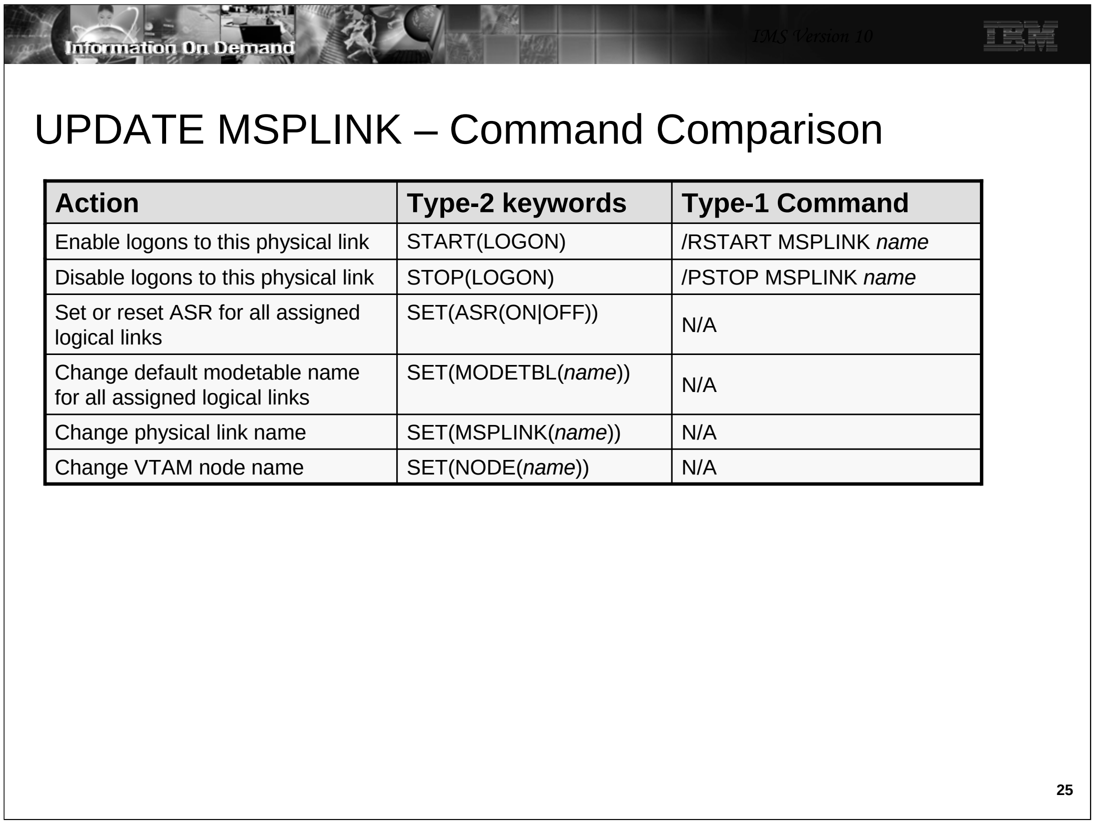
UPDATE commands for MSPLINK, MSLINK, and MSNAME may be used to dynamically add and modify MSC links. The addition of links requires available dummy definitions. This capability is most easily used with VTAM links. Dummy definitions must be of the correct link type. That is, CTC, VTAM, and MTM definitions cannot be changed to another type. For example, you cannot change a CTC MSPLINK definition to a VTAM MSPLINK. The UPDATE command cannot be used to change the DDNAME or address associated with a CTC MSPLINK definition. The SESSION= parameter on the MSPLINK system definition macro for VTAM physical links cannot be modified with the UPDATE command.

QUERY commands may be used to show the current definitions.



MSC Command Appendix

This section compares the capabilities of the QUERY and UPDATE commands for MSC resources with the capabilities of type-1 commands.

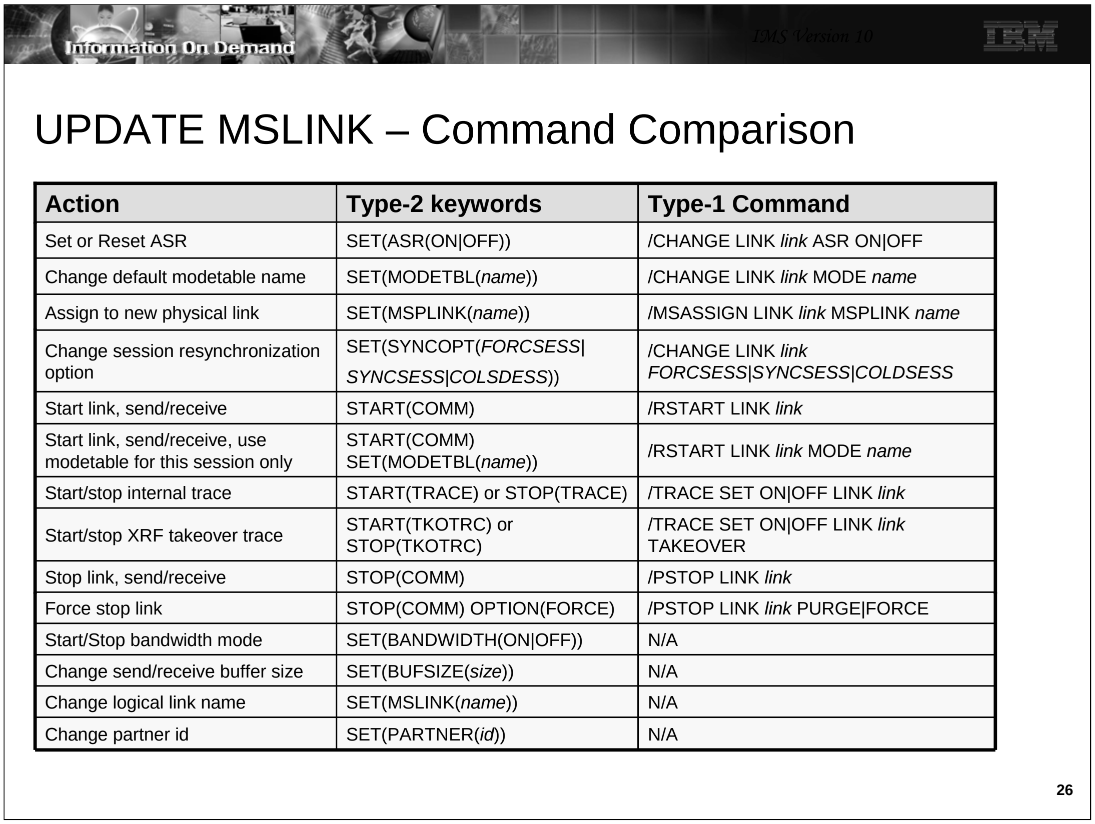

Information On Demand
IMS Version 10
IBM

UPDATE MSPLINK – Command Comparison

Action	Type-2 keywords	Type-1 Command
Enable logons to this physical link	START(LOGON)	/RSTART MSPLINK *name*
Disable logons to this physical link	STOP(LOGON)	/PSTOP MSPLINK *name*
Set or reset ASR for all assigned logical links	SET(ASR(ON|OFF))	N/A
Change default modetable name for all assigned logical links	SET(MODETBL(*name*))	N/A
Change physical link name	SET(MSPLINK(*name*))	N/A
Change VTAM node name	SET(NODE(*name*))	N/A

25

This table shows the UPD command and the comparable type-1 command. There are no type-1 commands to change the attributes.



Information On Demand
IMS Version 10

UPDATE MSLINK – Command Comparison

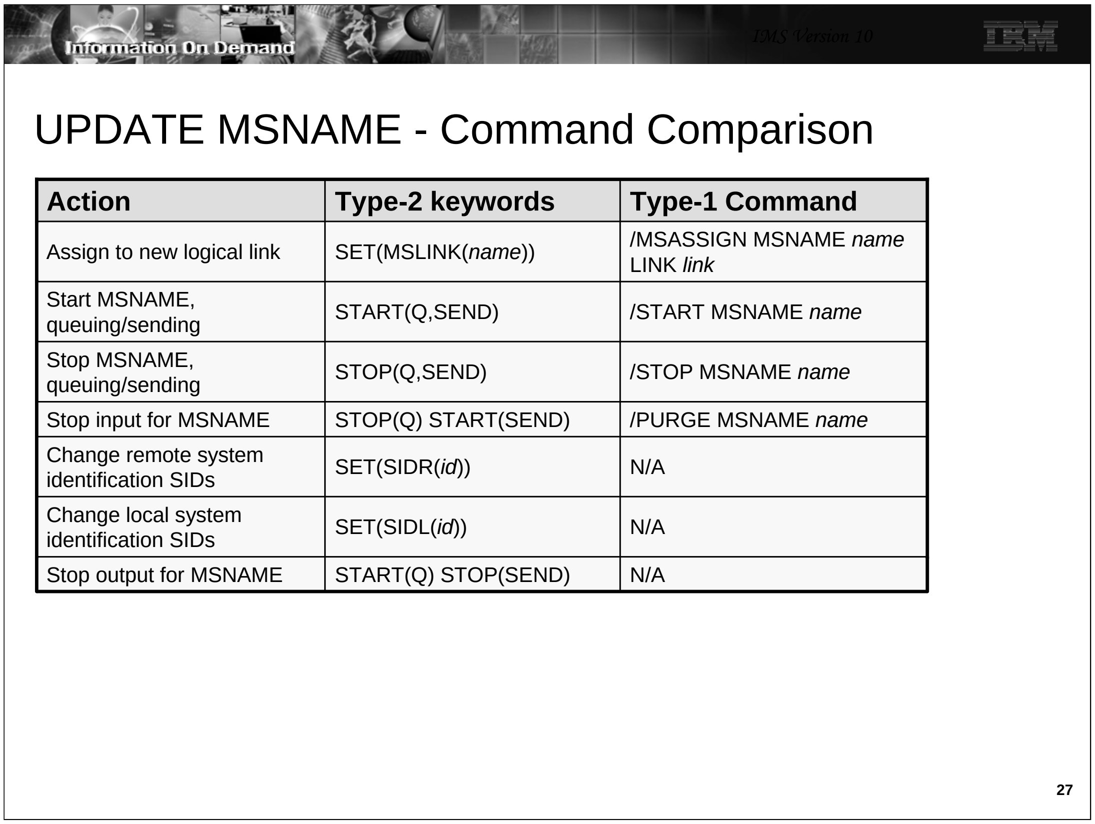
Action	Type-2 keywords	Type-1 Command
Set or Reset ASR	SET(ASR(ON|OFF))	/CHANGE LINK *link* ASR ON|OFF
Change default modetable name	SET(MODETBL(*name*))	/CHANGE LINK *link* MODE *name*
Assign to new physical link	SET(MSPLINK(*name*))	/MSASSIGN LINK *link* MSPLINK *name*
Change session resynchronization option	SET(SYNCOPT(*FORCSESS*| *SYNCSESS*|*COLSDESS*))	/CHANGE LINK *link* *FORCSESS*|*SYNCSESS*|*COLDSESS*
Start link, send/receive	START(COMM)	/RSTART LINK *link*

Start link, send/receive, use modetable for this session only	START(COMM)
	SET(MODETBL(*name*))
		/RSTART LINK *link* MODE *name*

Start/stop internal trace	START(TRACE) or STOP(TRACE)	/TRACE SET ON|OFF LINK *link*
Start/stop XRF takeover trace	START(TKOTRC) or STOP(TKOTRC)	/TRACE SET ON|OFF LINK *link* TAKEOVER
Stop link, send/receive	STOP(COMM)	/PSTOP LINK *link*
Force stop link	STOP(COMM) OPTION(FORCE)	/PSTOP LINK *link* PURGE|FORCE
Start/Stop bandwidth mode	SET(BANDWIDTH(ON|OFF))	N/A
Change send/receive buffer size	SET(BUFSIZE(*size*))	N/A
Change logical link name	SET(MSLINK(*name*))	N/A
Change partner id	SET(PARTNER(*id*))	N/A

26

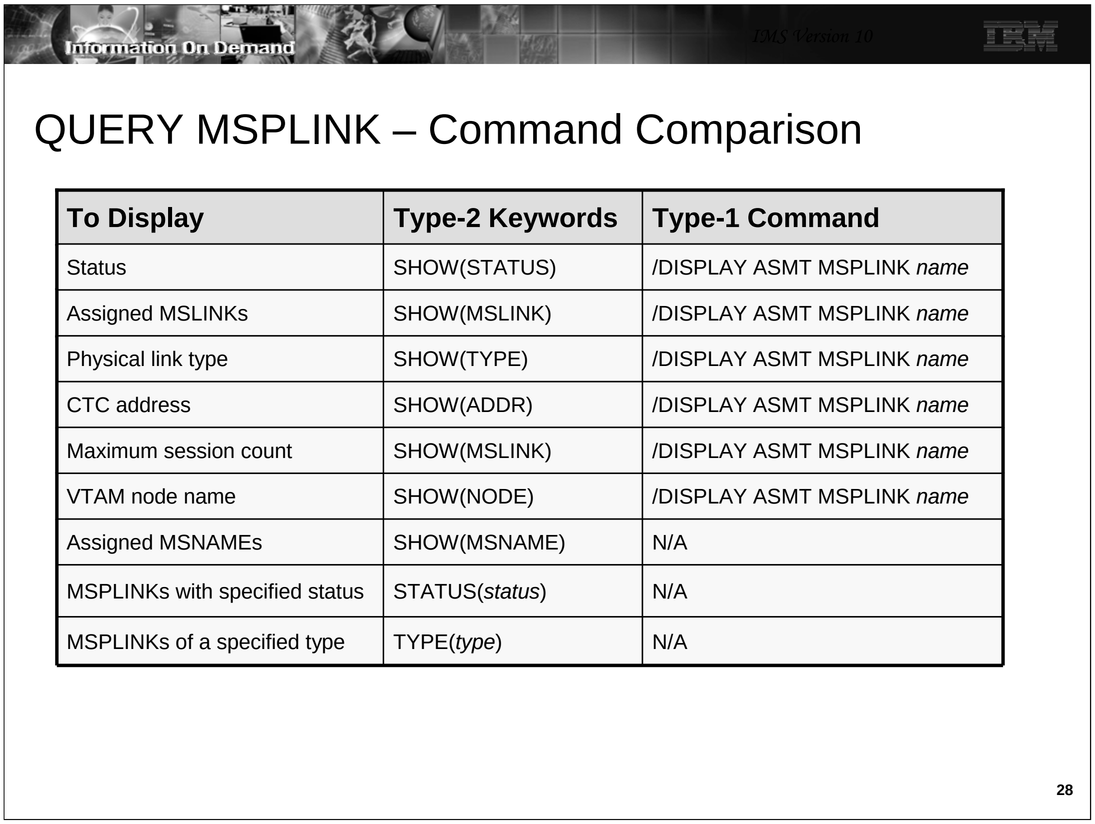
This table shows the type-1 and type-2 commands that will produce the Action shown in Column 1. There are no type-1 commands to change BANDWIDTH, BUFSIZE, MSLINK, or PARTNER.


Information On Demand
IMS Version 10

UPDATE MSNAME - Command Comparison

Action	Type-2 keywords	Type-1 Command
Assign to new logical link	SET(MSLINK(*name*))	/MSASSIGN MSNAME *name* LINK *link*
Start MSNAME, queuing/sending	START(Q,SEND)	/START MSNAME *name*
Stop MSNAME, queuing/sending	STOP(Q,SEND)	/STOP MSNAME *name*
Stop input for MSNAME	STOP(Q) START(SEND)	/PURGE MSNAME *name*
Change remote system identification SIDs	SET(SIDR(*id*))	N/A
Change local system identification SIDs	SET(SIDL(*id*))	N/A
Stop output for MSNAME	START(Q) STOP(SEND)	N/A

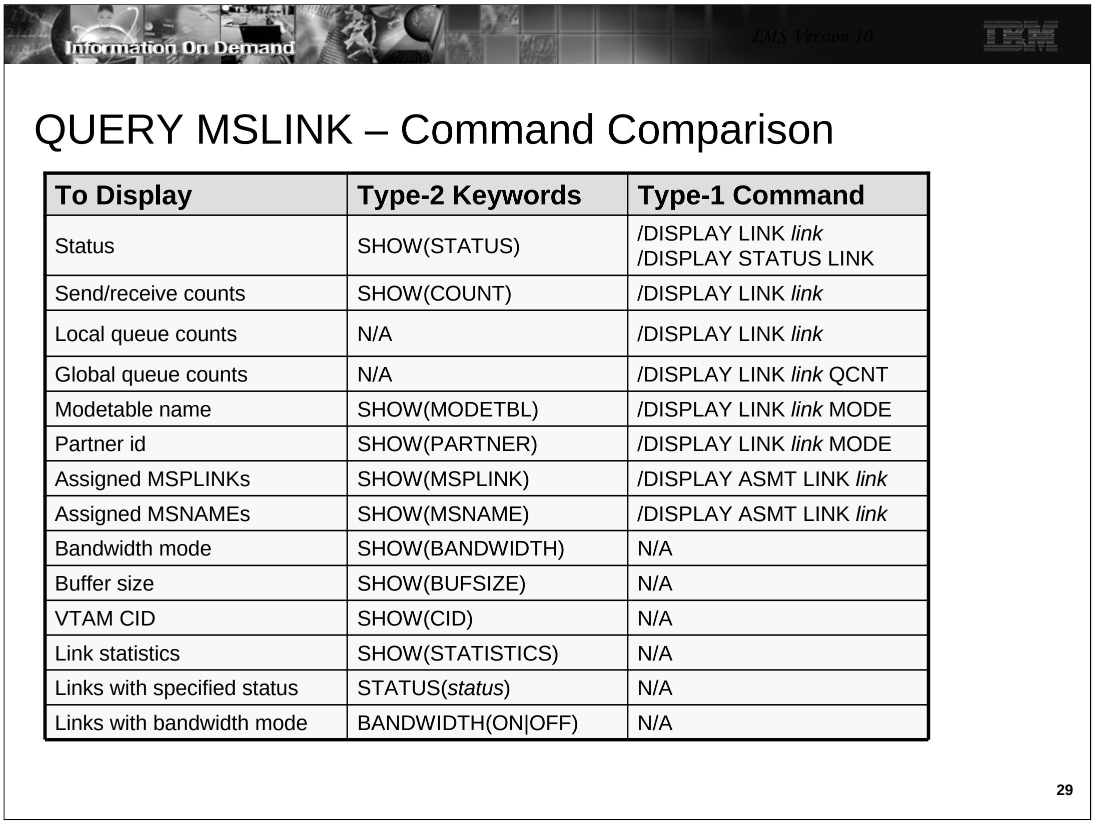
This table shows the type-1 and type-2 commands necessary to make the changes identified in column 1. Note that you cannot change the attributes with a type-1 command other than assigning it to a different logical link.



QUERY MSPLINK – Command Comparison

To Display	Type-2 Keywords	Type-1 Command
Status	SHOW(STATUS)	/DISPLAY ASMT MSPLINK *name*
Assigned MSLINKs	SHOW(MSLINK)	/DISPLAY ASMT MSPLINK *name*
Physical link type	SHOW(TYPE)	/DISPLAY ASMT MSPLINK *name*
CTC address	SHOW(ADDR)	/DISPLAY ASMT MSPLINK *name*
Maximum session count	SHOW(MSLINK)	/DISPLAY ASMT MSPLINK *name*
VTAM node name	SHOW(NODE)	/DISPLAY ASMT MSPLINK *name*
Assigned MSNAMEs	SHOW(MSNAME)	N/A
MSPLINKs with specified status	STATUS(*status*)	N/A
MSPLINKs of a specified type	TYPE(*type*)	N/A

This table shows you how to display attributes of the MSPLINK using either the type-1 /DISPLAY command or the type-2 QUERY command. For example, to display the VTAM node name of the remote IMS, submit either /DISPLAY ASMT MSPLINK name or QUERY MSPLINK NAME(xxx) SHOW(NODE). Note that the display command shows all of those attributes but QRY shows only what you ask for. There are some attributes that don't show up on the /DISPLAY command, e.g., the assigned MSNAMEs. Also you can't specify filters on the type-1 command.



Information On Demand

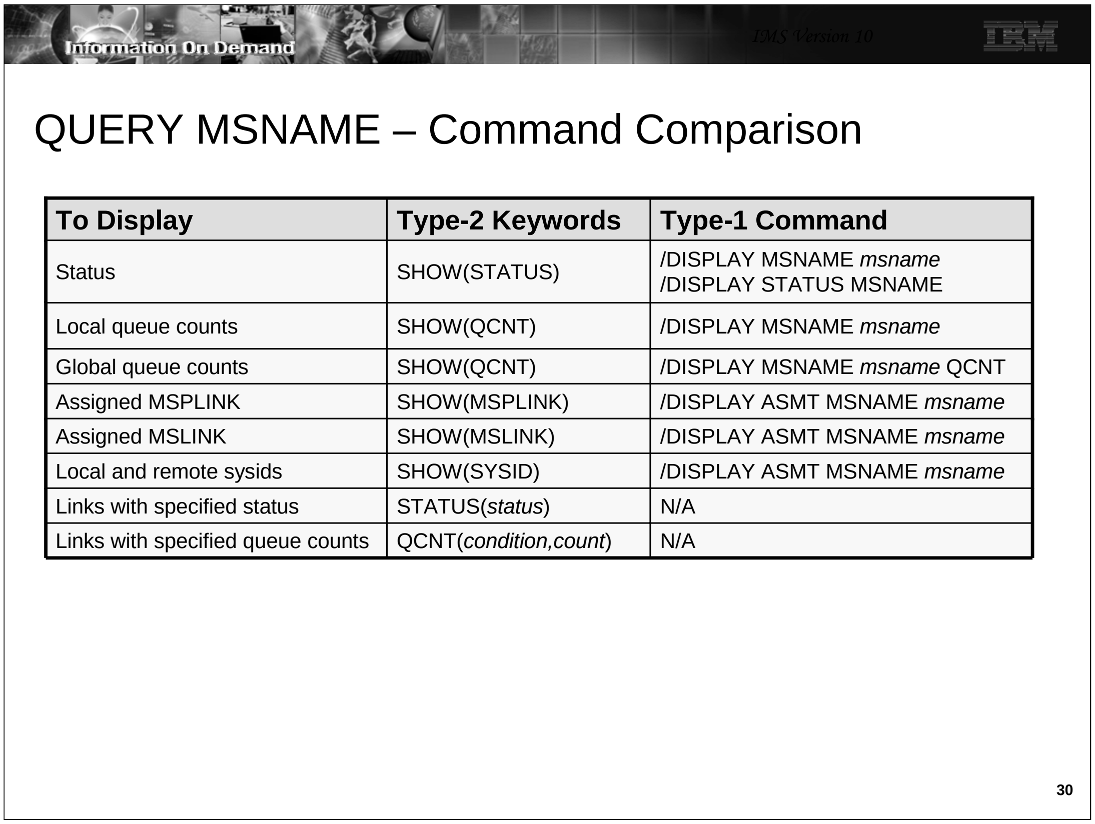
IMS Version 10

QUERY MSLINK – Command Comparison

To Display	Type-2 Keywords	Type-1 Command
Status	SHOW(STATUS)	/DISPLAY LINK *link* /DISPLAY STATUS LINK
Send/receive counts	SHOW(COUNT)	/DISPLAY LINK *link*
Local queue counts	N/A	/DISPLAY LINK *link*
Global queue counts	N/A	/DISPLAY LINK *link* QCNT
Modetable name	SHOW(MODETBL)	/DISPLAY LINK *link* MODE
Partner id	SHOW(PARTNER)	/DISPLAY LINK *link* MODE
Assigned MSPLINKs	SHOW(MSPLINK)	/DISPLAY ASMT LINK *link*
Assigned MSNAMEs	SHOW(MSNAME)	/DISPLAY ASMT LINK *link*
Bandwidth mode	SHOW(BANDWIDTH)	N/A
Buffer size	SHOW(BUFSIZE)	N/A
VTAM CID	SHOW(CID)	N/A
Link statistics	SHOW(STATISTICS)	N/A
Links with specified status	STATUS(*status*)	N/A
Links with bandwidth mode	BANDWIDTH(ON|OFF)	N/A

29

This table shows you how to display attributes of the MSLINK using either the type-1 /DISPLAY command or the type-2 QUERY command.   Note that the display link command shows the local and global queue counts whereas the QRY command does not. This is because the queue counts are shown on the QRY MSNAME command. There are several attributes that cannot be displayed using the type-1 command, for example BANDWIDTH, BUFSIZE, and the new link statistics. SHOW(STATISTICS) and SHOW(BANDWIDTH) are explained in detail in the MSC topic of TM Enhancements.

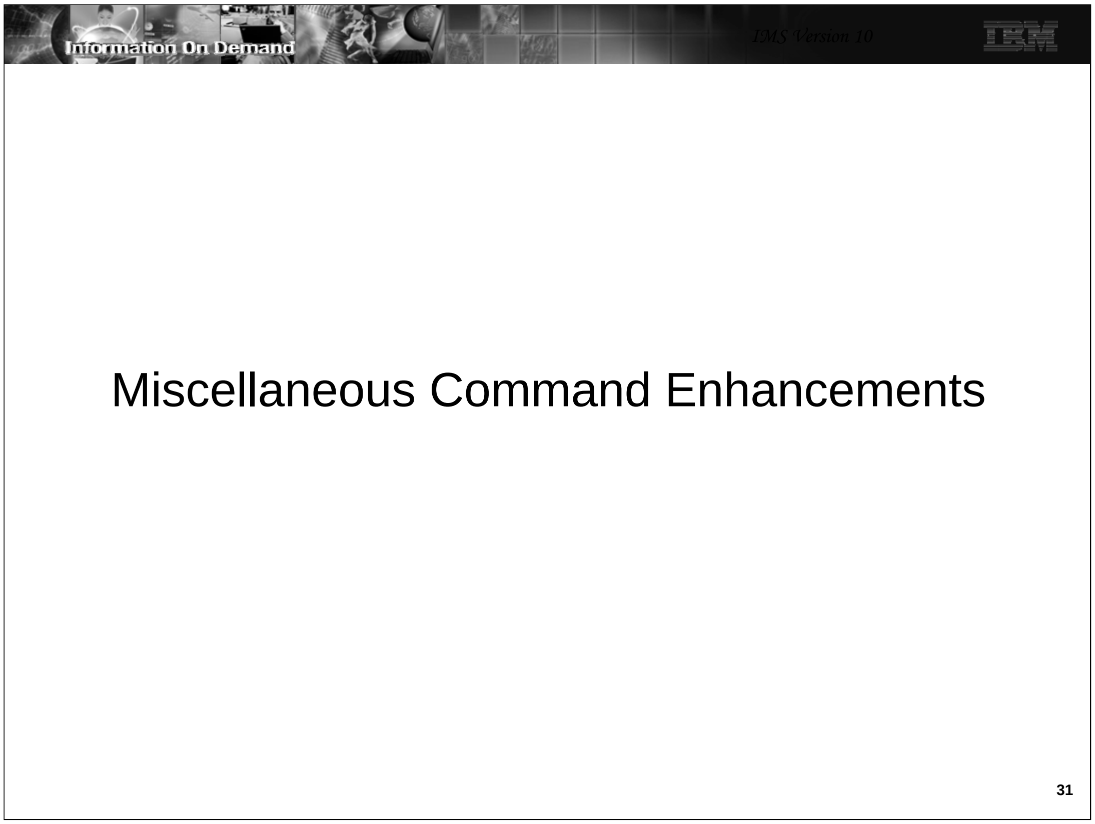

Information On Demand
IMS Version 10

QUERY MSNAME – Command Comparison

To Display	Type-2 Keywords	Type-1 Command
Status	SHOW(STATUS)	/DISPLAY MSNAME *msname* /DISPLAY STATUS MSNAME
Local queue counts	SHOW(QCNT)	/DISPLAY MSNAME *msname*
Global queue counts	SHOW(QCNT)	/DISPLAY MSNAME *msname* QCNT
Assigned MSPLINK	SHOW(MSPLINK)	/DISPLAY ASMT MSNAME *msname*
Assigned MSLINK	SHOW(MSLINK)	/DISPLAY ASMT MSNAME *msname*
Local and remote sysids	SHOW(SYSID)	/DISPLAY ASMT MSNAME *msname*
Links with specified status	STATUS(*status*)	N/A
Links with specified queue counts	QCNT(*condition,count*)	N/A

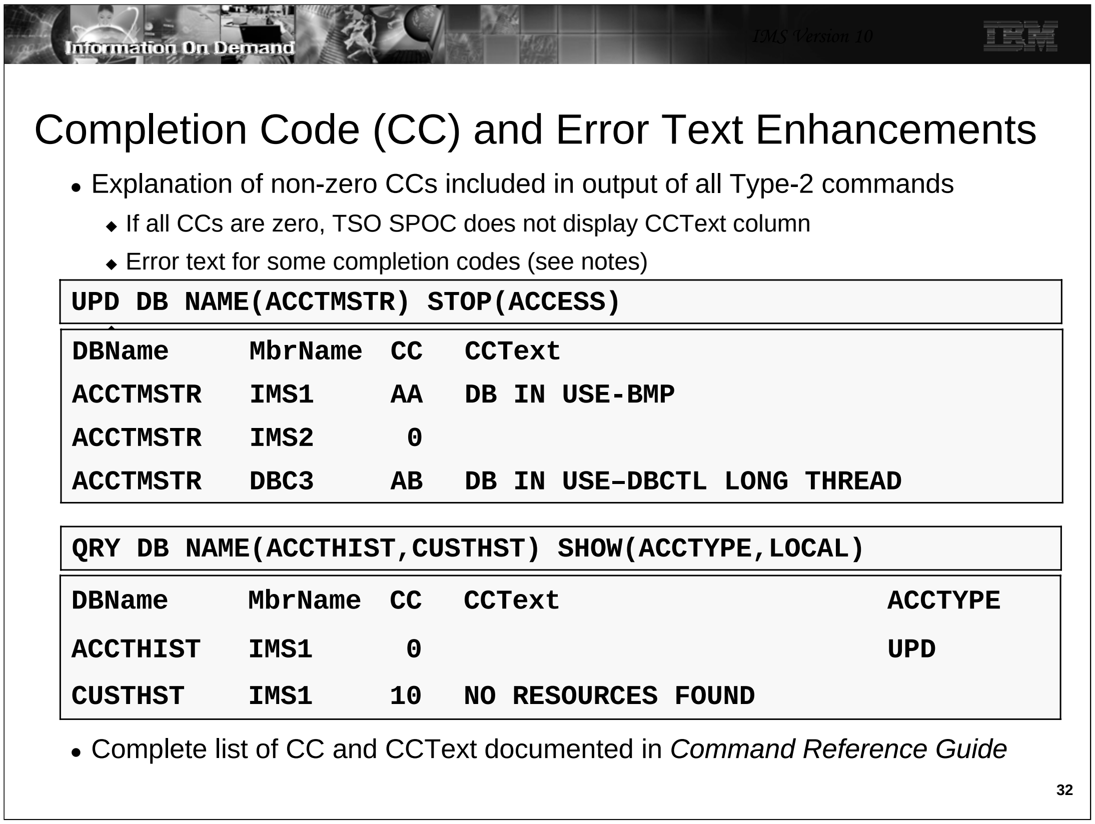
30

This table shows you how to display attributes of the MSNAME using either the type-1 /DISPLAY command or the type-2 QUERY command.



Miscellaneous Command Enhancements

This next section discusses additional command enhancements.


Information On Demand
IMS Version 10
IBM

Completion Code (CC) and Error Text Enhancements

- Explanation of non-zero CCs included in output of all Type-2 commands
  - If all CCs are zero, TSO SPOC does not display CCText column
  - Error text for some completion codes (see notes)

```
UPD DB NAME(ACCTMSTR) STOP(ACCESS)
```

```
DBName     MbrName  CC   CCText
ACCTMSTR   IMS1     AA   DB IN USE-BMP
ACCTMSTR   IMS2      0
ACCTMSTR   DBC3     AB   DB IN USE–DBCTL LONG THREAD
```

```
QRY DB NAME(ACCTHIST,CUSTHST) SHOW(ACCTYPE,LOCAL)
```

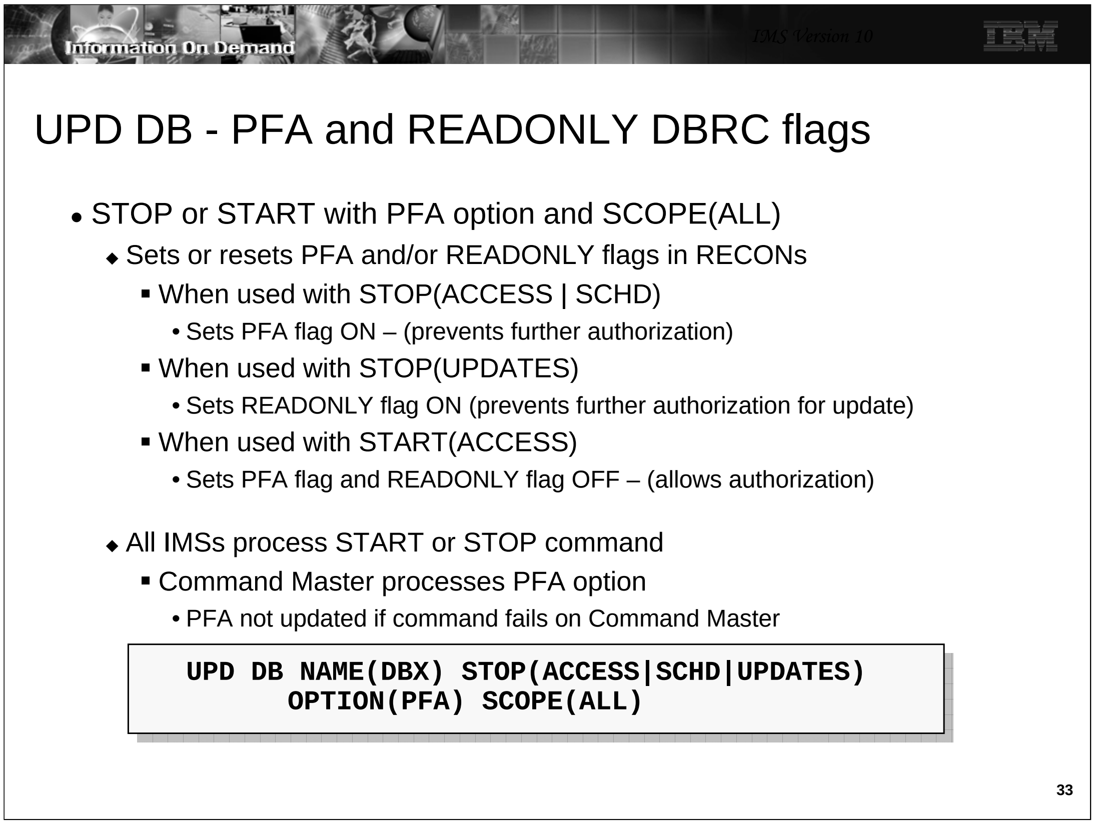
```
DBName     MbrName  CC   CCText                    ACCTYPE
ACCTHIST   IMS1      0                             UPD
CUSTHST    IMS1     10   NO RESOURCES FOUND
```

- Complete list of CC and CCText documented in *Command Reference Guide*

The chart shows an example of the SPOC displaying non-zero completion codes and the CCText returned by IMS with that CC for Type-2 commands. . If all CCs are equal to zero, the CCText column is omitted from the output display. The Command Reference manuals show all of the condition codes and the text for any command.

Some UPDATE and DELETE commands may also include 'Error Text'. For example, a DELETE DB command may include error text which identifies a program which prevented the command from being processed because the program was using the database. 'Error Text' is an additional column. The following is an example of a command response with 'Error Text'.

```
DELETE DB NAME(ACCTMSTR)

DBName    MbrName   CC CCText                     ErrorText
ACCTMSTR  IMS1      11 REFERENCED BY PROGRAM      ACCTUPDT
```



Information On Demand
IMS Version 10
IBM

UPD DB - PFA and READONLY DBRC flags

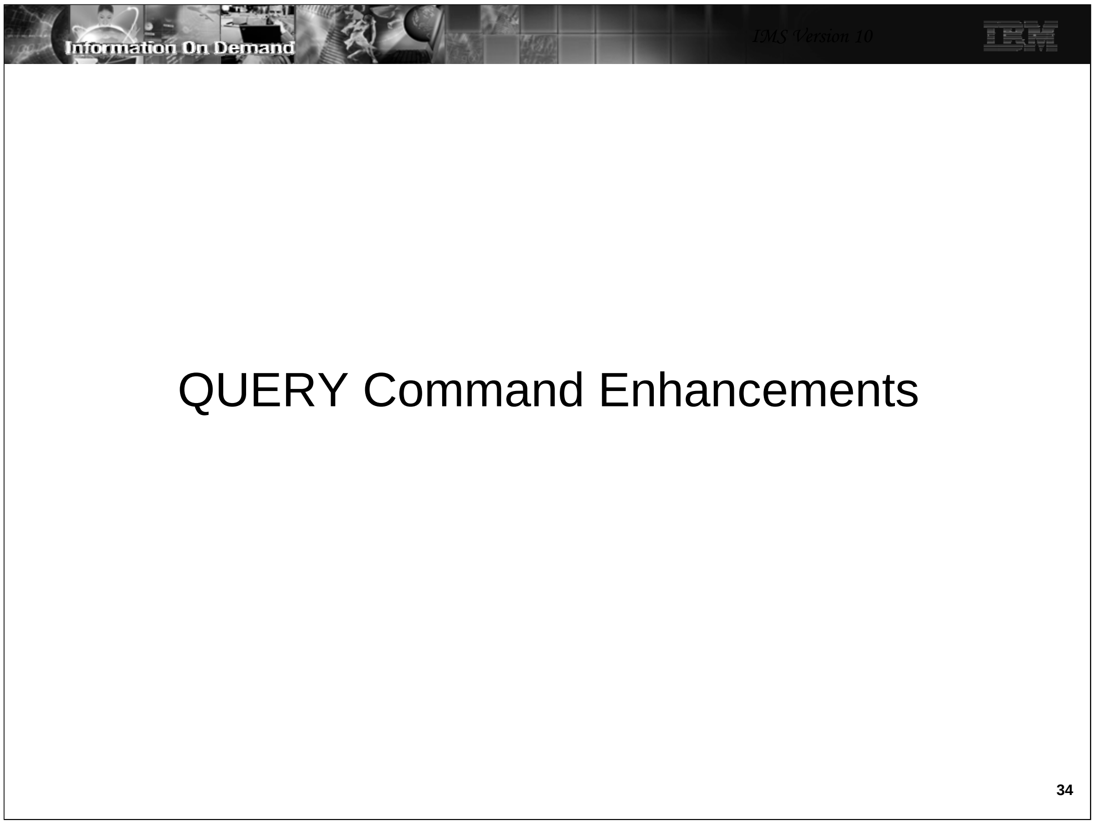
- STOP or START with PFA option and SCOPE(ALL)
  - Sets or resets PFA and/or READONLY flags in RECONs
    - When used with STOP(ACCESS | SCHD)
      - Sets PFA flag ON – (prevents further authorization)
    - When used with STOP(UPDATES)
      - Sets READONLY flag ON (prevents further authorization for update)
    - When used with START(ACCESS)
      - Sets PFA flag and READONLY flag OFF – (allows authorization)
  - All IMSs process START or STOP command
    - Command Master processes PFA option
      - PFA not updated if command fails on Command Master

```
UPD DB NAME(DBX) STOP(ACCESS|SCHD|UPDATES)
       OPTION(PFA) SCOPE(ALL)
```

33

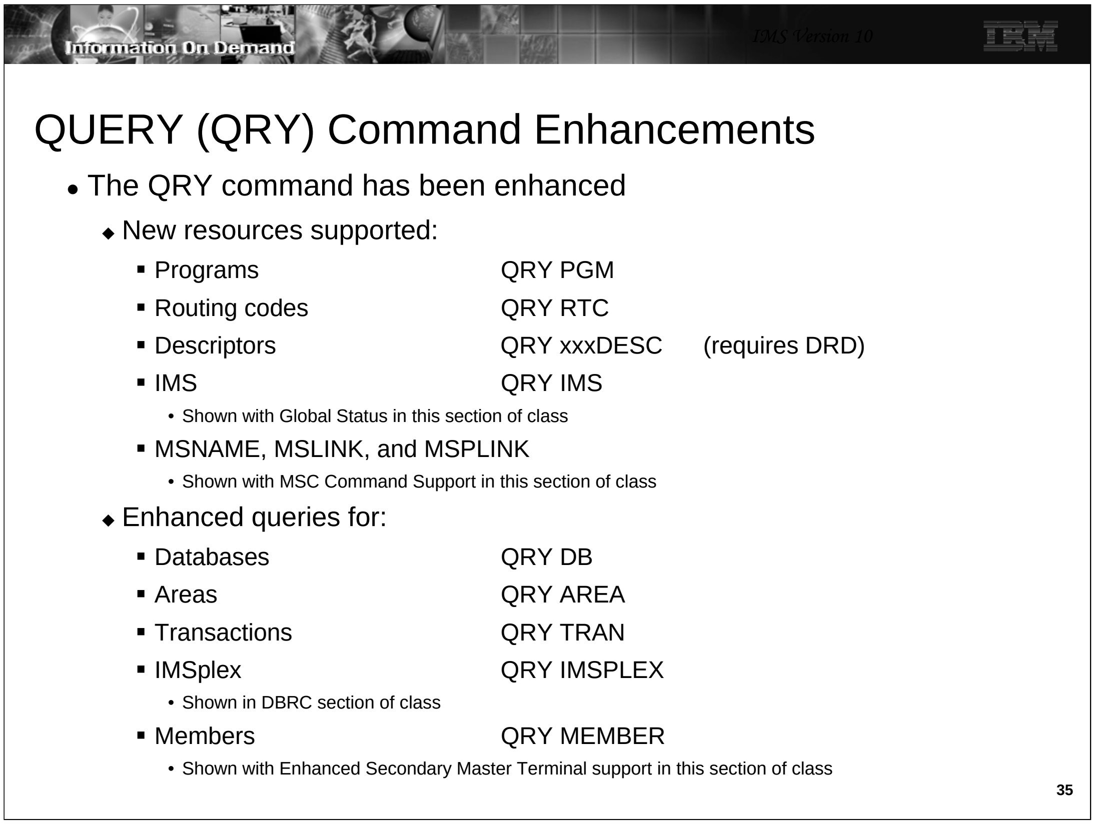
Some new options have been added to the UPD DB command. The new options set the PFA (Prevent Further Authorization) or READONLY flag in the RECONs. This is what the /DBR, /DBD, /STA, and /STO commands for databases or areas with the GLOBAL parameter do. For these type-1 commands, the flags are set when GLOBAL is specified unless the NOPFA parameter is also specified. For the UPD command, the PFA parameter must be specified along with SCOPE(ALL) for the flags to be affected.

The UPD command OPTION(PFA) sets or resets the PFA or READONLY flag in DBRC. Specifying the PFA option with a STOP(ACCESS) or STOP(SCHD) command sets the PFA (prohibit further authorization) flag in the RECONs. Specifying the PFA option with a STOP(UPDATES) command sets the READONLY flag in the RECONs. The READONLY prohibits update and exclusive authorizations. Both flags are set off with START(ACCESS) command when the PFA option is used.



QUERY Command Enhancements

This section addresses enhancements to the QRY command.


Information On Demand
IMS Version 10

QUERY (QRY) Command Enhancements

- The QRY command has been enhanced
  - New resources supported:
    - Programs — QRY PGM
    - Routing codes — QRY RTC
    - Descriptors — QRY xxxDESC (requires DRD)
    - IMS — QRY IMS
      - Shown with Global Status in this section of class
    - MSNAME, MSLINK, and MSPLINK
      - Shown with MSC Command Support in this section of class
  - Enhanced queries for:
    - Databases — QRY DB
    - Areas — QRY AREA
    - Transactions — QRY TRAN
    - IMSplex — QRY IMSPLEX
      - Shown in DBRC section of class
    - Members — QRY MEMBER
      - Shown with Enhanced Secondary Master Terminal support in this section of class

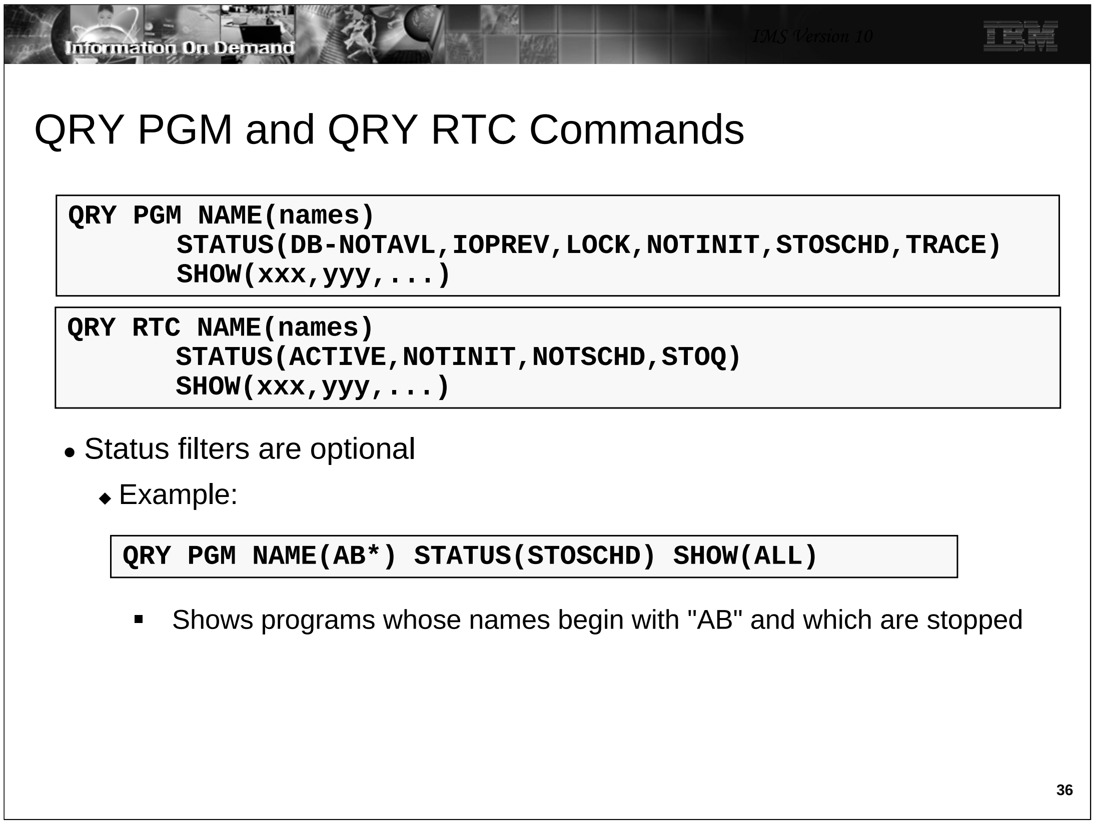
35

The QUERY (or QRY) command has been enhanced to support new resources and extend the command for resources supported in previous releases.

The QRY command for descriptors is rejected if DRD is not active. The other QRY commands may be used with or without DRD.

The only change in the QUERY TRAN command is the way of filtering requests for conversational, fast path, remote, and response mode. Previously, these were done with STATUS(xxxxxxx) where xxxxxxxx was CONV, FPE, FPN, RMT, or RESP. There are new alternatives in IMS V10. The new alternatives are CONV(Y|N), FP(N|E|P), REMOTE(Y|N)m and RESP(Y|N).

Some of the QRY command enhancements are discussed in other sections of the class.



Information On Demand
IMS Version 10

QRY PGM and QRY RTC Commands

```
QRY PGM NAME(names)
       STATUS(DB-NOTAVL,IOPREV,LOCK,NOTINIT,STOSCHD,TRACE)
       SHOW(xxx,yyy,...)
```

```
QRY RTC NAME(names)
       STATUS(ACTIVE,NOTINIT,NOTSCHD,STOQ)
       SHOW(xxx,yyy,...)
```

- Status filters are optional
  - Example:

```
QRY PGM NAME(AB*) STATUS(STOSCHD) SHOW(ALL)
```

  - Shows programs whose names begin with "AB" and which are stopped

36

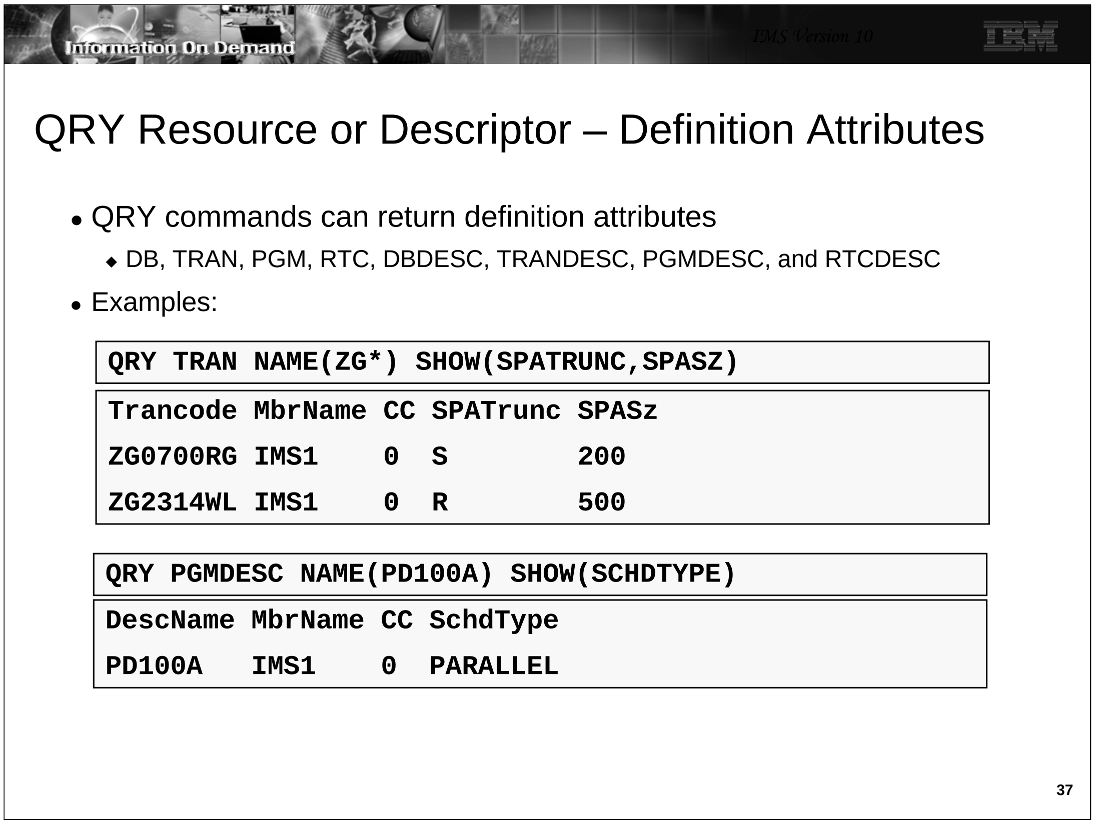
QRY PGM and QRY RTC are new in IMS V10.

The following statuses may be specified for QRY PGM:

DB-NOTAVL	At least one database is not available
IOPREV	BMP with GSAM can’t complete due to IO Prevention
LOCK	Program is LOCKED
NOTINIT	Program is not initialized – PSB or DMB not in ACBLIB
STOSCHD	Program scheduling has been stopped - UPD PGM xxx STOP(SCHD)
TRACE	Program trace is on - UPD PGM xxx START(TRACE)

The following statuses may specified for QRY RTC:

ACTIVE	RTC is active
NOTINIT	RTC is not initialized and can’t be used
NOTSCHD	RTC is not scheduled
STOQ	Routing code is stopped



QRY Resource or Descriptor – Definition Attributes

- QRY commands can return definition attributes
  - DB, TRAN, PGM, RTC, DBDESC, TRANDESC, PGMDESC, and RTCDESC
- Examples:

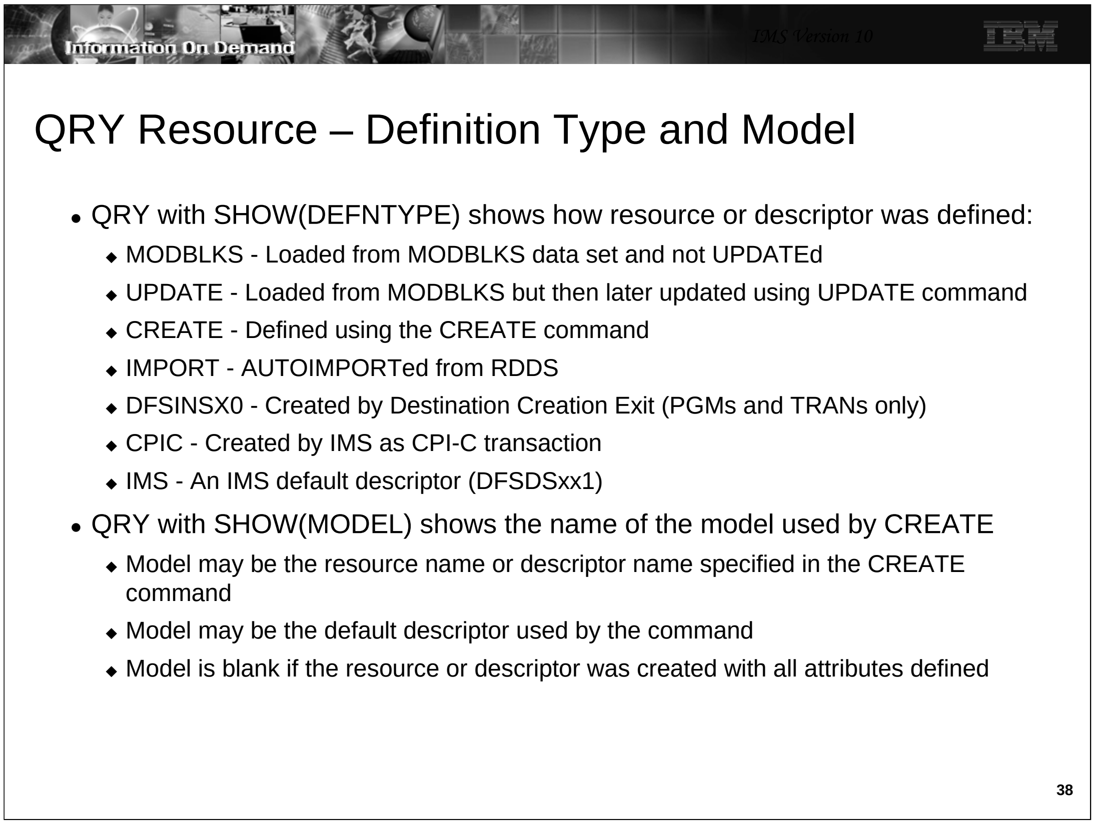
```
QRY TRAN NAME(ZG*) SHOW(SPATRUNC,SPASZ)
```

```
Trancode MbrName CC SPATrunc SPASz
ZG0700RG IMS1    0  S        200
ZG2314WL IMS1    0  R        500
```

```
QRY PGMDESC NAME(PD100A) SHOW(SCHDTYPE)
```

```
DescName MbrName CC SchdType
PD100A   IMS1    0  PARALLEL
```

QUERY commands may be used to show definition attributes for databases, programs, transactions, routing codes and their descriptors. These attributes may have been defined through system definition or through the use of DRD.


Information On Demand
IMS Version 10
IBM

QRY Resource – Definition Type and Model

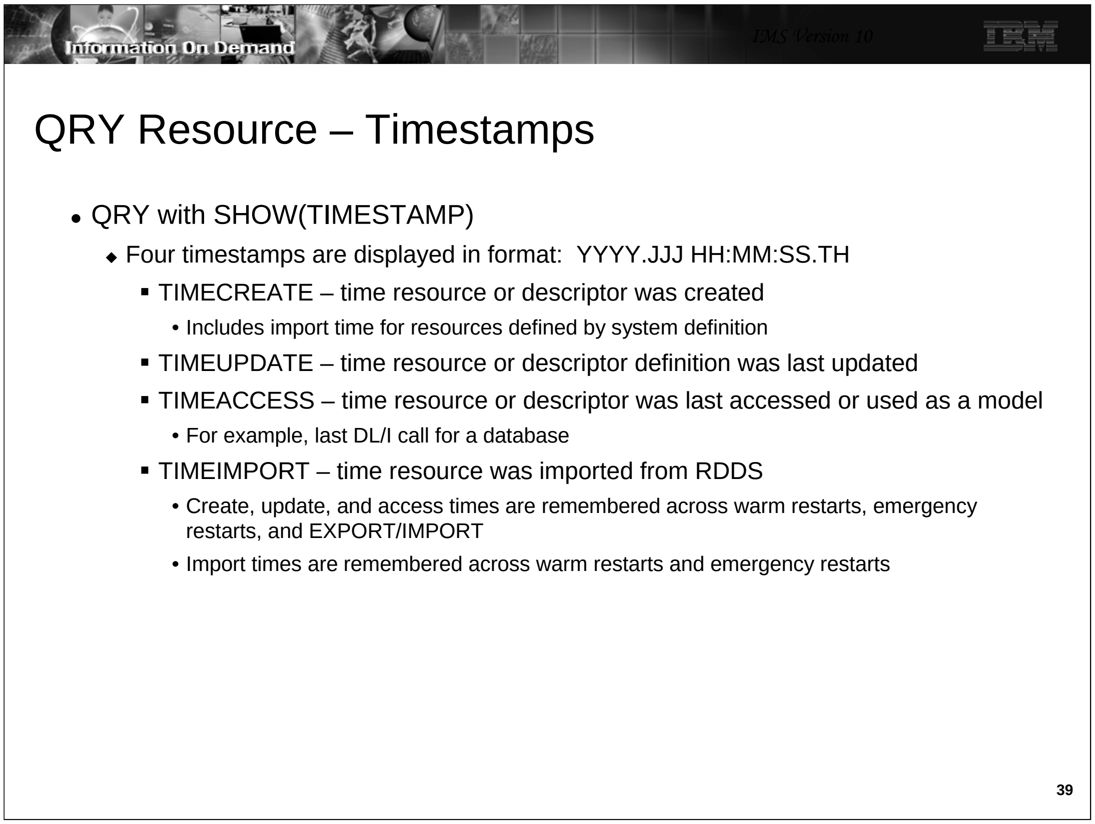
- QRY with SHOW(DEFNTYPE) shows how resource or descriptor was defined:
  - MODBLKS - Loaded from MODBLKS data set and not UPDATEd
  - UPDATE - Loaded from MODBLKS but then later updated using UPDATE command
  - CREATE - Defined using the CREATE command
  - IMPORT - AUTOIMPORTed from RDDS
  - DFSINSX0 - Created by Destination Creation Exit (PGMs and TRANs only)
  - CPIC - Created by IMS as CPI-C transaction
  - IMS - An IMS default descriptor (DFSDSxx1)
- QRY with SHOW(MODEL) shows the name of the model used by CREATE
  - Model may be the resource name or descriptor name specified in the CREATE command
  - Model may be the default descriptor used by the command
  - Model is blank if the resource or descriptor was created with all attributes defined

38

For all resources and descriptors, there are several new attributes that can be shown.

The first is definition type which shows how that resource or descriptor was created. The slide shows the possibilities. CREATE and IMPORT may be returned for both resources and descriptors. MODBLKS, UPDATE, DFSINSX0, and CPIC may be returned only for resources. IMS may be returned only for descriptors.

The second is the name of the model, if any, used by the CREATE command which created the definition.



Information On Demand

IMS Version 10

QRY Resource – Timestamps

- QRY with SHOW(TIMESTAMP)
  - Four timestamps are displayed in format: YYYY.JJJ HH:MM:SS.TH
    - TIMECREATE – time resource or descriptor was created
      - Includes import time for resources defined by system definition
    - TIMEUPDATE – time resource or descriptor definition was last updated
    - TIMEACCESS – time resource or descriptor was last accessed or used as a model
      - For example, last DL/I call for a database
    - TIMEIMPORT – time resource was imported from RDDS
      - Create, update, and access times are remembered across warm restarts, emergency restarts, and EXPORT/IMPORT
      - Import times are remembered across warm restarts and emergency restarts

39

Four timestamps can be displayed. These are the time the resource was created, the time it was last updated, the time is was last accessed, and the time it was imported from an RDDS.

TIMECREATE is the time of the CREATE command, the time the definition was IMPORTed from RDDS, the time the definition was loaded from MODBLKS, or the time a transaction or program definition was created by DFSINSX0.

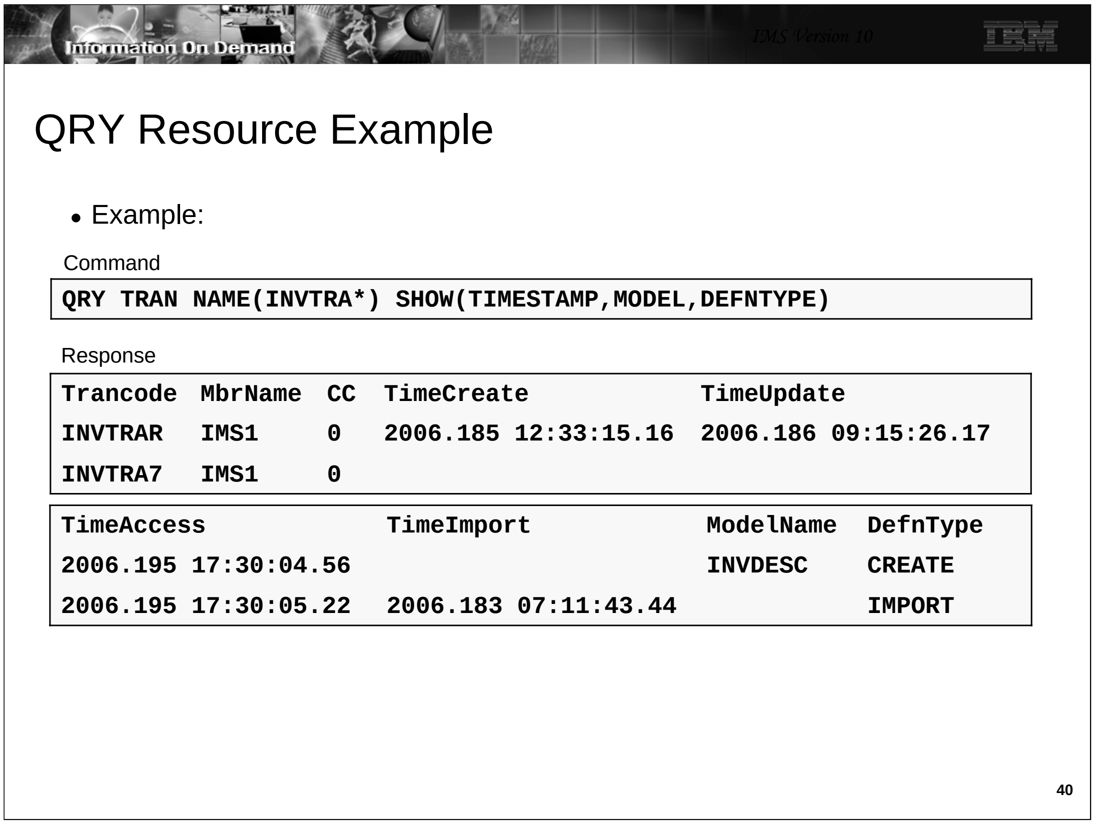
TIMEUPDATE is the time that an UPDATE command that updated a definitional attribute, not a status.

TIMEACCESS may be used to understand the last time a resource or descriptor was accessed. For resources and descriptors the meaning of TIMEACCESS is:

DB	time DB was last accessed by an application program (DL/I call)
PGM	time program (PSB) was last scheduled
TRAN	time message was last enqueued or dequeued by a program
RTC	time message was last enqueued to BALG using this routing code
DESC	time descriptor last used as a model in CREATE command

The create, update and access times are remembered across warm start, emergency restart, EXPORT and IMPORT. Import times are remembered across warm and emergency restarts.

The updating of the last access time is not logged between system checkpoints. After a restart, the last access time reflects the time recorded in the restart checkpoint log records. For warm starts this is the time recorded in the shutdown checkpoint. For emergency restarts, any access between the last system checkpoint and the failure of the IMS system will not be reflected in the information available to the restarted IMS system. For cold starts that import definitions from the RDDS, the times will be those in the RDDS. This will be the last access times at the time of the last export.



QRY Resource Example

- Example:

Command

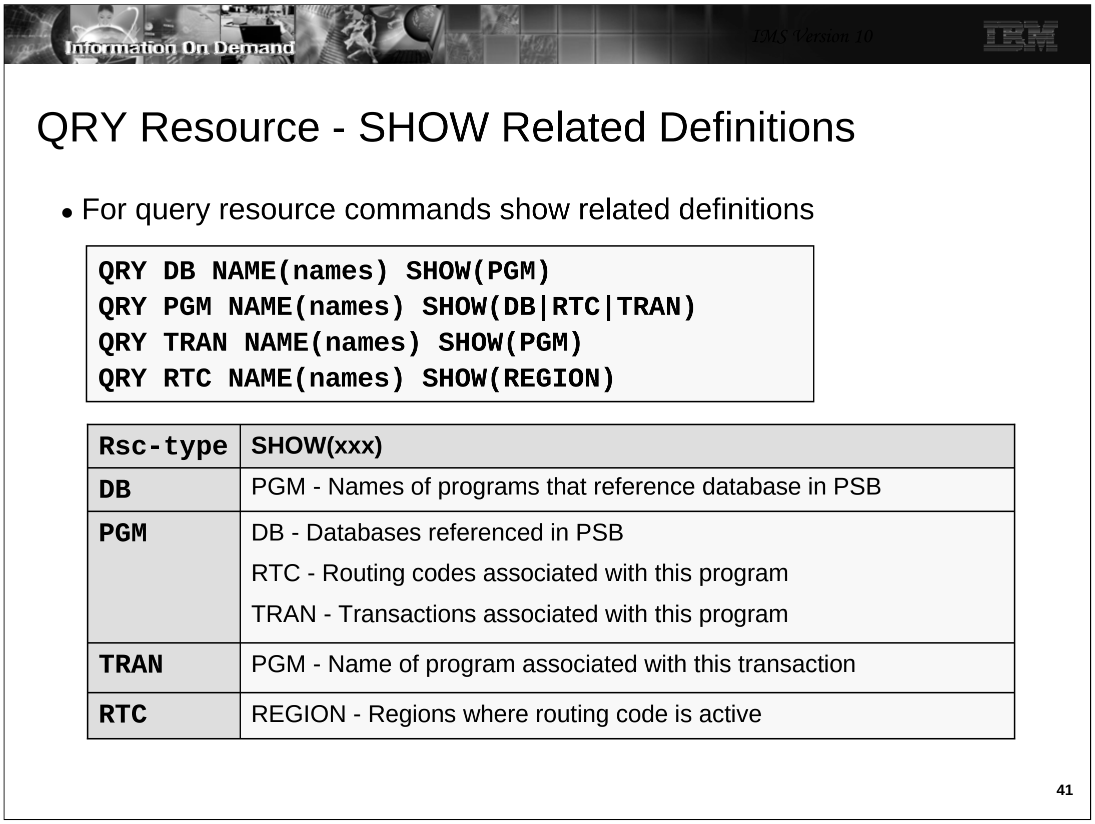
```
QRY TRAN NAME(INVTRA*) SHOW(TIMESTAMP,MODEL,DEFNTYPE)
```

Response

```
Trancode  MbrName  CC  TimeCreate           TimeUpdate
INVTRAR   IMS1     0   2006.185 12:33:15.16  2006.186 09:15:26.17
INVTRA7   IMS1     0
```

```
TimeAccess             TimeImport             ModelName  DefnType
2006.195 17:30:04.56                          INVDESC    CREATE
2006.195 17:30:05.22   2006.183 07:11:43.44              IMPORT
```

Timestamps, model, and definition type have been added to the QRY command SHOW parameters. The example above is what would be shown by the TSO SPOC. The second part of the example above would be displayed by scrolling to the right.



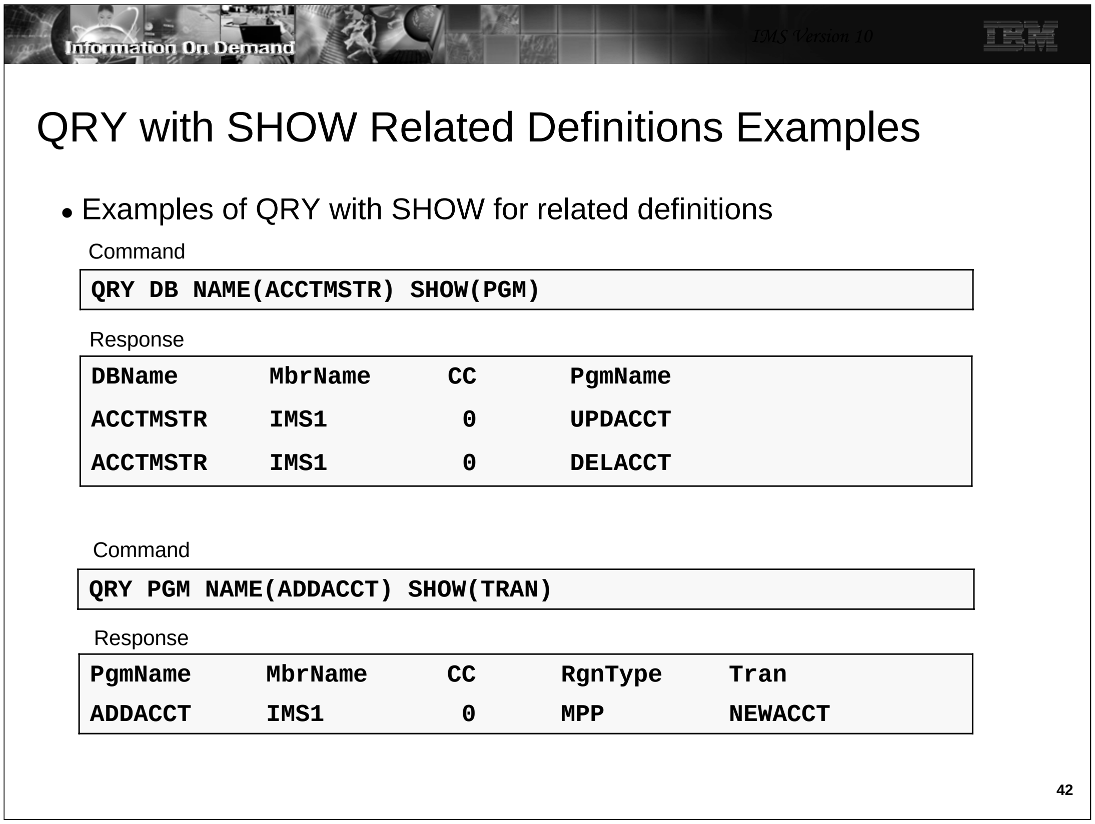
QRY Resource - SHOW Related Definitions

- For query resource commands show related definitions

```
QRY DB NAME(names) SHOW(PGM)
QRY PGM NAME(names) SHOW(DB|RTC|TRAN)
QRY TRAN NAME(names) SHOW(PGM)
QRY RTC NAME(names) SHOW(REGION)
```

Rsc-type	SHOW(xxx)
DB	PGM - Names of programs that reference database in PSB
PGM	DB - Databases referenced in PSB RTC - Routing codes associated with this program TRAN - Transactions associated with this program
TRAN	PGM - Name of program associated with this transaction
RTC	REGION - Regions where routing code is active

For resource types, you can show the related resources. These are shown here. Queries for databases accessed by programs and programs which access databases access ACBLIB for this information.



QRY with SHOW Related Definitions Examples

- Examples of QRY with SHOW for related definitions

Command

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

Response

```
DBName      MbrName     CC      PgmName
ACCTMSTR    IMS1         0      UPDACCT
ACCTMSTR    IMS1         0      DELACCT
```

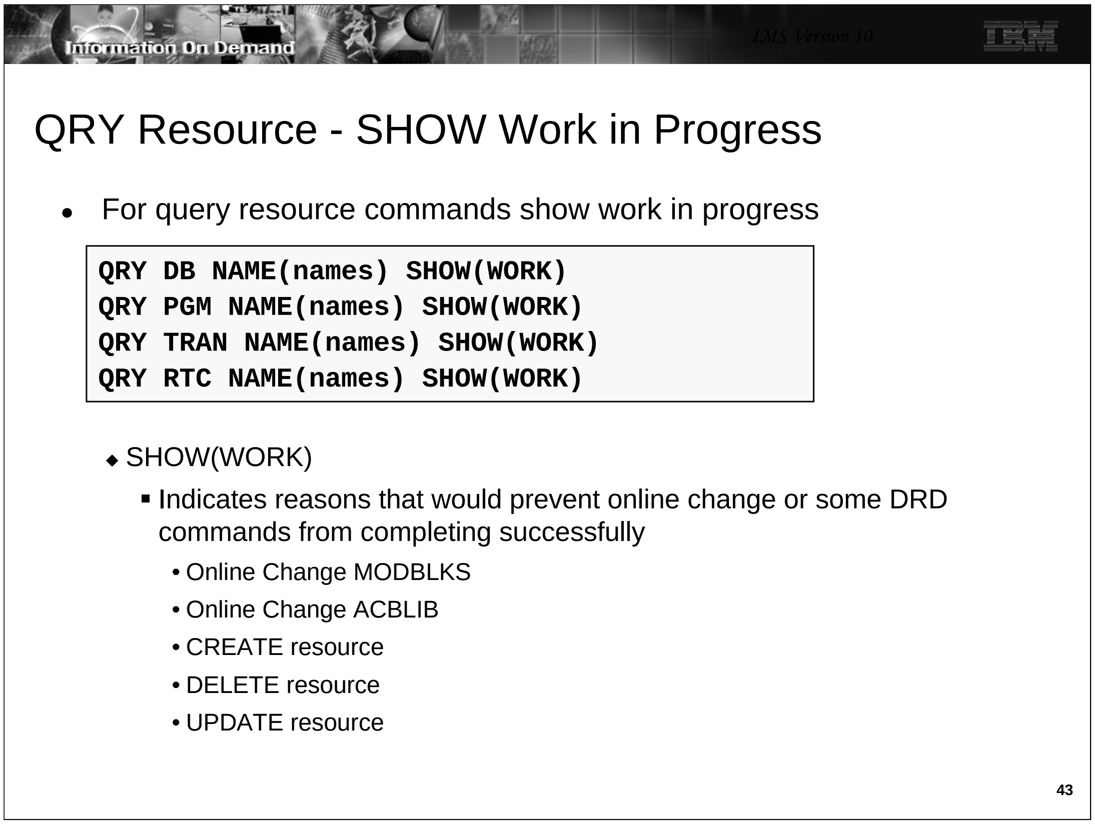
Command

```
QRY PGM NAME(ADDACCT) SHOW(TRAN)
```

Response

```
PgmName     MbrName     CC      RgnType     Tran
ADDACCT     IMS1         0      MPP         NEWACCT
```

These are examples of QUERY commands for the programs which access database ACCTMSTR and for the transactions which invoke program ADDACCT.


Information On Demand
IMS Version 10
IBM

QRY Resource - SHOW Work in Progress

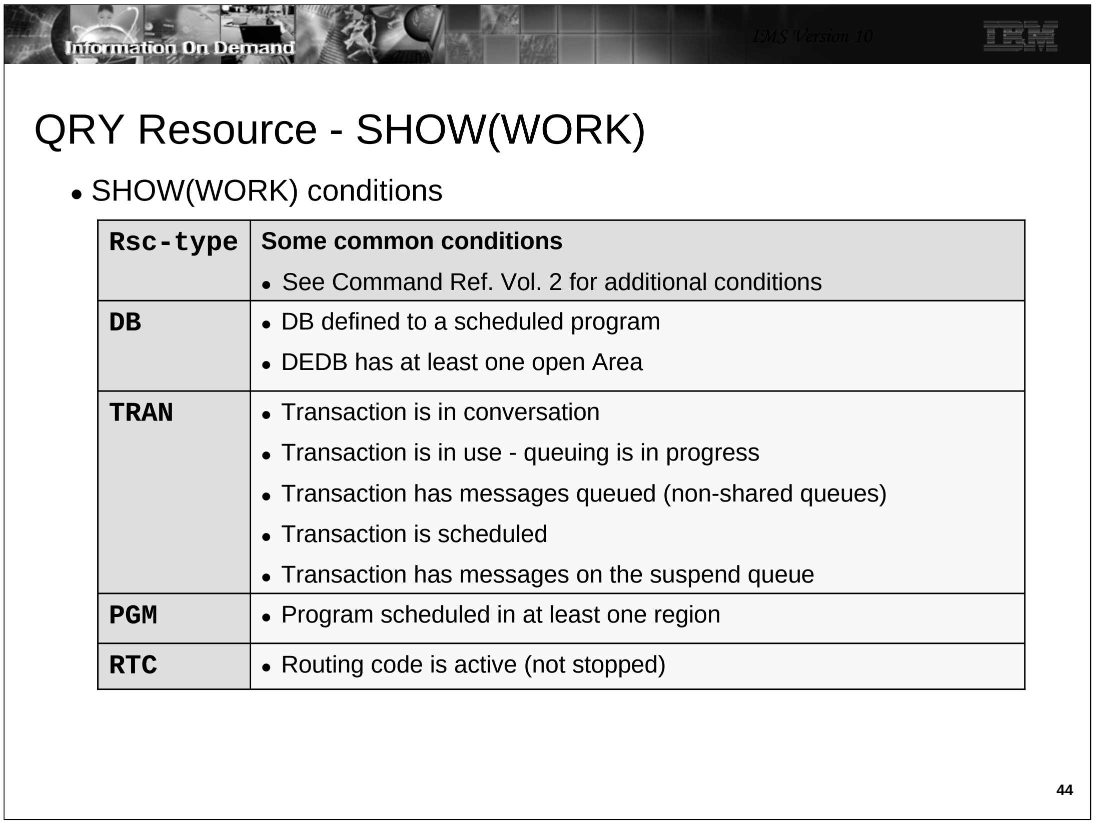
- For query resource commands show work in progress

```
QRY DB NAME(names) SHOW(WORK)
QRY PGM NAME(names) SHOW(WORK)
QRY TRAN NAME(names) SHOW(WORK)
QRY RTC NAME(names) SHOW(WORK)
```

- SHOW(WORK)
  - Indicates reasons that would prevent online change or some DRD commands from completing successfully
    - Online Change MODBLKS
    - Online Change ACBLIB
    - CREATE resource
    - DELETE resource
    - UPDATE resource

43

For all resource types, you can show the WORK status. This is typically a status that would keep OLC or DRD from completing successfully.

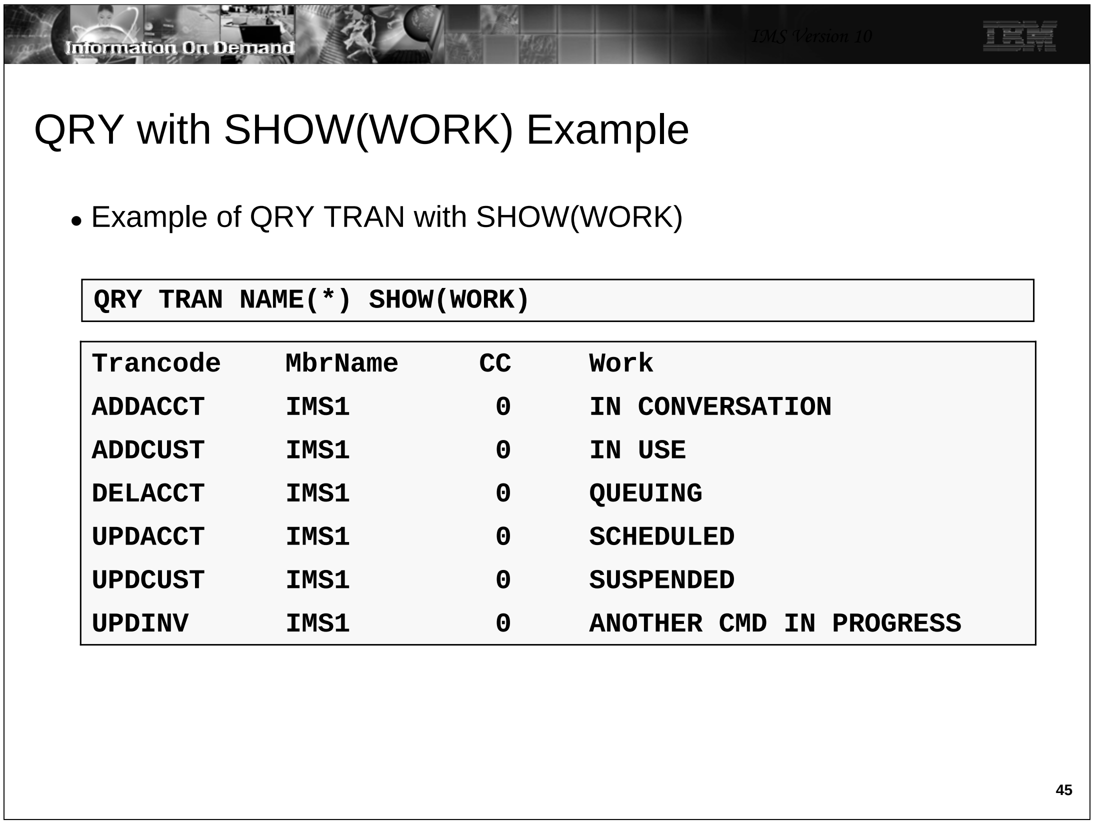

Information On Demand
IMS Version 10
IBM

QRY Resource - SHOW(WORK)

- SHOW(WORK) conditions

Rsc-type	**Some common conditions** • See Command Ref. Vol. 2 for additional conditions
DB	• DB defined to a scheduled program • DEDB has at least one open Area
TRAN	• Transaction is in conversation • Transaction is in use - queuing is in progress • Transaction has messages queued (non-shared queues) • Transaction is scheduled • Transaction has messages on the suspend queue
PGM	• Program scheduled in at least one region
RTC	• Routing code is active (not stopped)

This table shows some common conditions that would be shown in the response to a QRY command with SHOW(WORK). All of the possible responses for this field are shown in the Command Reference Volume 2 manual.



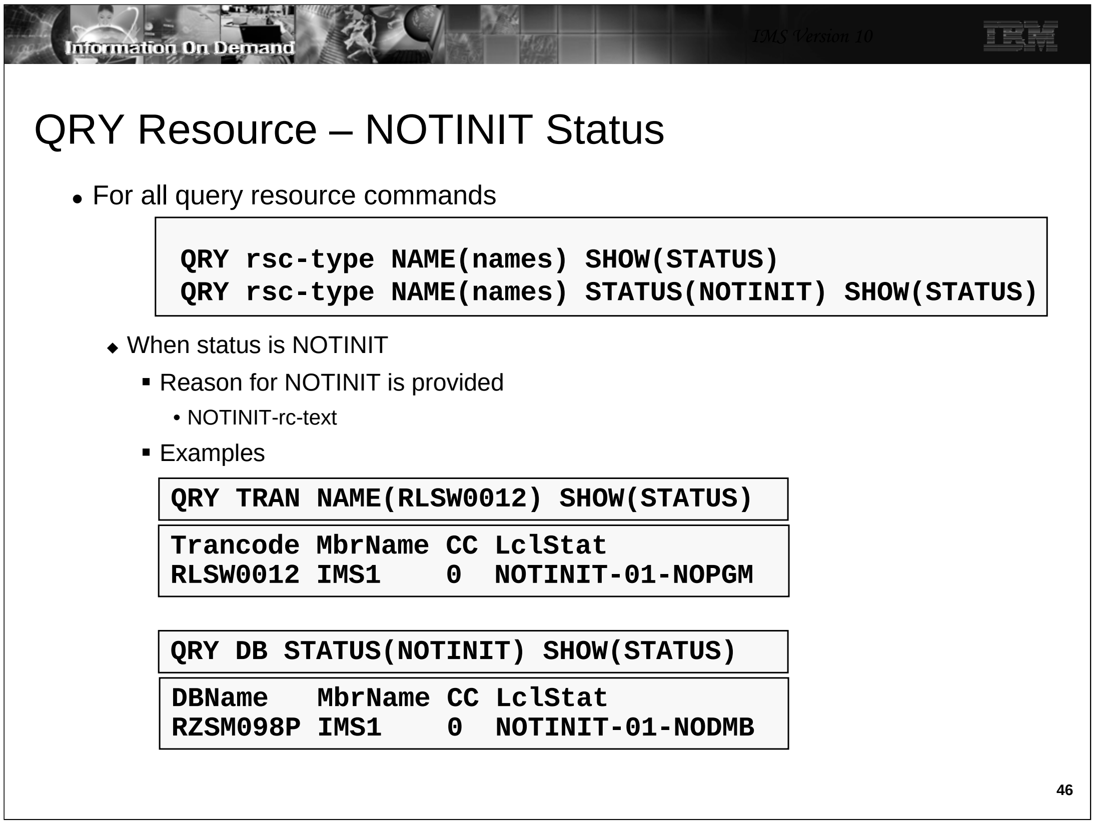
QRY with SHOW(WORK) Example

- Example of QRY TRAN with SHOW(WORK)

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

```
Trancode    MbrName     CC     Work
ADDACCT     IMS1         0     IN CONVERSATION
ADDCUST     IMS1         0     IN USE
DELACCT     IMS1         0     QUEUING
UPDACCT     IMS1         0     SCHEDULED
UPDCUST     IMS1         0     SUSPENDED
UPDINV      IMS1         0     ANOTHER CMD IN PROGRESS
```

The QRY TRAN command can be used to display the work status of a transaction. It will show those conditions which would prevent a transaction from being deleted.



Information On Demand

IMS Version 10

QRY Resource – NOTINIT Status

- For all query resource commands

```
QRY rsc-type NAME(names) SHOW(STATUS)
QRY rsc-type NAME(names) STATUS(NOTINIT) SHOW(STATUS)
```

  - When status is NOTINIT
    - Reason for NOTINIT is provided
      - NOTINIT-rc-text
    - Examples

```
QRY TRAN NAME(RLSW0012) SHOW(STATUS)
```

```
Trancode MbrName CC LclStat
RLSW0012 IMS1    0  NOTINIT-01-NOPGM
```

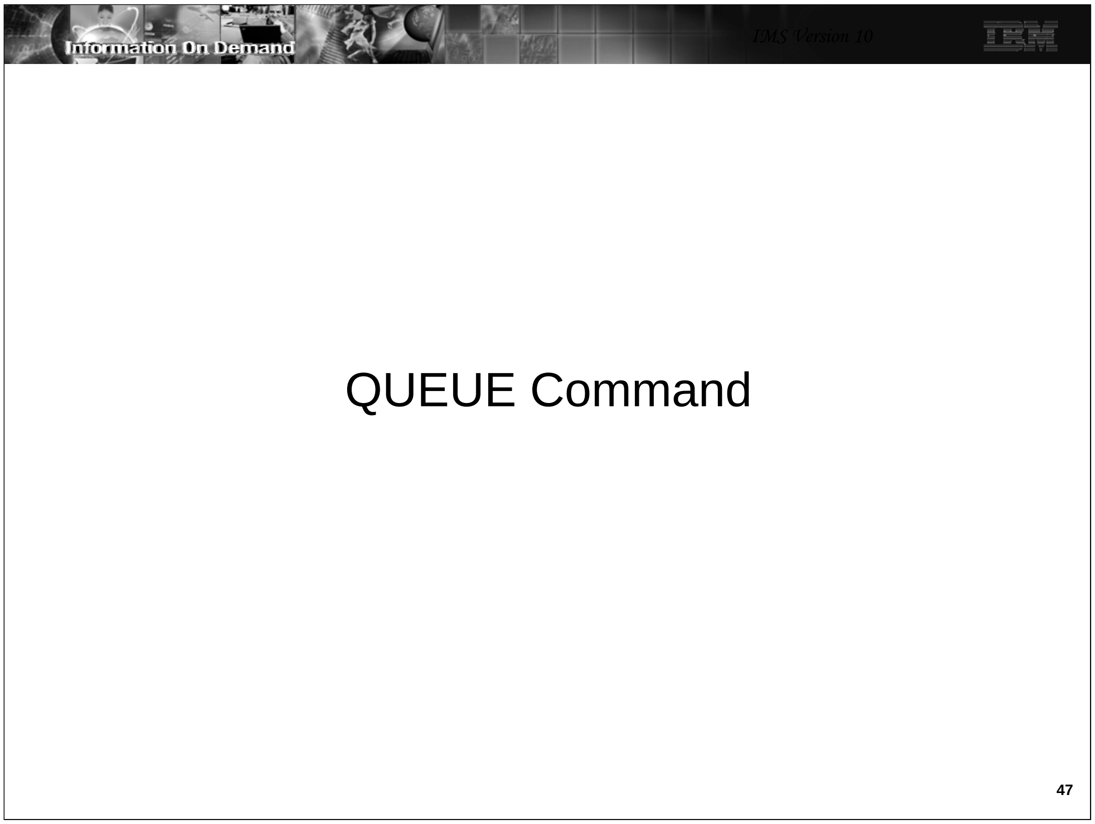
```
QRY DB STATUS(NOTINIT) SHOW(STATUS)
```

```
DBName   MbrName CC LclStat
RZSM098P IMS1    0  NOTINIT-01-NODMB
```

46

When a resource has a status of NOTINIT, IMS also keeps the reason it was not initialized. The QUERY command may be used to show this. The response includes NOTINIT followed by the reason code and a description of what that reason code means. For example, a program just created or loaded from MODBLKS or an RDDS may have a status on NOTINIT because there is no PSB in ACBLIB. The Command Reference Vol 2 manual identifies all of the reasons for a resource having a NOTINIT status.

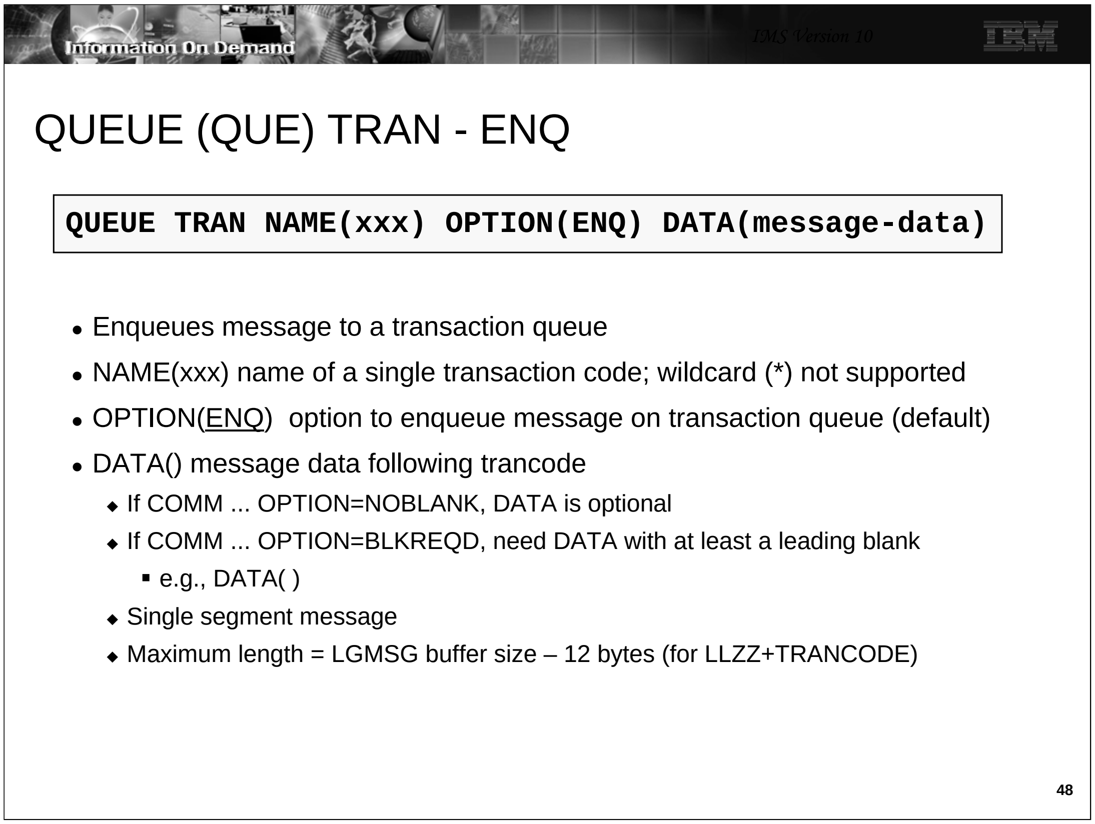
The second form of the command shown above lets you query for resources that have any NOTINIT status and the output will tell you the reason.



QUEUE Command

The QUEUE command in new in IMS V10.

Security for the QUQUE command is provided by OM with an exit, RACF, or both. The commands must be placed in the OPERCMDS resource class with a prefix of the CSL plex name. The short form of the command must be registered and may be followed by either TRAN or LTERM. For example, if IMSPLEX=PLEX1, then to provide security for the QUEUE TRAN command, you would RDEFINE CSLPLEX1.QUE.TRAN in the OPERCMDS class. Then PERMIT authorized users UPDATE access.



QUEUE (QUE) TRAN - ENQ

```
QUEUE TRAN NAME(xxx) OPTION(ENQ) DATA(message-data)
```

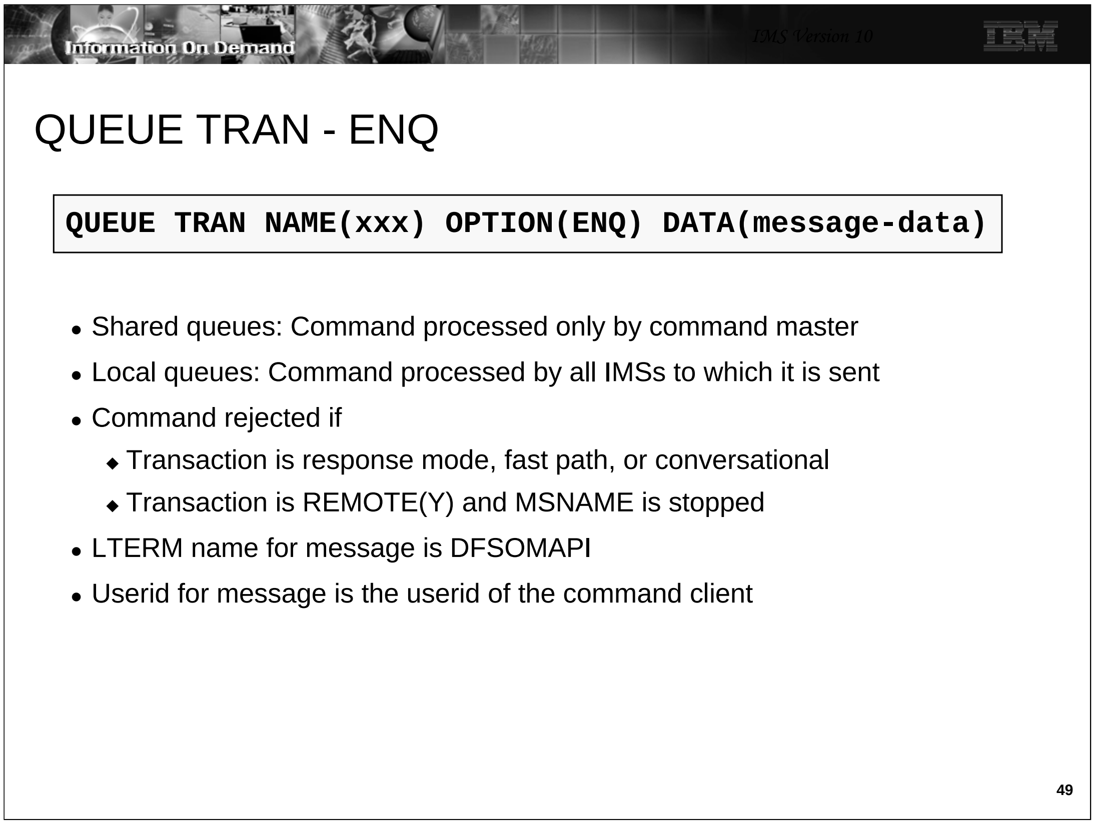
- Enqueues message to a transaction queue
- NAME(xxx) name of a single transaction code; wildcard (*) not supported
- OPTION(ENQ) option to enqueue message on transaction queue (default)
- DATA() message data following trancode
  - If COMM ... OPTION=NOBLANK, DATA is optional
  - If COMM ... OPTION=BLKREQD, need DATA with at least a leading blank
    - e.g., DATA( )
  - Single segment message
  - Maximum length = LGMSG buffer size – 12 bytes (for LLZZ+TRANCODE)

This shows the syntax when using the QUE command to enqueue a transaction to the transaction queue.

-NAME(xxx) identifies the transaction code – there can be only a single name in this command

-OPTION(ENQ) indicates that the trancode and DATA should be enqueued to the transaction queue (local or global); ENQ is the default

-DATA(xxx) is optional; if supplied, the trancode is followed by the data within the parentheses. Note that whatever is in DATA is moved character for character following the trancode. If OPTION=BLANK is specified on the COMM macro, then DATA must have a leading blank. All messages are single segment, and the maximum size of the DATA can be the size of the LGMSG buffer minus 12 bytes for LLZZ and the trancode.



QUEUE TRAN - ENQ

```
QUEUE TRAN NAME(xxx) OPTION(ENQ) DATA(message-data)
```

- Shared queues: Command processed only by command master
- Local queues: Command processed by all IMSs to which it is sent
- Command rejected if
  - Transaction is response mode, fast path, or conversational
  - Transaction is REMOTE(Y) and MSNAME is stopped
- LTERM name for message is DFSOMAPI
- Userid for message is the userid of the command client

The command is processed only by the command master in the shared queues environment. It is processed by all the IMSs to which it is sent in a local queues environment.

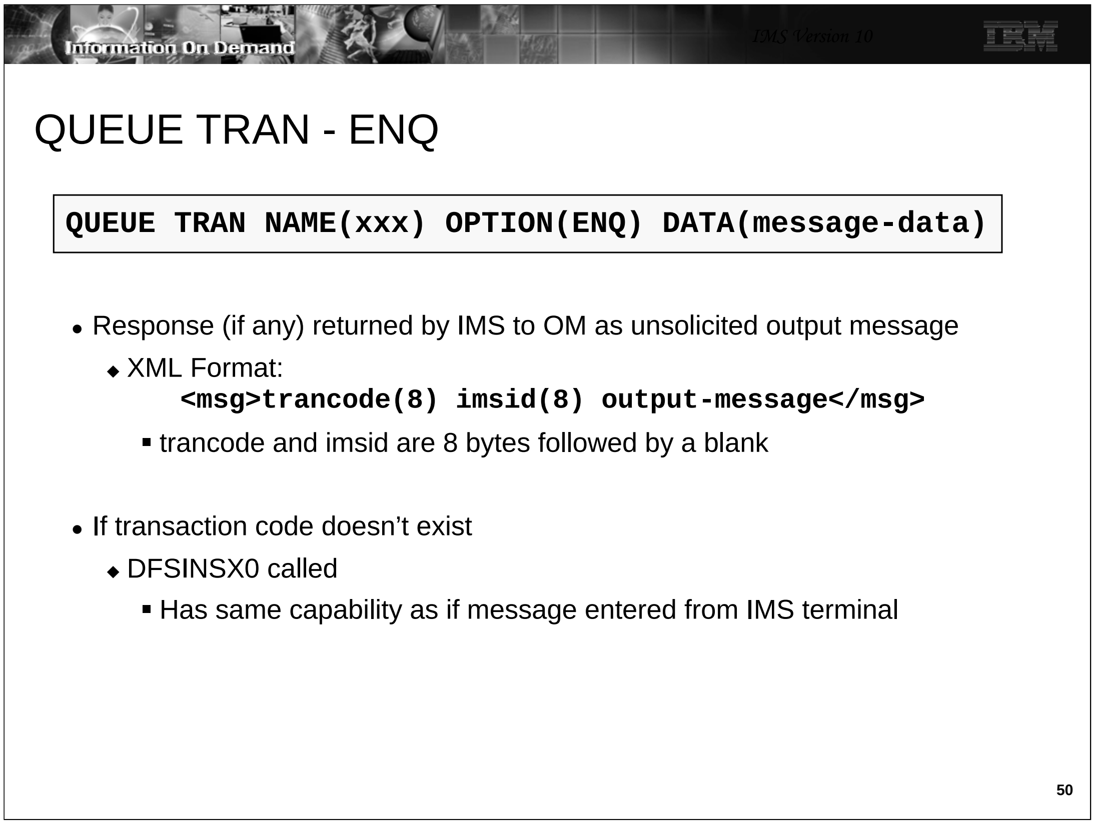
The message data does not include the LLZZ fields for the segment. IMS adds these fields.

Response mode, fast path, and conversational transactions are rejected if entered. Remote transactions are rejected if the logical link is stopped.

If there is not an RM structure and therefore, there is no global status, the command is rejected if the local status of the transaction on the command master is STOPPED.

If there is an RM structure and if the global status in a shared queues environment is STOQ, the command will be rejected even if the local status is not stopped. If there is an RM structure and if the global status in a shared queues environment is STAQ, the command will be accepted even if the local status is stopped. In other words, if there is an RM structure with shared queues, the global status controls whether the command is accepted or rejected.

The LTERM name used for messages enqueued with this command is DFSOMAPI. The userid associated with the message is the userid of the command client, such as TSO SPOC.



Information On Demand

IMS Version 10

QUEUE TRAN - ENQ

```
QUEUE TRAN NAME(xxx) OPTION(ENQ) DATA(message-data)
```

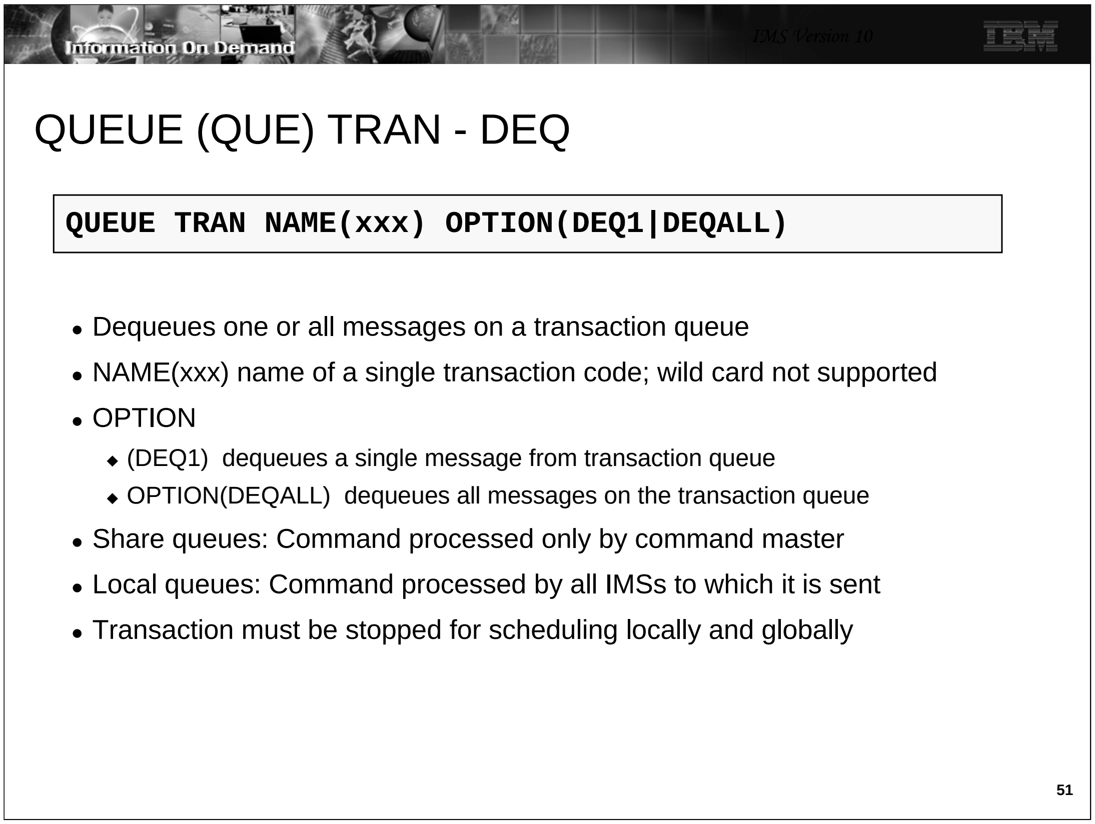
- Response (if any) returned by IMS to OM as unsolicited output message
  - XML Format:
    `<msg>trancode(8) imsid(8) output-message</msg>`
    - trancode and imsid are 8 bytes followed by a blank
- If transaction code doesn’t exist
  - DFSINSX0 called
    - Has same capability as if message entered from IMS terminal

Any response from this transaction is returned by IMS to OM in XML format as an unsolicited message. The XML tags are shown on the slide. The new OM2 Unsolicited Output Message support will place this message in the audit trail if enabled, and the SPOC (or a REXX program) can be used to display the contents of the message.

In the above XML format, 8 bytes is reserved for the trancode, followed by a blank, then 8 bytes for the IMSID, followed by a blank, then the message text.

If an IMS V8 or V9 system processes the message and inserts to the IO-PCB, the program will get an AD status code. This requires APARS PK30188 for IMS V8 and PK30189 for IMS V9.

If the transaction name is unknown, DFSINSX0 is invoked. The exit has the same capabilities as if a message from the network arrived with an unknown destination.


Information On Demand
IMS Version 10
IBM

QUEUE (QUE) TRAN - DEQ

```
QUEUE TRAN NAME(xxx) OPTION(DEQ1|DEQALL)
```

- Dequeues one or all messages on a transaction queue
- NAME(xxx) name of a single transaction code; wild card not supported
- OPTION
  - (DEQ1) dequeues a single message from transaction queue
  - OPTION(DEQALL) dequeues all messages on the transaction queue
- Share queues: Command processed only by command master
- Local queues: Command processed by all IMSs to which it is sent
- Transaction must be stopped for scheduling locally and globally

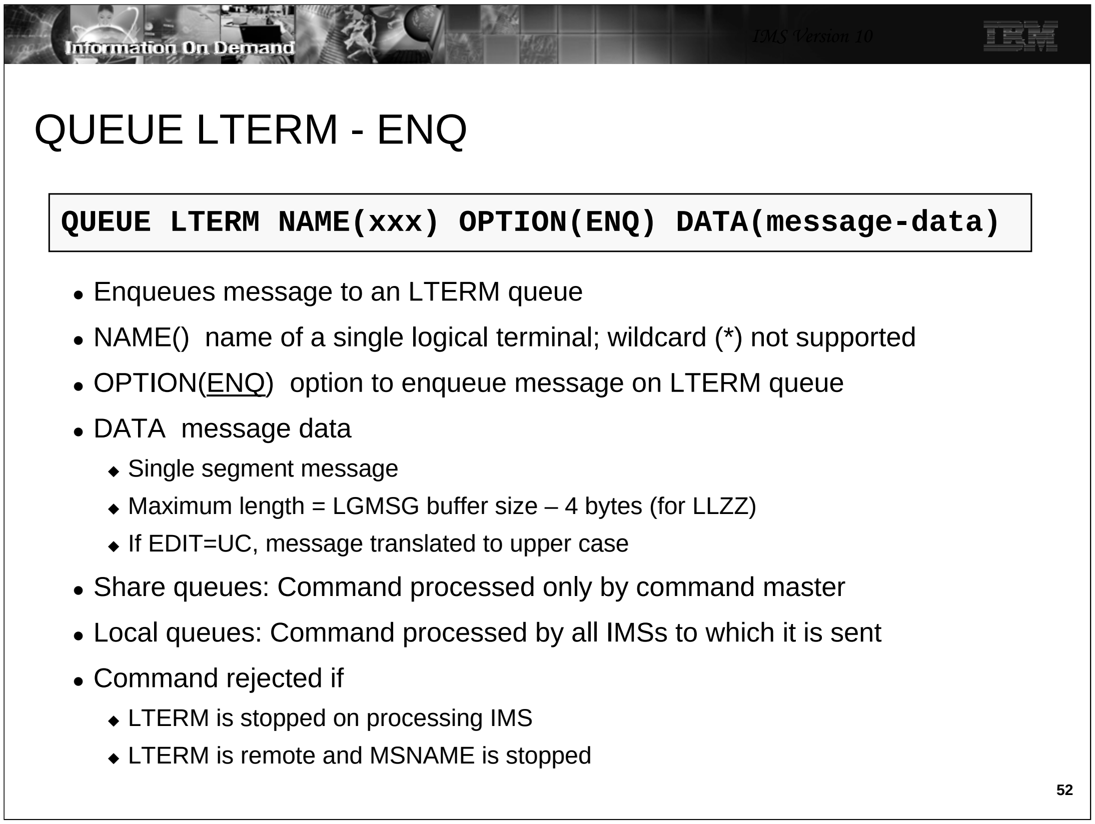
51

Transactions can be dequeued from either the local or shared message queue by specifying OPTION(DEQ|DEQALL). DEQ deletes one message. DEQALL deletes all the messages on the queue. In a shared queues environment, the transactions are deleted by the command master. In a local queue environment, each IMS to which the command is routed deletes transactions from its local queue.

The transaction must be stopped for queuing and scheduling on the IMS processing the command.

The QUEUE TRAN … OPTION(DEQ1|DEQALL) command is not supported for Fast Path exclusive transactions.

If the transaction is unknown, DFSINSX0 can create the transaction for purposes of dequeuing it. This is useful only in a SQ environment since, if it is unknown in a local queues environment, there won't be any messages on the queue.



Information On Demand

IMS Version 10

QUEUE LTERM - ENQ

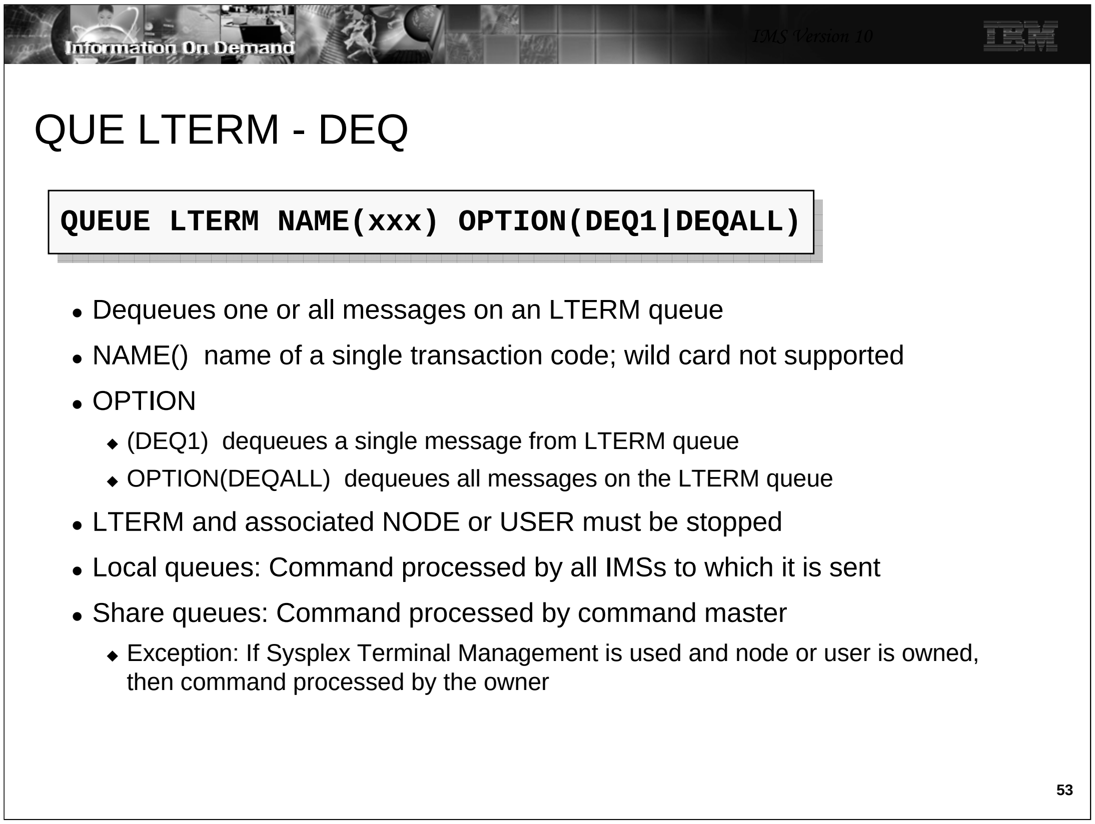
```
QUEUE LTERM NAME(xxx) OPTION(ENQ) DATA(message-data)
```

- Enqueues message to an LTERM queue
- NAME() name of a single logical terminal; wildcard (*) not supported
- OPTION(ENQ) option to enqueue message on LTERM queue
- DATA message data
  - Single segment message
  - Maximum length = LGMSG buffer size – 4 bytes (for LLZZ)
  - If EDIT=UC, message translated to upper case
- Share queues: Command processed only by command master
- Local queues: Command processed by all IMSs to which it is sent
- Command rejected if
  - LTERM is stopped on processing IMS
  - LTERM is remote and MSNAME is stopped

52

This chart shows the syntax of the QUEUE LTERM command. There can be only one name specified in the command. This is similar to the QUEUE TRAN command, except that the data can be 8 bytes longer since there is no trancode. Messages must be single segment.

In a SQ environment, only the command master queues the message. In a local queues environment, each IMS queues the message. The command is rejected by any IMS where the LTERM is stopped or if it is a remote LTERM and the MSNAME is stopped.



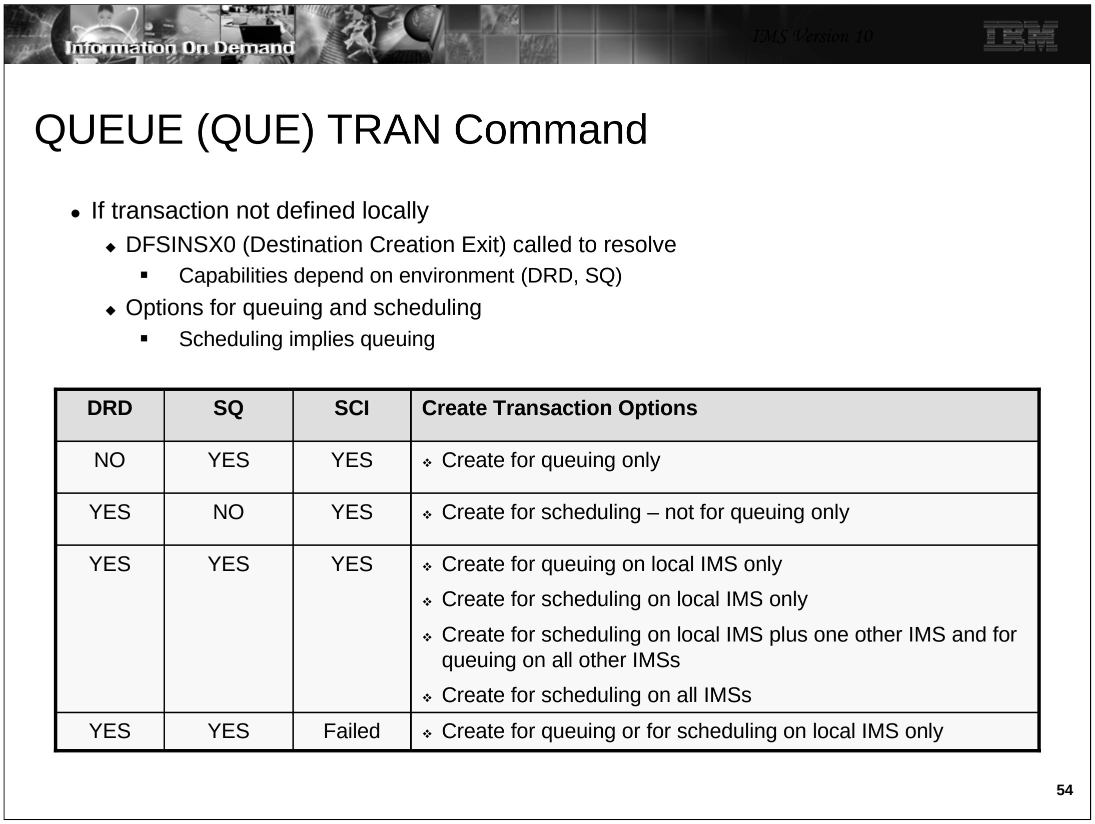
QUE LTERM - DEQ

```
QUEUE LTERM NAME(xxx) OPTION(DEQ1|DEQALL)
```

- Dequeues one or all messages on an LTERM queue
- NAME() name of a single transaction code; wild card not supported
- OPTION
  - (DEQ1) dequeues a single message from LTERM queue
  - OPTION(DEQALL) dequeues all messages on the LTERM queue
- LTERM and associated NODE or USER must be stopped
- Local queues: Command processed by all IMSs to which it is sent
- Share queues: Command processed by command master
  - Exception: If Sysplex Terminal Management is used and node or user is owned, then command processed by the owner

Messages can be dequeued from the local or shared LTERM queues. For this command to be accepted, the LTERM and associated NODE or USER must be stopped on the IMS processing the command.

This command is a little different when Sysplex Terminal Management is active (which means that SQ is also active). If STM is not enabled, the command is processed only by the command master. If STM is enabled, the command is processed by either the NODE or USER owner or by the command master if the node or user is not owned.


Information On Demand
IMS Version 10

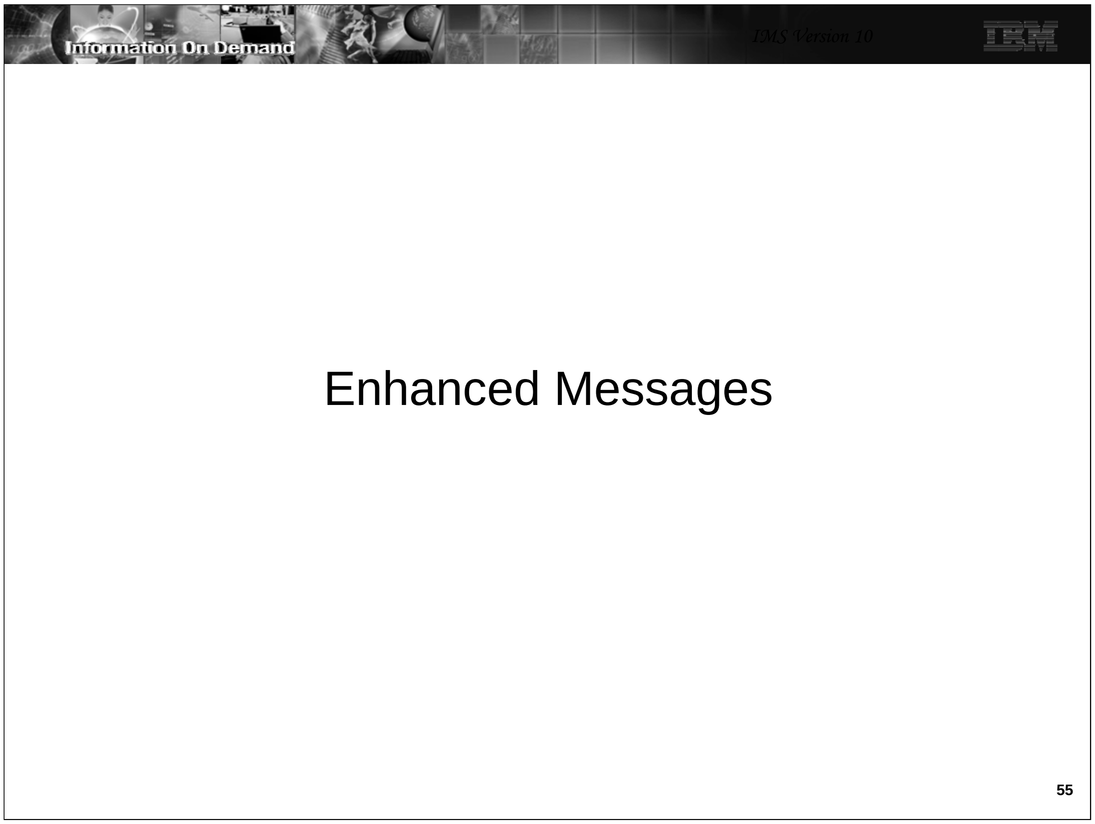
QUEUE (QUE) TRAN Command

- If transaction not defined locally
  - DFSINSX0 (Destination Creation Exit) called to resolve
    - Capabilities depend on environment (DRD, SQ)
  - Options for queuing and scheduling
    - Scheduling implies queuing

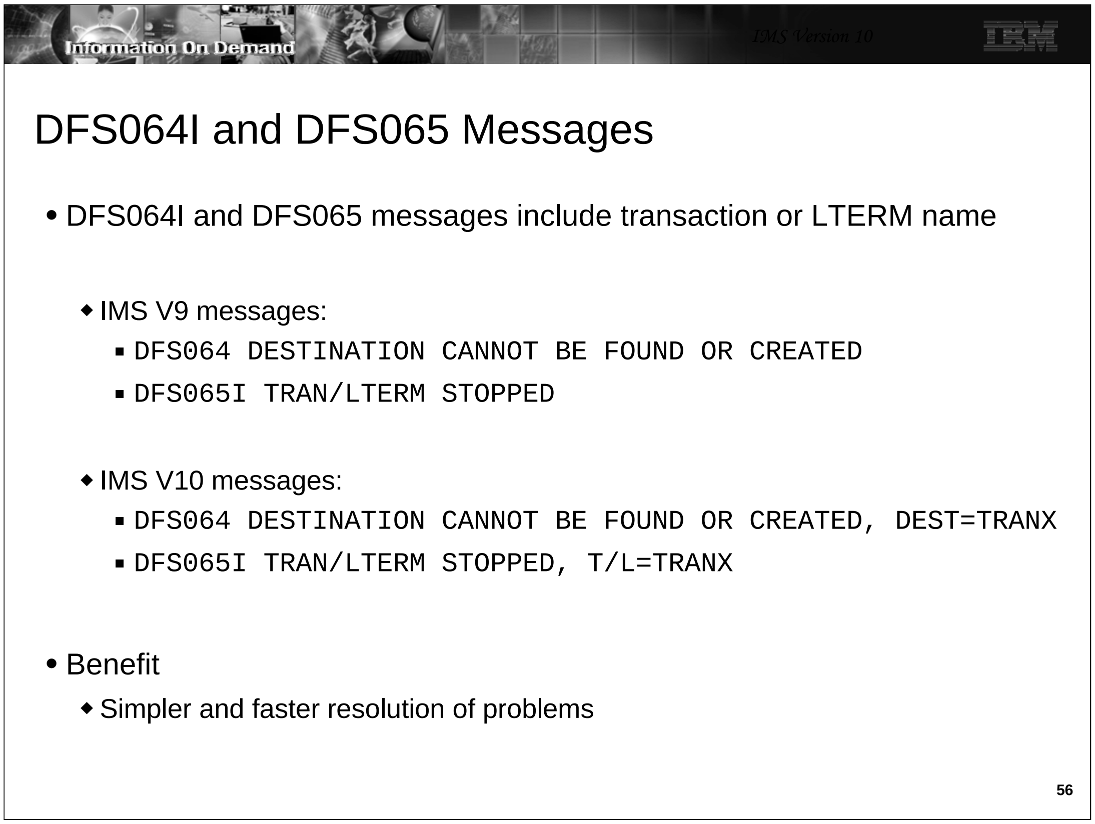
DRD	SQ	SCI	Create Transaction Options
NO	YES	YES	❖ Create for queuing only
YES	NO	YES	❖ Create for scheduling – not for queuing only
YES	YES	YES	❖ Create for queuing on local IMS only ❖ Create for scheduling on local IMS only ❖ Create for scheduling on local IMS plus one other IMS and for queuing on all other IMSs ❖ Create for scheduling on all IMSs
YES	YES	Failed	❖ Create for queuing or for scheduling on local IMS only

54

If QUE TRAN is entered and the transaction is not defined to IMS, then IMS will invoke the Destination Determination Exit (DFSINSX0) to make a decision about how to process the command. The options are the same as for a message with an unknown destination arriving from the network. The table shows the options depending on the environment – DRD, SQ, and SCI. SCI should always be there, since CSL is required for Type-2 commands, but the last row of the table identifies the options if, for some reason, SCI has failed.



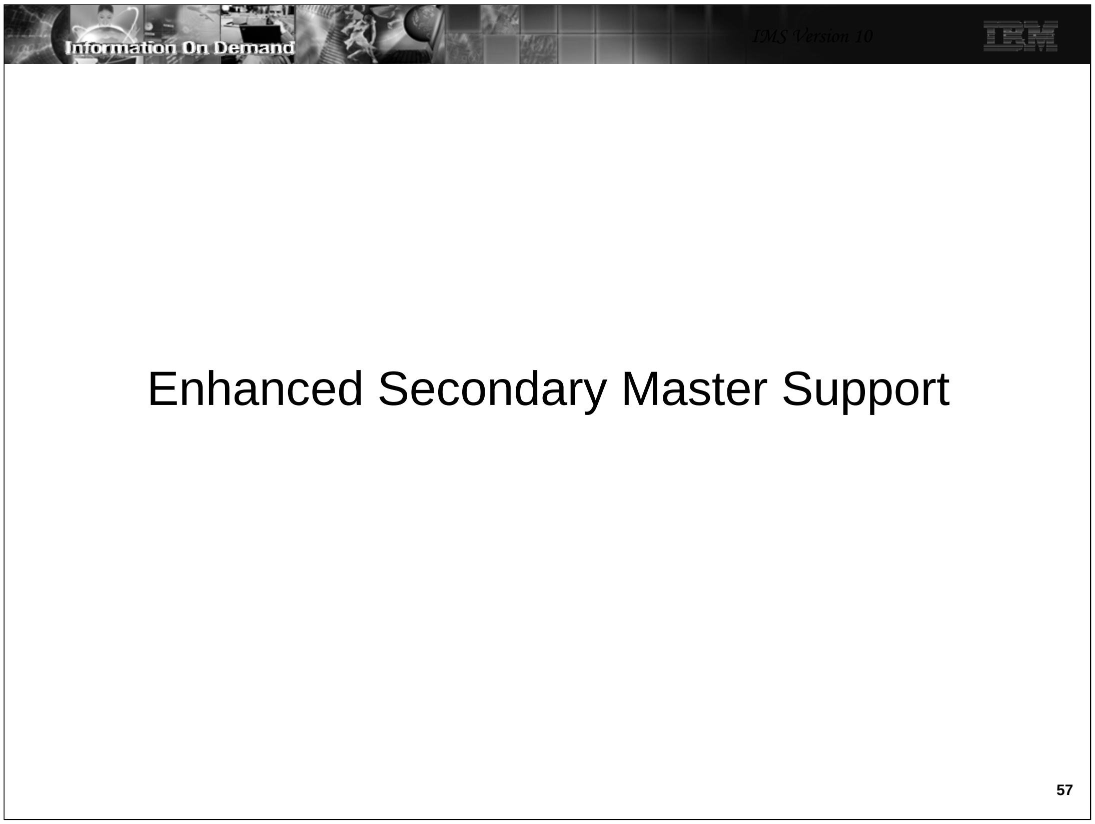
Enhanced Messages



DFS064I and DFS065 Messages

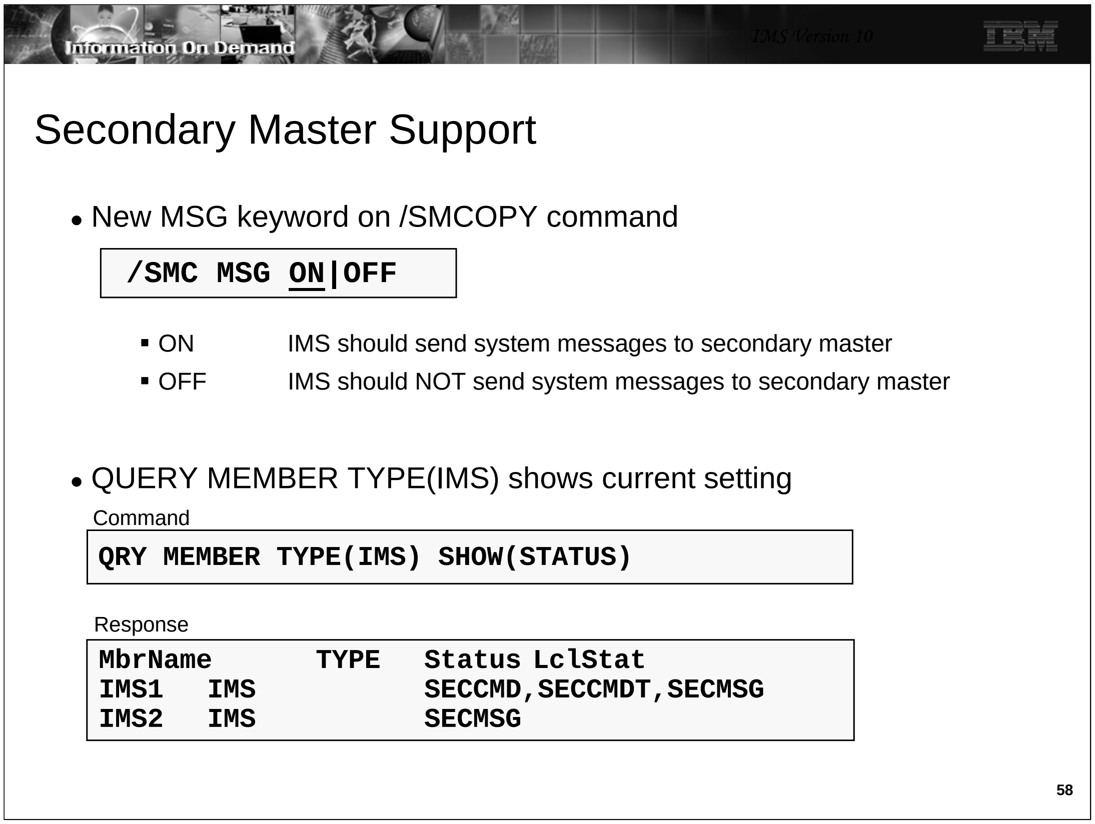
- DFS064I and DFS065 messages include transaction or LTERM name
  - IMS V9 messages:
    - `DFS064 DESTINATION CANNOT BE FOUND OR CREATED`
    - `DFS065I TRAN/LTERM STOPPED`
  - IMS V10 messages:
    - `DFS064 DESTINATION CANNOT BE FOUND OR CREATED, DEST=TRANX`
    - `DFS065I TRAN/LTERM STOPPED, T/L=TRANX`
- Benefit
  - Simpler and faster resolution of problems

The text in the DFS064 and DFS065I messages has been enhanced to include the name of the transaction or LTERM for which the message is issued. This is especially useful in situations where the user of the application is not aware of the transaction code being used since it is hidden either in the program producing the input message or in the MFS definitions being used.



Enhanced Secondary Master Support

Although the DRD CREATE and DELETE process does not support MSC resources, there is some enhanced support for MSC in V10.


Information On Demand

Secondary Master Support

- New MSG keyword on /SMCOPY command

```
/SMC MSG ON|OFF
```

- ON IMS should send system messages to secondary master
- OFF IMS should NOT send system messages to secondary master

- QUERY MEMBER TYPE(IMS) shows current setting

Command

```
QRY MEMBER TYPE(IMS) SHOW(STATUS)
```

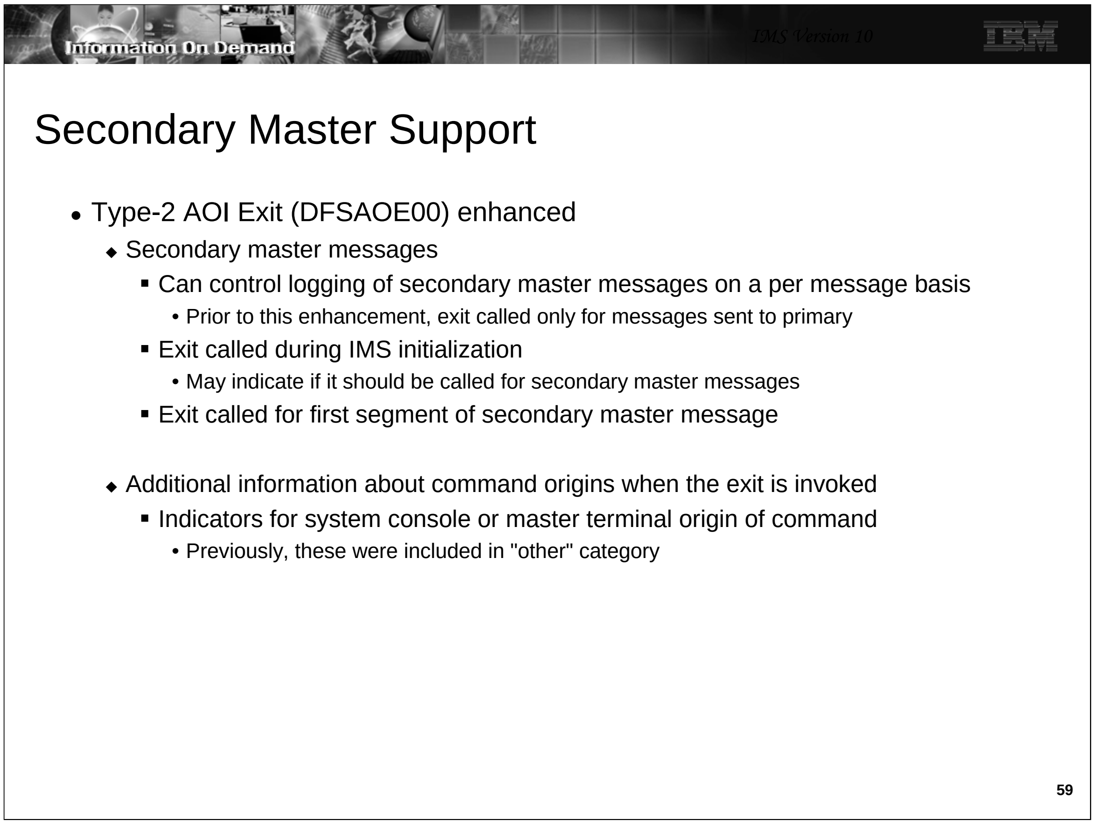
Response

```
MbrName       TYPE   Status LclStat
IMS1   IMS           SECCMD,SECCMDT,SECMSG
IMS2   IMS           SECMSG
```

You can turn off the logging of system messages to the secondary master in V10 by entering the command: /SMC MSG OFF.

In previous releases the /SMCOPY command was used to control the copying of commands and command responses to the secondary master. It did not affect the copying of system messages to the secondary master. /SMC MASTER ON|OFF controlled the sending of commands from the master terminal and their responses the secondary master. /SMC TERMINAL ON|OFF controlled the sending of commands from non-master terminals and their responses the secondary master. IMS V10 adds the control of whether or not systems messages are sent to the secondary master. In previous releases all system messages were sent to the secondary master.

A status of SECMSG indicates this status in response to a QRY MEMBER TYPE(IMS) SHOW(STATUS) command indicates that system messages are sent to the secondary master. SECCMD indicates that commands from the master are sent to the secondary master. SECCMDT indicates that commands from the other terminals are sent to the secondary master.



Information On Demand
IMS Version 10
IBM

Secondary Master Support

- Type-2 AOI Exit (DFSAOE00) enhanced
  - Secondary master messages
    - Can control logging of secondary master messages on a per message basis
      - Prior to this enhancement, exit called only for messages sent to primary
    - Exit called during IMS initialization
      - May indicate if it should be called for secondary master messages
    - Exit called for first segment of secondary master message
  - Additional information about command origins when the exit is invoked
    - Indicators for system console or master terminal origin of command
      - Previously, these were included in "other" category

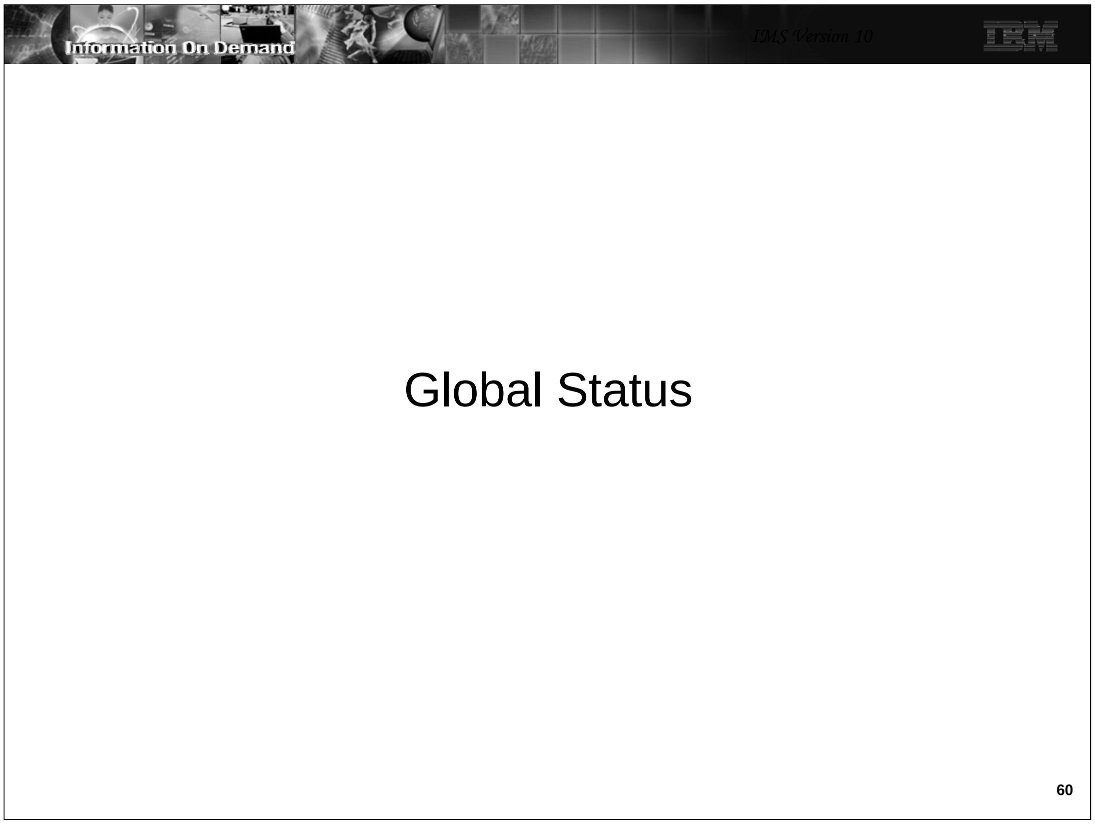
59

The Type-2 AOI Exit (DFSAOE00) has been enhanced in two ways.

First, it can now control the logging of system messages to the secondary master on a message-by-message basis. The exit decides during IMS initialization if it wants to be called for secondary master messages. In prior versions, it was called only for messages to the primary master) If it is called for secondary master messages, it can reply to IMS to process the message normally or cancel the message to the secondary master. If the message is also going to the primary, it is called a second time.

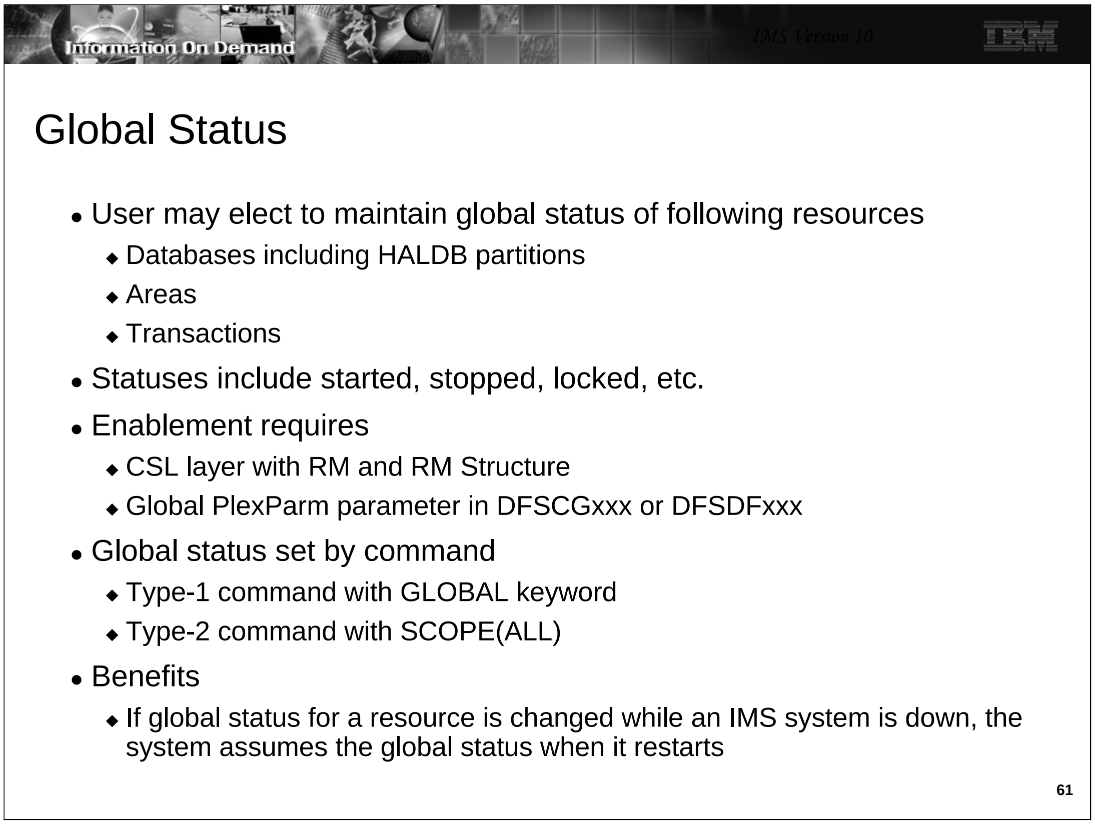
Second, when the exit routine is invoked for a command, the function-specific parameter list also includes whether the command originated with the system console or the MTO. In earlier versions of IMS these origins were included in the "other" category.

A more complete description of the use of this exit for secondary master logging is described in the Exit Routine Reference manual.



Global Status

This section discusses the concept of “global status” which is enabled by defining a Global PLEXPARM parameter.


Information On Demand
IMS Version 10
IBM

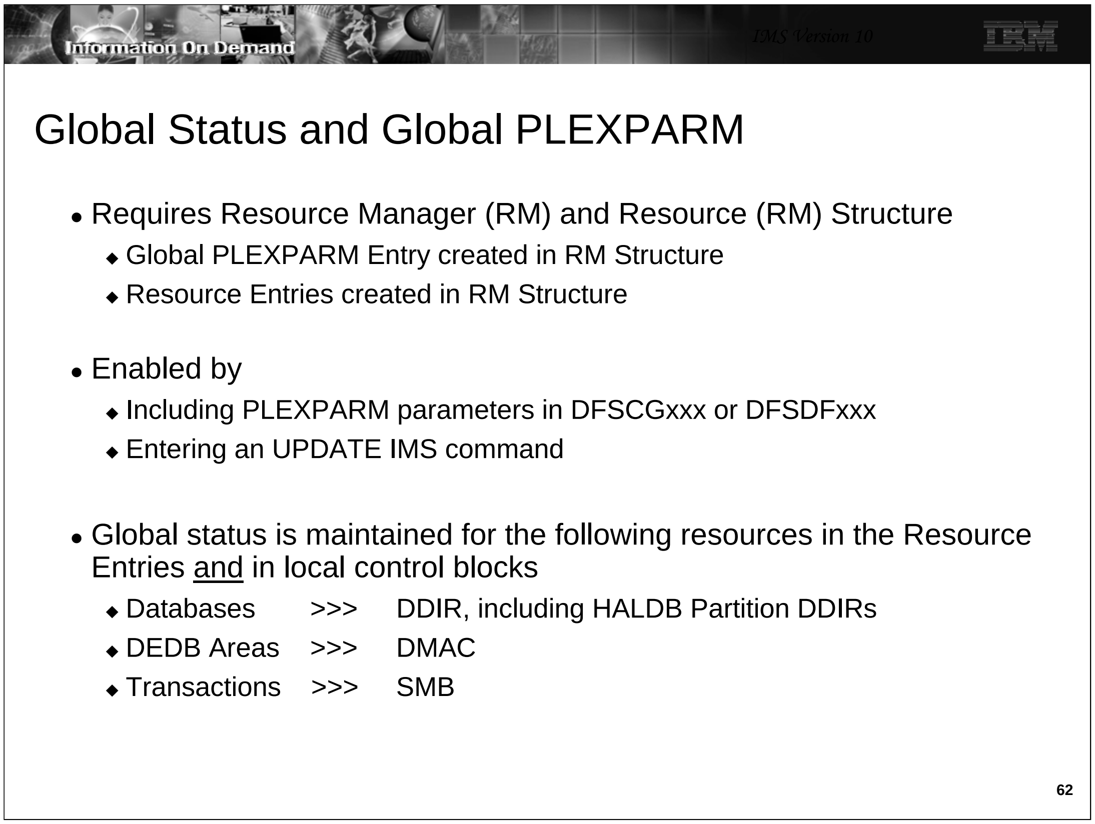
Global Status

- User may elect to maintain global status of following resources
  - Databases including HALDB partitions
  - Areas
  - Transactions
- Statuses include started, stopped, locked, etc.
- Enablement requires
  - CSL layer with RM and RM Structure
  - Global PlexParm parameter in DFSCGxxx or DFSDFxxx
- Global status set by command
  - Type-1 command with GLOBAL keyword
  - Type-2 command with SCOPE(ALL)
- Benefits
  - If global status for a resource is changed while an IMS system is down, the system assumes the global status when it restarts

61

The user can elect to maintain global status for databases, areas, and transactions in a Resource Structure. Global status is enabled initially by coding the GSTSxxx parameters in the DFSCGxxx or DFSDFxxx PROCLIB member, and may be updated (or added) later by the UPD IMS command. Global status for a resource is set with a global command. Global commands are type-1 commands with the GLOBAL keyword or type-2 commands with SCOPE(ALL) specified.

The benefit of global status is that it allows users to more easily manage sysplex environments where some IMS systems are not always up. The user may change the status of a resource, such as starting it or stopping it, while some IMS systems are not up. When these systems are started, the resources in them assume the statuses that were changed while they were not running.


Information On Demand
IMS Version 10
IBM

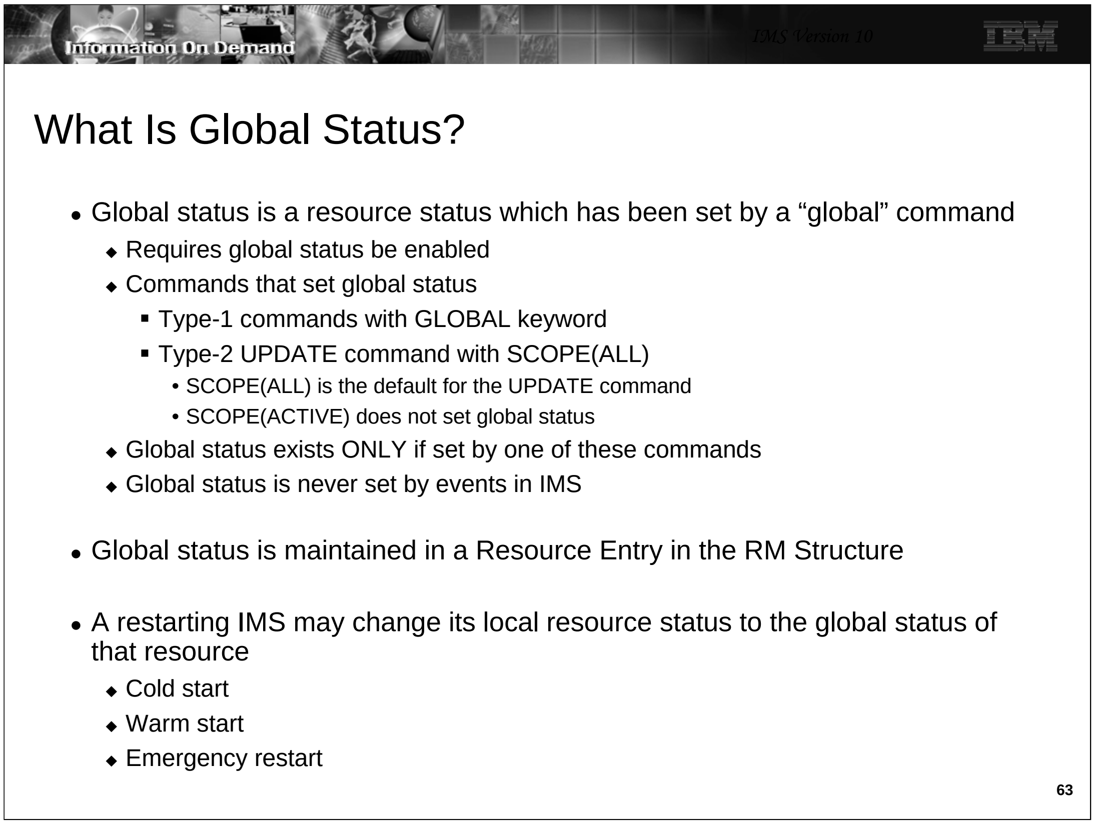
Global Status and Global PLEXPARM

- Requires Resource Manager (RM) and Resource (RM) Structure
  - Global PLEXPARM Entry created in RM Structure
  - Resource Entries created in RM Structure
- Enabled by
  - Including PLEXPARM parameters in DFSCGxxx or DFSDFxxx
  - Entering an UPDATE IMS command
- Global status is maintained for the following resources in the Resource Entries <u>and</u> in local control blocks
  - Databases >>> DDIR, including HALDB Partition DDIRs
  - DEDB Areas >>> DMAC
  - Transactions >>> SMB

62

Maintaining global status on databases, DEDB areas, and transactions requires the CSL environment with a Resource Manager and Resource Structure. Resource entries are created in the structure and hold the global status. Enabling parameters are coded in DFSCGxxx, or the CSL section of DFSDFxxx. They may also be enabled after IMS is up and running using the UPDATE command.

Resources for which global status is kept are databases (all kinds except MSDBs), HALDB partitions, DEDB areas, and transactions. The global status is kept in both the resource entry and the local control block.


Information On Demand
IMS Version 10
IBM

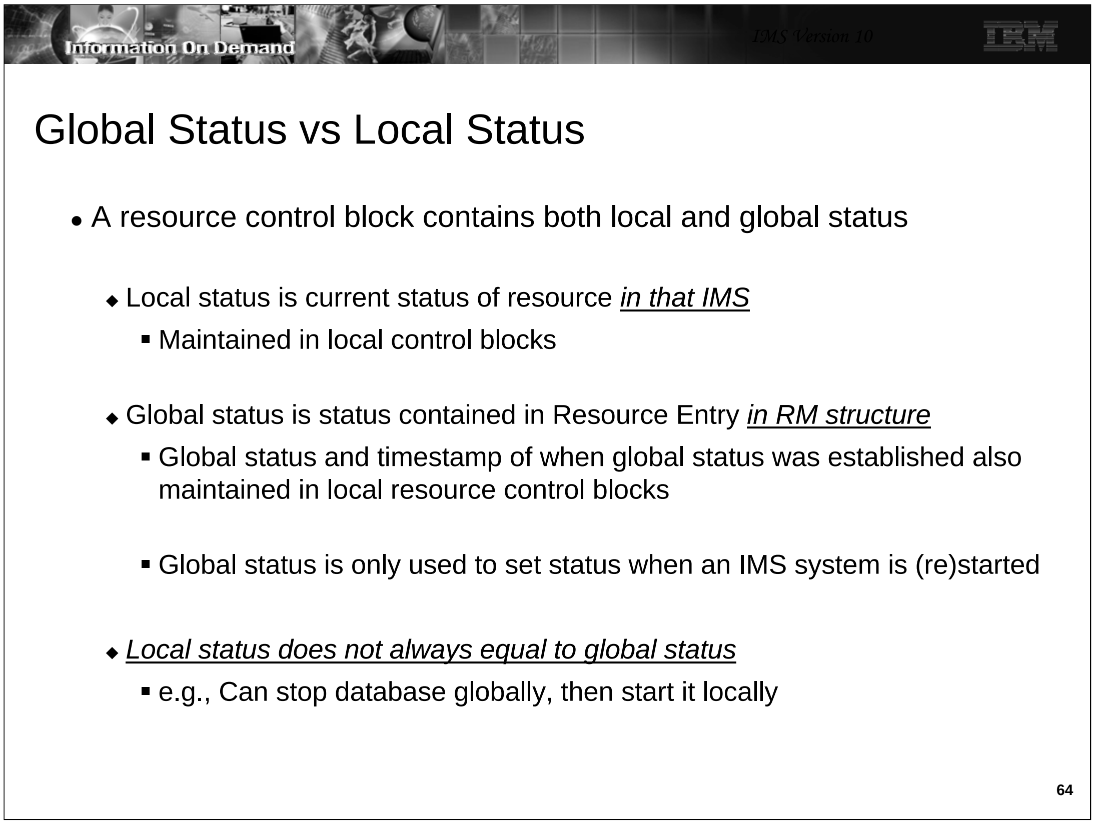
What Is Global Status?

- Global status is a resource status which has been set by a “global” command
  - Requires global status be enabled
  - Commands that set global status
    - Type-1 commands with GLOBAL keyword
    - Type-2 UPDATE command with SCOPE(ALL)
      - SCOPE(ALL) is the default for the UPDATE command
      - SCOPE(ACTIVE) does not set global status
  - Global status exists ONLY if set by one of these commands
  - Global status is never set by events in IMS
- Global status is maintained in a Resource Entry in the RM Structure
- A restarting IMS may change its local resource status to the global status of that resource
  - Cold start
  - Warm start
  - Emergency restart

Global status is status which has been set by a “global command” or more accurately, a command with global scope.

Both type-1 and type-2 command may be global in scope. For type-1 commands, this means including the keyword GLOBAL on the command - for example /DBR DATABASE XXX GLOBAL. For type-2 commands, this means including SCOPE(ALL) on the command - for example, UPD DB NAME(XXX) STOP(ACCESS) SCOPE(ALL). Global status is only set by one of these commands. It is never set by events in IMS.

When IMS is restarted, in may change its local status for a resource to the global status found in the structure. This is always true for cold starts, and sometimes true for warm or emergency restart. The conditions under which global status will override local status will be discussed a little later.

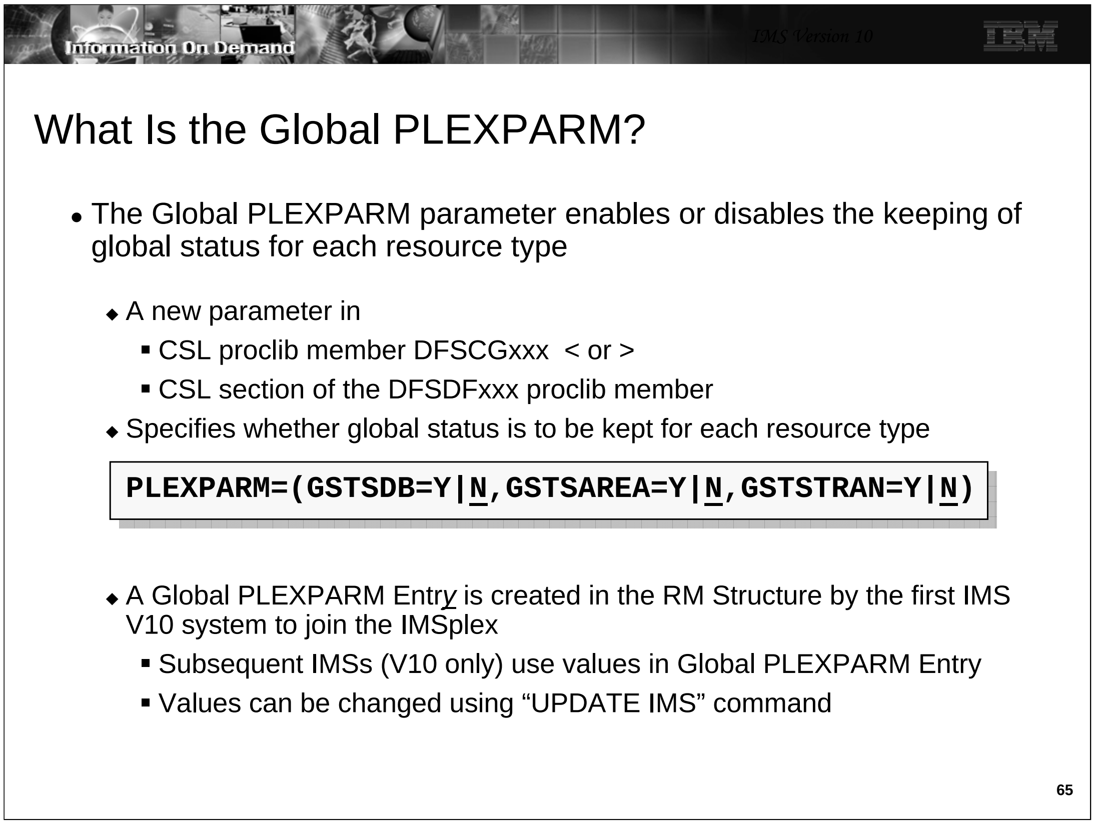


Global Status vs Local Status

- A resource control block contains both local and global status
  - Local status is current status of resource *in that IMS*
    - Maintained in local control blocks
  - Global status is status contained in Resource Entry *in RM structure*
    - Global status and timestamp of when global status was established also maintained in local resource control blocks
    - Global status is only used to set status when an IMS system is (re)started
  - *Local status does not always equal to global status*
    - e.g., Can stop database globally, then start it locally

Resource controls blocks (DDIRs, DMACs, and SMBs) contain BOTH the local status and the last known global status. Global status is also contained in the resource entry of the structure. Both the control block and the structure also have the timestamp of when that status was set.

Local status does not always equal global status. For example, a database can be stopped globally, but then later started locally on one IMS. We will talk more later about this concept of local vs global status.



What Is the Global PLEXPARM?

- The Global PLEXPARM parameter enables or disables the keeping of global status for each resource type
  - A new parameter in
    - CSL proclib member DFSCGxxx < or >
    - CSL section of the DFSDFxxx proclib member
  - Specifies whether global status is to be kept for each resource type

```
PLEXPARM=(GSTSDB=Y|N,GSTSAREA=Y|N,GSTSTRAN=Y|N)
```

  - A Global PLEXPARM Entry is created in the RM Structure by the first IMS V10 system to join the IMSplex
    - Subsequent IMSs (V10 only) use values in Global PLEXPARM Entry
    - Values can be changed using “UPDATE IMS” command

Whether or not global status is maintained is determined by the “global plexparm” parameters in CSL (either DFSCGxxx or the CSL section of DFSDFxxx). The PLEXPARM parameter has values for databases, DEDB areas, and transactions. In each case, the default is N (no), meaning that global status is not to be maintained for that resource type.

When any IMS V10 in the IMSplex joins the IMSplex, it looks for this parameter in the RM structure. If it does not find it, it creates a global plexparm entry with values for GSTSDB, GSTSAREA, and GSTSTRAN.

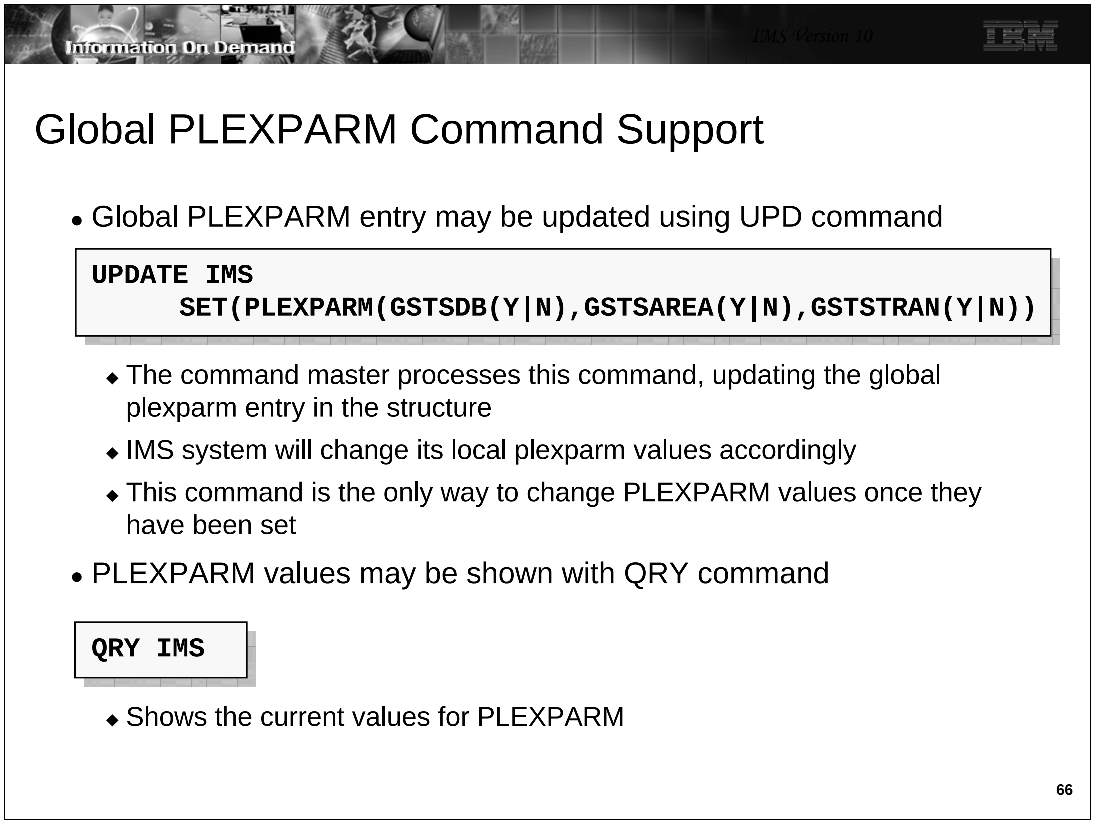
The user can change these parameters later using the UPDATE IMS command. The UPDATE IMS command is new in V10.

When an IMS system starts, it uses the values from the PLEXPARM entry in the RM structure if they exist. If they do not exist, it sets them from its parameters (those specified in DFSCGxxx or DFSDFxxx). If the values in the structure entry and those from the IMS system differ, message DFS3425I is issued. The message is:

DFS3425I PLEXPARM INIT VALUE rrrrrrrr=xxxx OVERRIDDEN WITH GLOBAL PLEXPARM VALUE rrrrrrrr=y

where rrrrrrr is GSTSDB, GSTSAREA, or GSTSTRAN, x is Y, N, or NULL and y is Y or N.

DBCTL systems do not have transaction resources. DCCTL systems do not have database or area resources. The PLEXPARM value for the resource type they do not support is NULL within these systems. Nevertheless, they can set values for all three parameters in their PROCLIB member or with the UPDATE IMS command.


Information On Demand
IMS Version 10
IBM

Global PLEXPARM Command Support

- Global PLEXPARM entry may be updated using UPD command

```
UPDATE IMS
     SET(PLEXPARM(GSTSDB(Y|N),GSTSAREA(Y|N),GSTSTRAN(Y|N))
```

  - The command master processes this command, updating the global plexparm entry in the structure
  - IMS system will change its local plexparm values accordingly
  - This command is the only way to change PLEXPARM values once they have been set

- PLEXPARM values may be shown with QRY command

```
QRY IMS
```

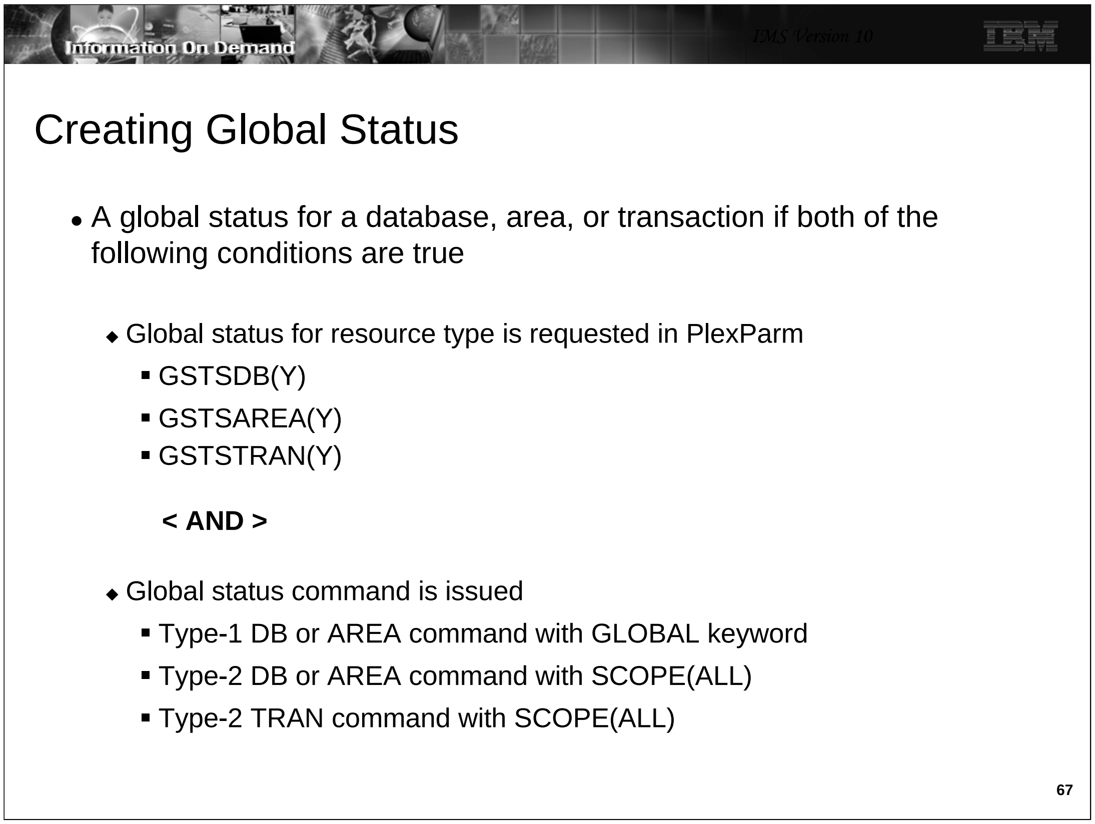
  - Shows the current values for PLEXPARM

66

The global PLEXPARM parameters set in the CSL PROCLIB member are only effective for the first IMS to join the IMSplex. Afterward, they may be changed only by using the UPDATE IMS command as shown in this slide. When this command is entered, the values of the global PLEXPARM parameters in the entry of the structure are updated by the command master. Each IMS to which the command is routed will also change its local PLEXPARM values.

Restarting an IMS with a different set of PLEXPARM values does NOT change the global values in the structure entry.

The QUERY IMS command may be used to display the current settings of the PLEXPARM parameters.


Information On Demand
IMS Version 10

Creating Global Status

- A global status for a database, area, or transaction if both of the following conditions are true
  - Global status for resource type is requested in PlexParm
    - GSTSDB(Y)
    - GSTSAREA(Y)
    - GSTSTRAN(Y)

      < AND >
  - Global status command is issued
    - Type-1 DB or AREA command with GLOBAL keyword
    - Type-2 DB or AREA command with SCOPE(ALL)
    - Type-2 TRAN command with SCOPE(ALL)

67

Database and area entries are added to the structure if:

-they are not already there (obviously)

-global status is enabled (GSTSxxx = Y)

-and global status exists

This means that IMS does not automatically add an entry to the structure just because it has a database or someone stopped or started it locally. It only adds an entry if someone enters a global command, giving the DB or area global status. Once it has global status, then it stays in the structure until it no longer has global status. This would occur only if someone changes GSTSxxx to N. If that happens, the database or area entries are deleted from the structure.

In the sense used here a "global command" is a type-1 DB or AREA command with the GLOBAL keyword specified or a type-2 command with SCOPE(ALL) specified. These commands set global status for the resource.

On the other hand, there is a significant difference between the type-1 commands with the GLOBAL keyword and type-2 commands with SCOPE(ALL). Type-1 commands with the GLOBAL keyword are sent to all of the data sharing partners via an IRLM notify. If you issue a type-1 command with the GLOBAL keyword it will be processed by all the data sharing systems. SCOPE(ALL) on a type-2 command does NOT cause the command to be processed by other systems. It only affects the global status in the RM structure and the local status to be set in the systems to which the command is routed. Type-2 commands are only processed by the systems to which they are directly sent. IRLM notifies are not used to send type-2 commands to other systems.

Examples of Setting or Changing Global Status

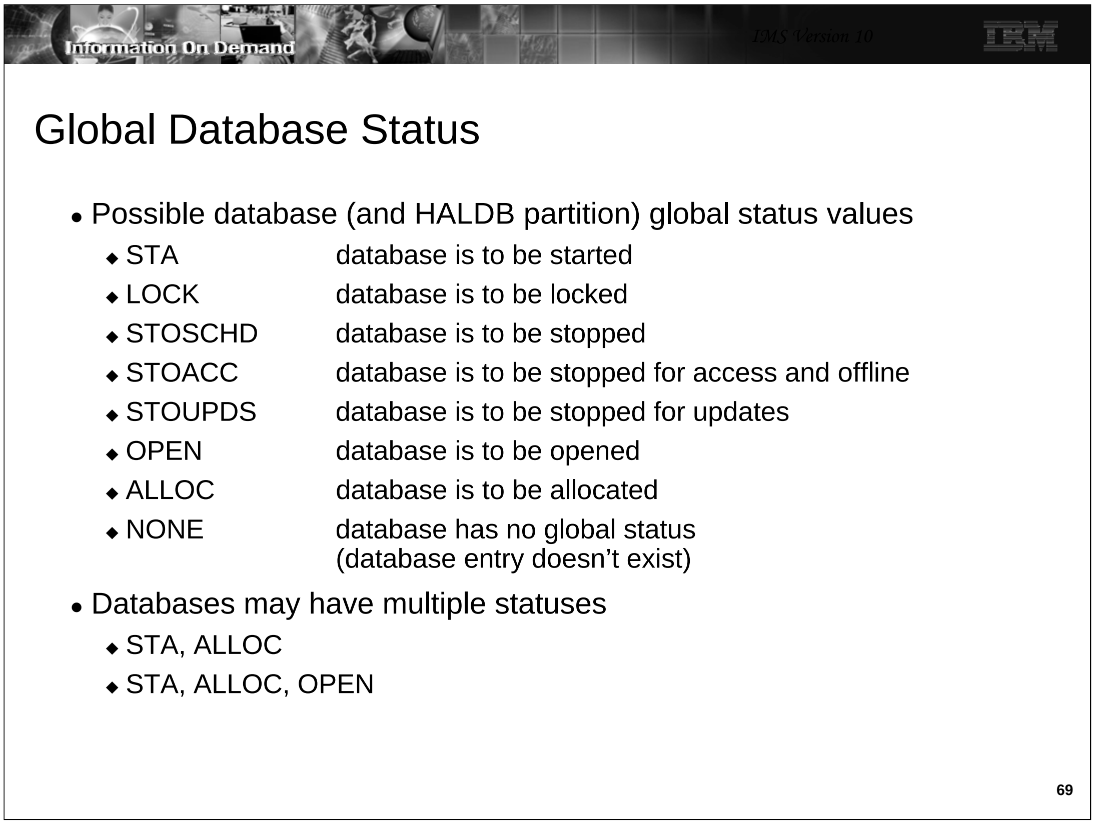
```
/DBR DB ABC GLOBAL

UPD DB NAME(ABC) STOP(ACCESS) SCOPE(ALL)

UPD DB NAME(ABC) STOP(ACCESS)

UPD TRAN NAME(XYZ) STOP(SCHD) SCOPE(ALL)

UPD TRAN NAME(XYZ) STOP(SCHD)
```

- GLOBAL changes global status for type-1 commands
- SCOPE(ALL) changes global status for type-2 commands
  - SCOPE(ALL) is the default for the UPD command

You should note that even though the first three commands on this page set the global status of database ABC to STOACC, they are not equivalent commands.

The GLOBAL parameter on type-1 /DBR, /DBD, /STA, and /STO commands for databases and areas causes the command to be sent to the data sharing partner IMS systems. This is done via IRLM notifies. Since the other systems receive the command they change the local status of the resource.

On the other hand, the SCOPE(ALL) parameter on type-2 commands does not cause the command to be sent to any other IMS systems. If you route the command to only one system, only that system is affected by the command. The local status on other systems is not changed. If you want to change the local status on all systems, you should route the command to all systems.



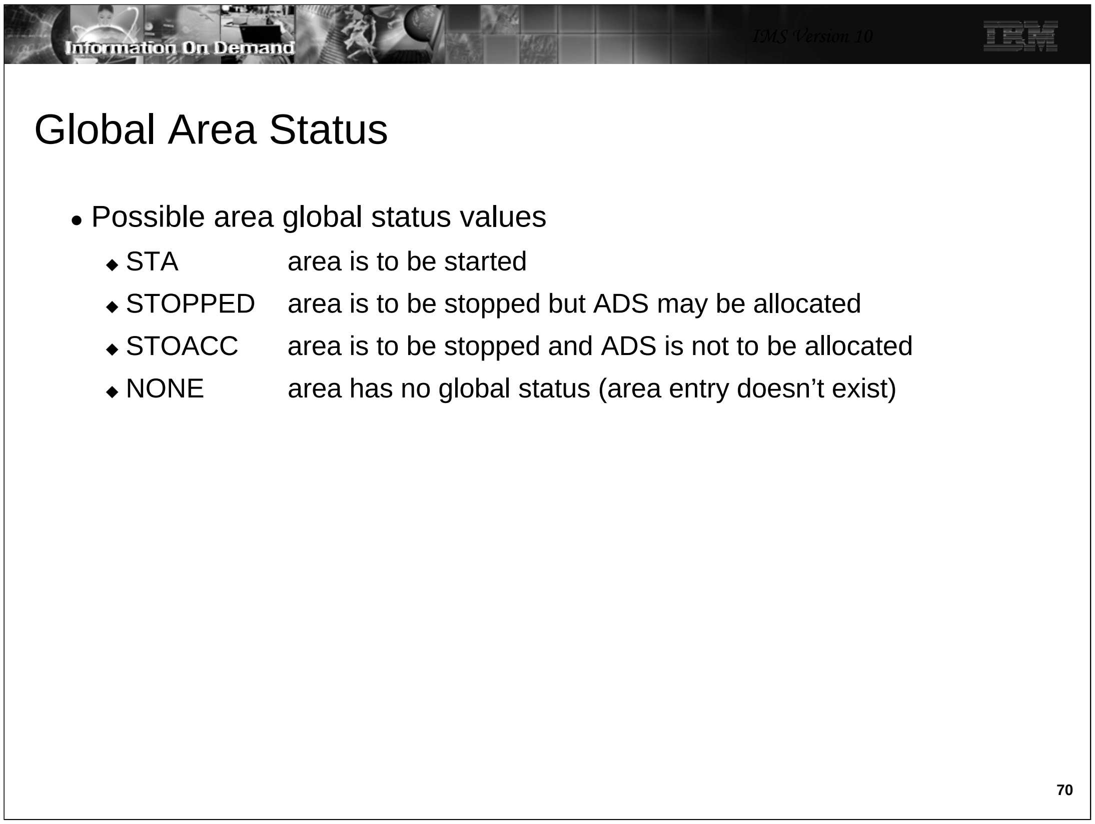
Global Database Status

- Possible database (and HALDB partition) global status values

◆ STA	database is to be started
◆ LOCK	database is to be locked
◆ STOSCHD	database is to be stopped
◆ STOACC	database is to be stopped for access and offline
◆ STOUPDS	database is to be stopped for updates
◆ OPEN	database is to be opened
◆ ALLOC	database is to be allocated
◆ NONE	database has no global status (database entry doesn't exist)

- Databases may have multiple statuses
  - STA, ALLOC
  - STA, ALLOC, OPEN

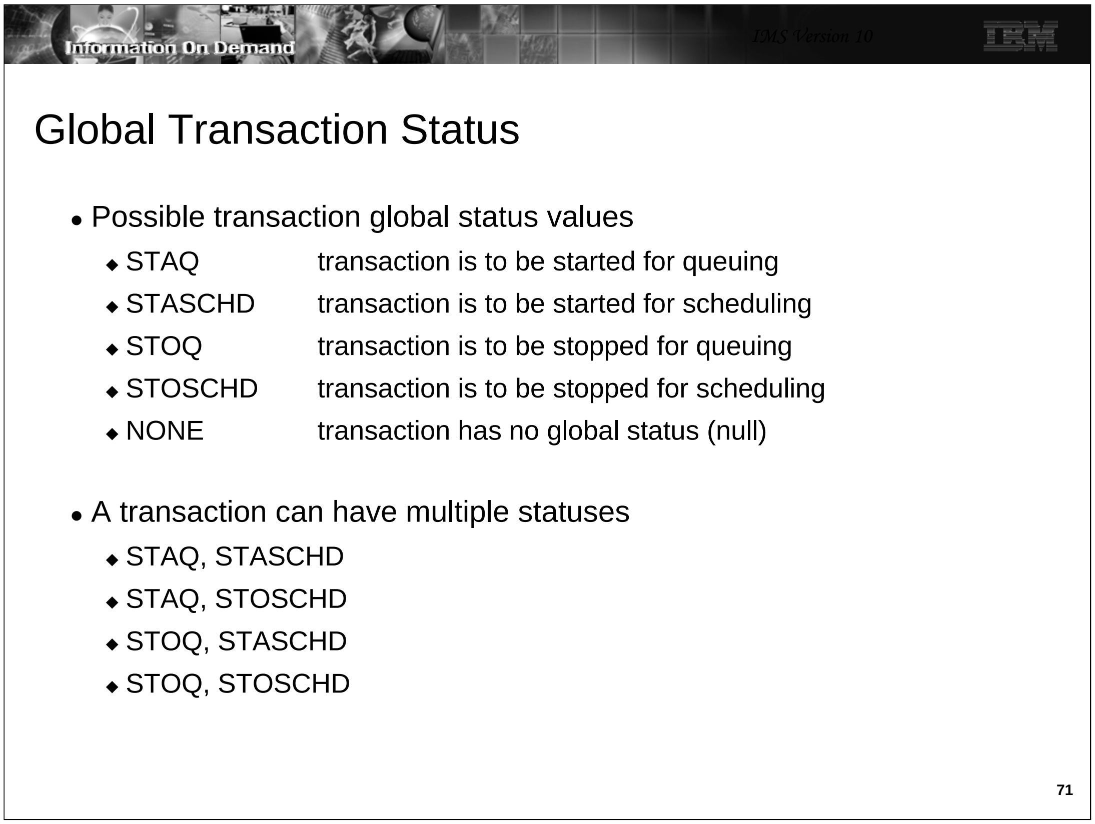
This chart shows the possible global statuses that an database or HALDB partition can have. It may also have no status (NONE) in which case there would be no database entry. It is also possible to have multiple global statuses, such as started, allocated, and open. You will note that the notation for these global statuses is sometimes different than the same local status. For example, you can have a global status of STOACC and a local status of STOPPED which mean the same thing. Another example is that you could have a global status of STOUPDS and a local status of READONLY which mean the same thing.


Information On Demand
IMS Version 10
IBM

Global Area Status

- Possible area global status values
  - STA area is to be started
  - STOPPED area is to be stopped but ADS may be allocated
  - STOACC area is to be stopped and ADS is not to be allocated
  - NONE area has no global status (area entry doesn’t exist)

This chart shows the possible global statuses for DEDB areas.



Information On Demand

IMS Version 10

Global Transaction Status

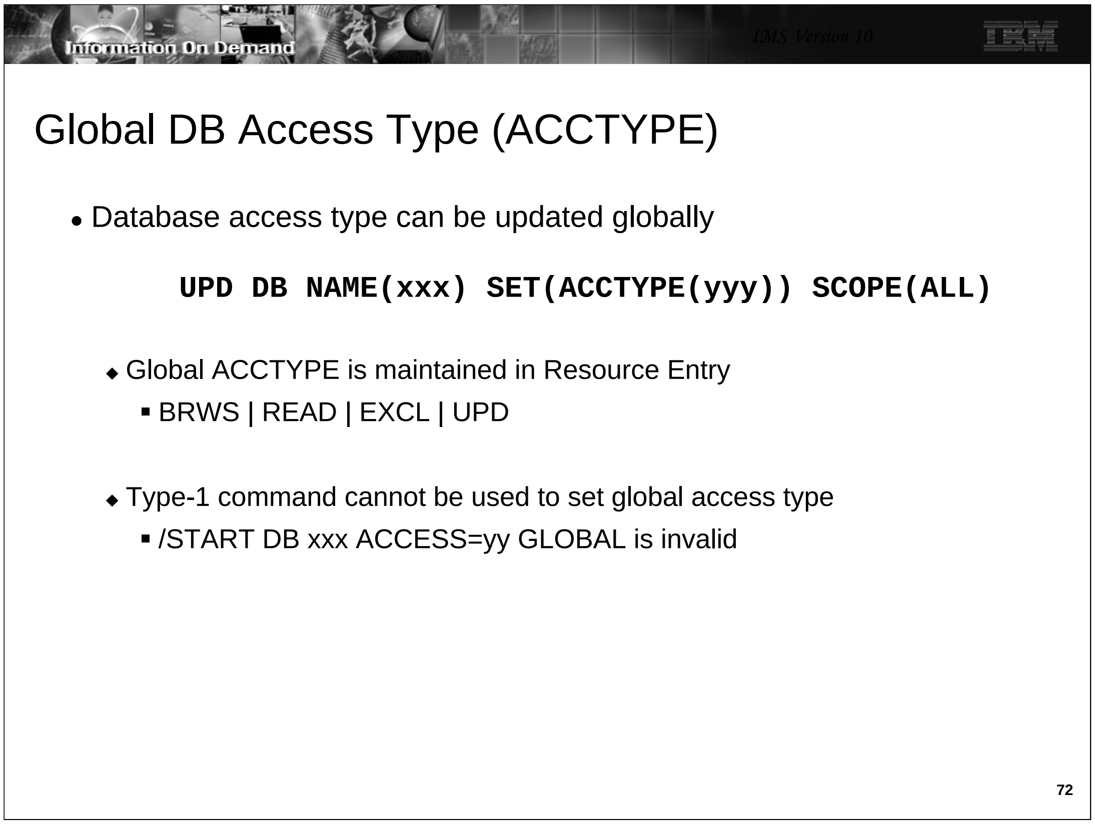
- Possible transaction global status values

STAQ	transaction is to be started for queuing
STASCHD	transaction is to be started for scheduling
STOQ	transaction is to be stopped for queuing
STOSCHD	transaction is to be stopped for scheduling
NONE	transaction has no global status (null)

- A transaction can have multiple statuses
  - STAQ, STASCHD
  - STAQ, STOSCHD
  - STOQ, STASCHD
  - STOQ, STOSCHD

71

This chart shows the possible global statuses for transactions. Like databases, a transaction can have multiple global statuses.



Global DB Access Type (ACCTYPE)

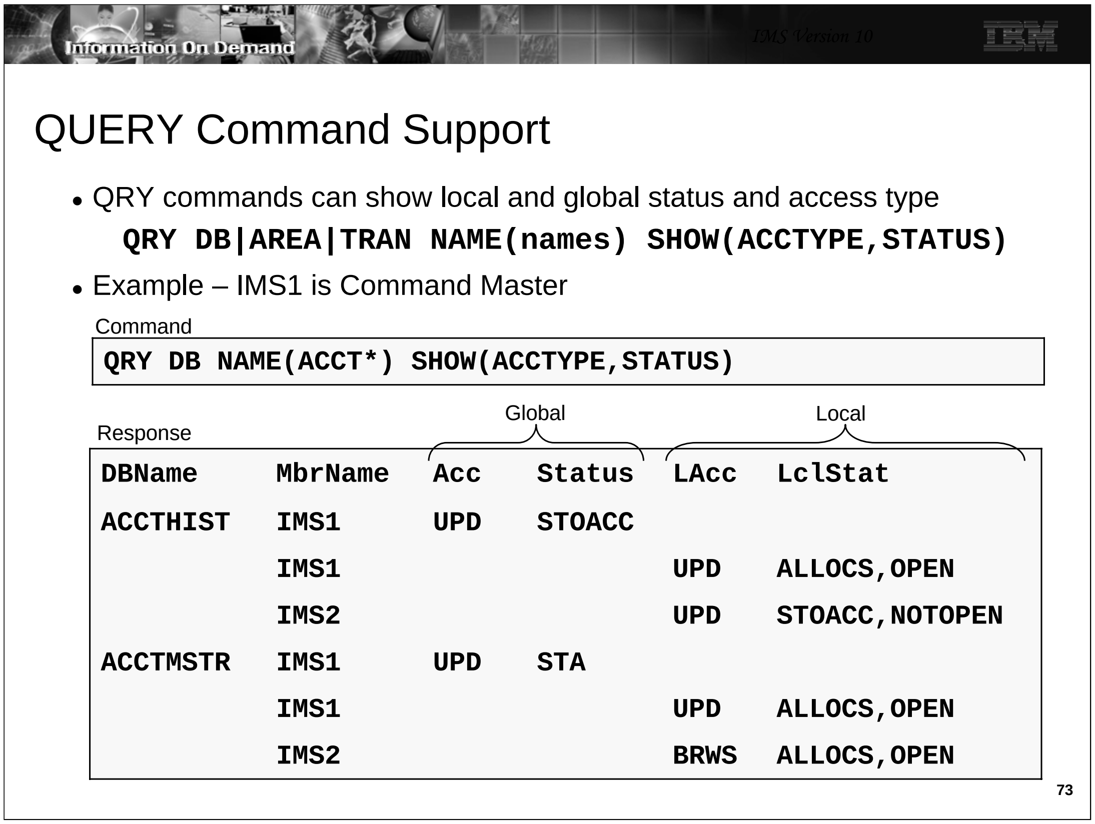
- Database access type can be updated globally

```
UPD DB NAME(xxx) SET(ACCTYPE(yyy)) SCOPE(ALL)
```

  - Global ACCTYPE is maintained in Resource Entry
    - BRWS | READ | EXCL | UPD
  - Type-1 command cannot be used to set global access type
    - /START DB xxx ACCESS=yy GLOBAL is invalid

ACCTYPE is the only DB “attribute” that is maintained globally. If an UPDATE DB NAME(XXX) SET(ACCTYPE(yyyy)) SCOPE(ALL) is issued, the access type of yyyy is set in the DB entry. One is created if it doesn’t already exist. IMSs being started will set their ACCTYPE to yyyy if they have not been active since the global setting was changed.

It is invalid to include both the ACCESS= and GLOBAL keywords in a type-1 command.



QUERY Command Support

- QRY commands can show local and global status and access type

  QRY DB|AREA|TRAN NAME(names) SHOW(ACCTYPE,STATUS)

- Example – IMS1 is Command Master

Command

```
QRY DB NAME(ACCT*) SHOW(ACCTYPE,STATUS)
```

Response

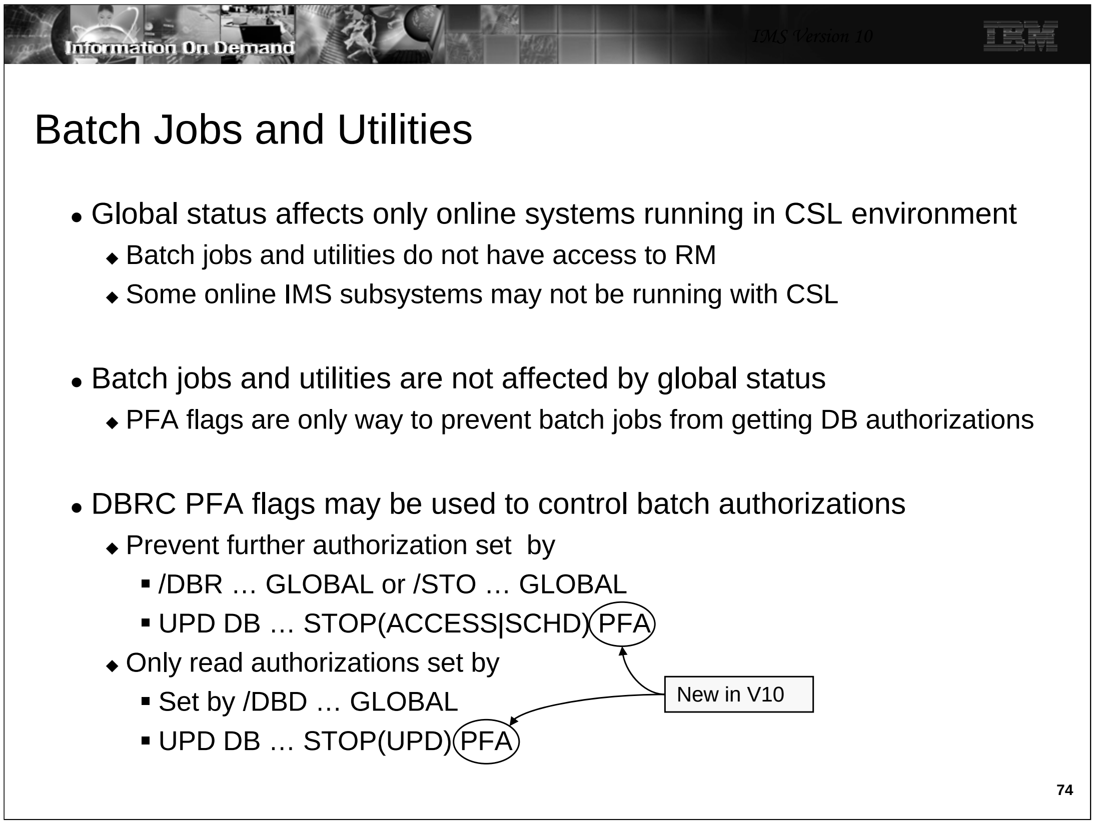
```
                             Global                  Local
DBName     MbrName   Acc    Status    LAcc   LclStat
ACCTHIST   IMS1      UPD    STOACC
           IMS1                       UPD    ALLOCS,OPEN
           IMS2                       UPD    STOACC,NOTOPEN
ACCTMSTR   IMS1      UPD    STA
           IMS1                       UPD    ALLOCS,OPEN
           IMS2                       BRWS   ALLOCS,OPEN
```

This is an example of a command showing local and global DB status and local and global DB access type. Because neither LOCAL nor GLOBAL was specified, both are shown in the response.

The response could have been created by the following actions.

A /DBR DB ACCTHIST GLOBAL or a UPD DB NAME(ACCTHIST) STOP(ACCESS) SCOPE(ALL) command was issued. If the UPD DB command was used, it was routed to both IMS systems. This set the global status for the database to STOACC. It also caused the database to be closed on both IMS1 and IMS2. Later a /START DB ACCTHIST or an UPD DB NAME(ACCTHIST) START(ACCESS) SCOPE(ACTIVE) command was issued only on IMS1. This started the database locally on IMS1. The database was then allocated and opened. The QRY command shows the results for ACCTHIST.

Database ACCTMSTR was started on both systems with a global status of STA and a global access type UPD. A /DBD DB ACCTMSTR or a UPD DB NAME(ACCTMSTR) SET(ACCTYPE(BRWS)) was sent only to IMS2. This caused IMS2 to change its local access type to BRWS which is read-only. It also closed and reopened the database data sets for read. The QRY command shows the results for ACCTMSTR.


Information On Demand
IMS Version 10

Batch Jobs and Utilities

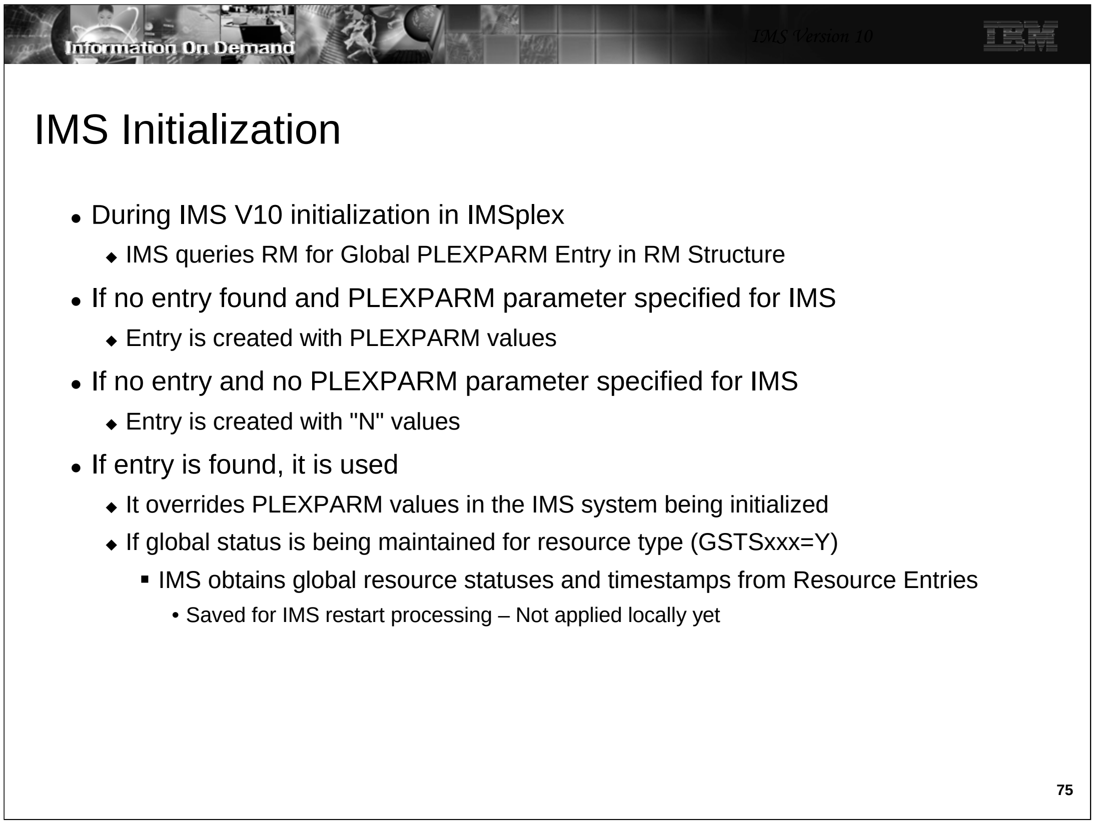
- Global status affects only online systems running in CSL environment
  - Batch jobs and utilities do not have access to RM
  - Some online IMS subsystems may not be running with CSL
- Batch jobs and utilities are not affected by global status
  - PFA flags are only way to prevent batch jobs from getting DB authorizations
- DBRC PFA flags may be used to control batch authorizations
  - Prevent further authorization set by
    - /DBR … GLOBAL or /STO … GLOBAL
    - UPD DB … STOP(ACCESS|SCHD) PFA
  - Only read authorizations set by
    - Set by /DBD … GLOBAL
    - UPD DB … STOP(UPD) PFA

New in V10

The PFA (prohibit further authorization) flag in DBRC prevents further authorization of a database to another IMS subsystem, including batch jobs. Because non-online IMS systems do not run with CSL and do not have access to the global status of a database, a global command such as UPD DB NAME(XYZ) STOP(ACCESS) SCOPE(ALL) would not prevent a batch job from getting authorization to that database. Only the PFA flag can do that. It also prevents online IMS systems that are not running with CSL and global status support from being able to authorize the DB. The point is global status is effective only within an IMSplex running with CSL and global status enabled.

DEDB areas also have a PFA flag, but cannot be accessed by batch jobs so it is seldom as much of an issue.

As we will see in the commands section of the class, the type-2 UPD DB class has been enhanced in V10 to set PFA flags.



Information On Demand
IMS Version 10
IBM

IMS Initialization

- During IMS V10 initialization in IMSplex
  - IMS queries RM for Global PLEXPARM Entry in RM Structure
- If no entry found and PLEXPARM parameter specified for IMS
  - Entry is created with PLEXPARM values
- If no entry and no PLEXPARM parameter specified for IMS
  - Entry is created with "N" values
- If entry is found, it is used
  - It overrides PLEXPARM values in the IMS system being initialized
  - If global status is being maintained for resource type (GSTSxxx=Y)
    - IMS obtains global resource statuses and timestamps from Resource Entries
      - Saved for IMS restart processing – Not applied locally yet

75

During initialization, IMS queries RM for the global PLEXPARM entry and takes action depending on whether or not the entry exists.

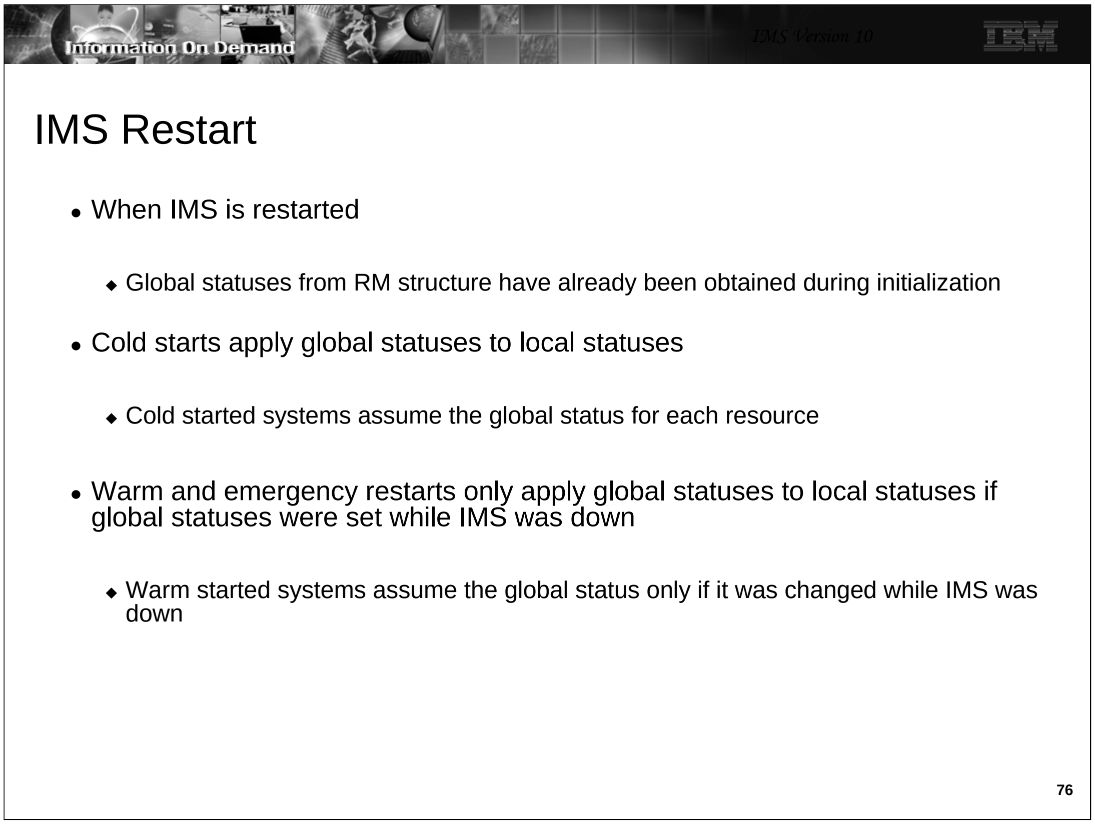
If the entry does not exist, IMS will create one according to the type of IMS it is and according to the PLEXPARM parameter in DFSCGxxx of DFSDFxxx. Note that the default for each of the parameters is N (no global status to be maintained).

-DB/DC will add an entry with values from PLEXPARM for all three resource types.

-DBCTL will add an entry with values from PLEXPARM for databases and areas, but will leave the transaction parameter NULL

-DCCTL will add an entry with values for transactions but leave databases and areas NULL

If there is no PLEXPARM parameter in DFSCGxxx of DFSDFxxx, IMS will set the GSTSxxx values to N or NULL, depending on subsystem type.


Information On Demand
IMS Version 10

IMS Restart

- When IMS is restarted
  - Global statuses from RM structure have already been obtained during initialization
- Cold starts apply global statuses to local statuses
  - Cold started systems assume the global status for each resource
- Warm and emergency restarts only apply global statuses to local statuses if global statuses were set while IMS was down
  - Warm started systems assume the global status only if it was changed while IMS was down

76

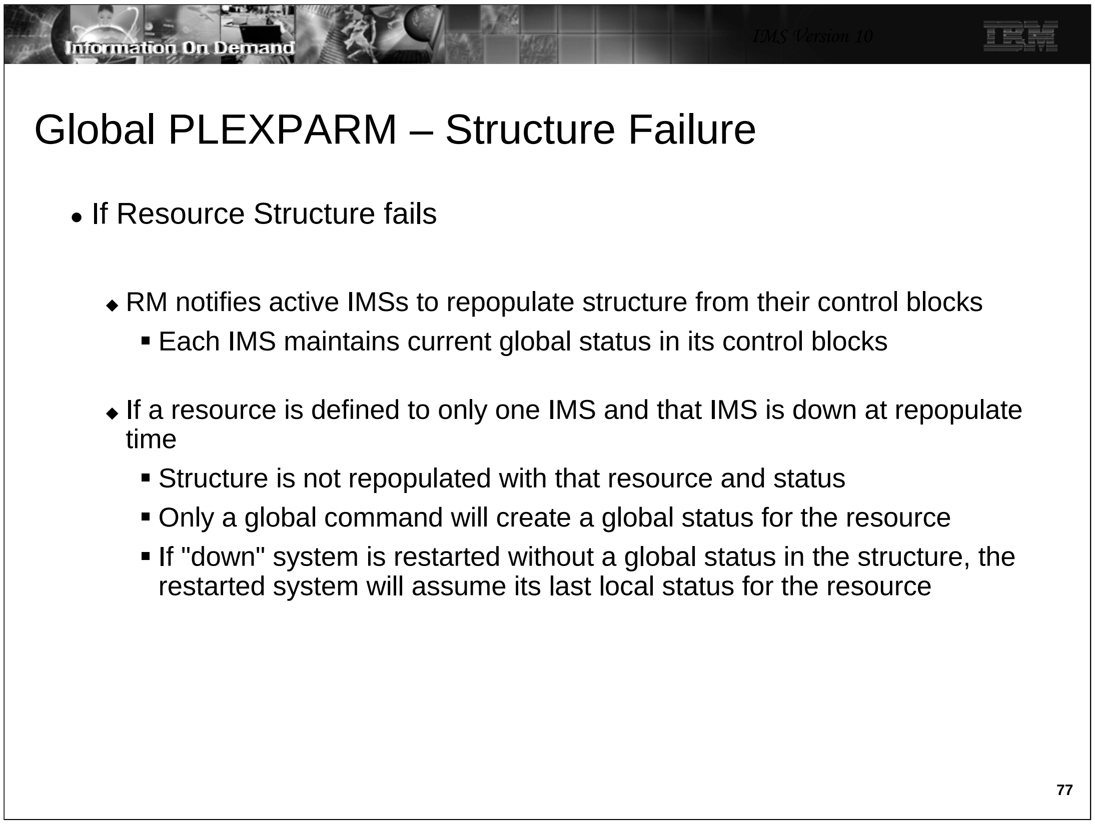
When IMS is restarted, it has already obtained the global values from the structure. It is during restart that IMS decides whether or not to apply this global status to the local control blocks.

-For cold starts IMS always sets the local status to the global status.

-For warm and emergency restart, the local status from the log records used for restart are first applied. (1) If the global status read from the structure was set at a time when IMS was down, the global status is applied to the control block. This occurs because someone changed the status globally while IMS was down. If IMS had been up, it would have changed its status at the time. (2) If the global status was set before IMS went down, then the local status is retained. The status on this IMS must have been changed after the global status was set.

-For an /ERE COLDBASE restart, the database and area statuses are set according to the rules for cold start. Transaction statuses are set according to the rules for emergency restart.

-For an /ERE COLDCOMM restart, the transaction statuses are set according to the rules of cold start. Database and area statuses are set according to the rules of emergency restart.

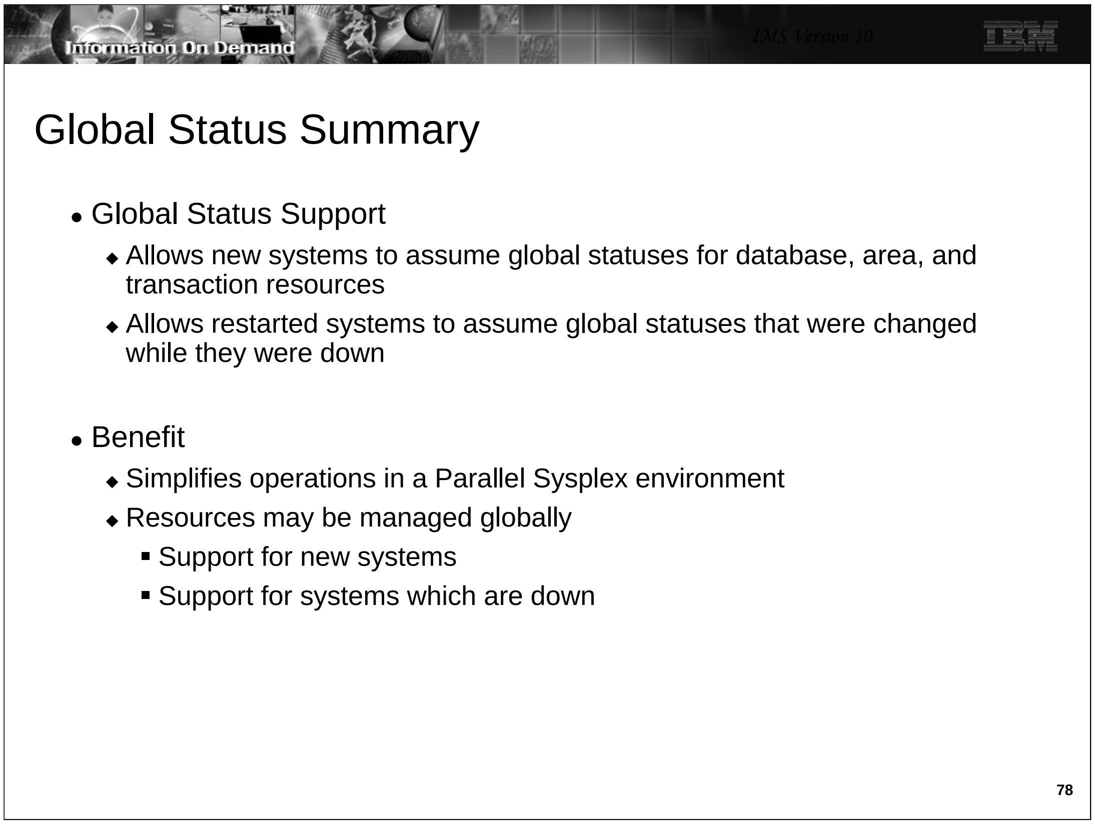

Information On Demand
IMS Version 10

Global PLEXPARM – Structure Failure

- If Resource Structure fails
  - RM notifies active IMSs to repopulate structure from their control blocks
    - Each IMS maintains current global status in its control blocks
  - If a resource is defined to only one IMS and that IMS is down at repopulate time
    - Structure is not repopulated with that resource and status
    - Only a global command will create a global status for the resource
    - If "down" system is restarted without a global status in the structure, the restarted system will assume its last local status for the resource

If the resource structure fails, RM notifies each IMSplex member to repopulate the structure from their control blocks. Remember, the control blocks have both the local and global status. Whichever IMS gets to the entry first sets the global status.

If a resource is defined to just one IMS, and that IMS is down at repopulate time (or all IMSs to which the resource is defined are down), then it cannot be repopulated. If a DB or AREA resource is later dynamically added to another IMS, there is no global status. If a transaction is dynamically entered by another IMS, a transaction entry is created with NULL status. If the DB or AREA resource then acquires global status, a DB or AREA entry is created with the appropriate global status. A transaction entry is updated with the global status. When the down IMS is restarted, it takes its global status from the structure.

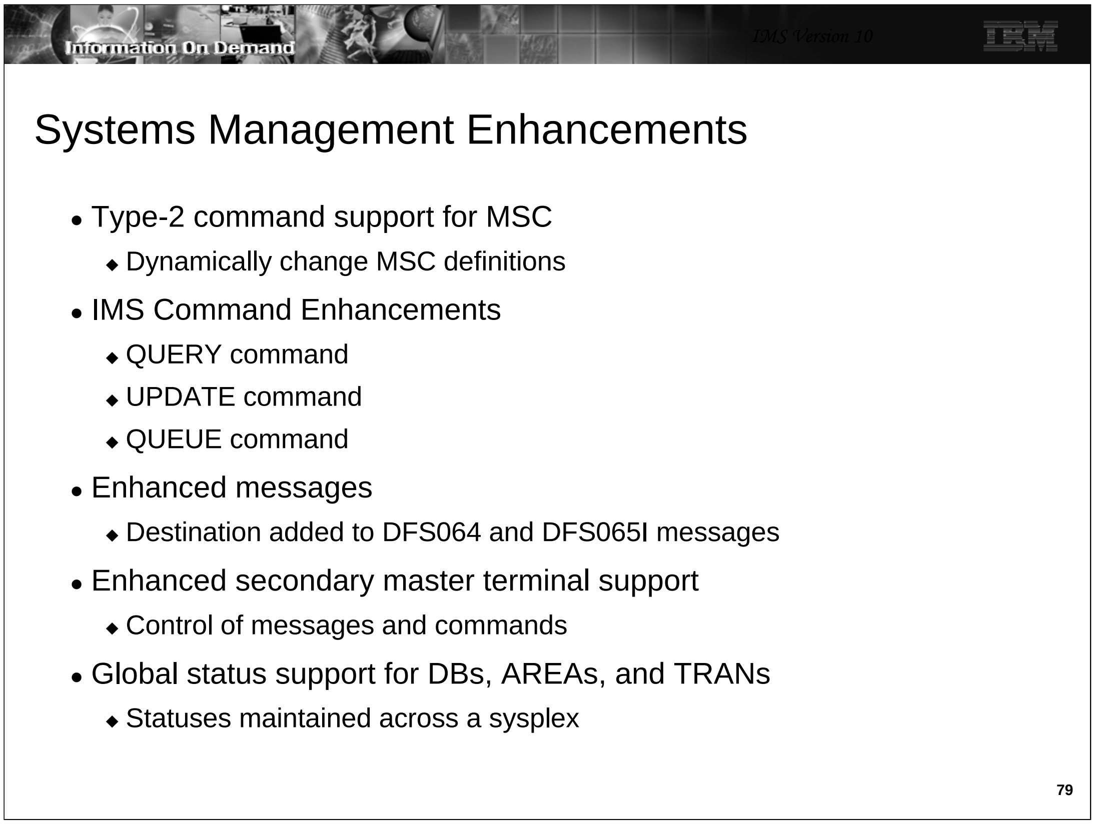


Information On Demand
IMS Version 10
IBM

Global Status Summary

- Global Status Support
  - Allows new systems to assume global statuses for database, area, and transaction resources
  - Allows restarted systems to assume global statuses that were changed while they were down
- Benefit
  - Simplifies operations in a Parallel Sysplex environment
  - Resources may be managed globally
    - Support for new systems
    - Support for systems which are down

The user can elect to maintain global statuses for databases, areas, and transactions in a Resource Structure. Global status is enabled initially by coding the GSTSxxx parms in the DFSCGxxx or DFSDFxxx PROCLIB member. It may be updated or added later by the UPD IMS command. Both type-1 and type-2 commands can set global statuses for databases and areas, Only type-2 commands can set global statuses for transactions. The benefit is that it allows the user to set a global status while some IMSs are down and then have an IMS accept that status when it is restarted.



Systems Management Enhancements

- Type-2 command support for MSC
  - Dynamically change MSC definitions
- IMS Command Enhancements
  - QUERY command
  - UPDATE command
  - QUEUE command
- Enhanced messages
  - Destination added to DFS064 and DFS065I messages
- Enhanced secondary master terminal support
  - Control of messages and commands
- Global status support for DBs, AREAs, and TRANs
  - Statuses maintained across a sysplex