

Mainframe - IMS DC support

- [Introduction](#)
- [Prototype link between GSAM DBD and JCL dataset](#)
- [IMS Transaction Manager file](#)
 - [Call link between Cobol Program and IMS Transaction](#)
 - [IMS Transaction Flow](#)
 - [Example](#)
- [IMS MFS Maps file](#)

Introduction

The IMS/DC (Data Communications) system provides users with online "real time" access to information in IMS databases. MPP transaction programs are NOT submitted with OS JCL. The online IMS Control Region automatically schedules the MPP program necessary to process a transaction. This page summarizes how the Mainframe Analyzer handles IMS/DC.



Support for IMS/DC was introduced in **CAST AIP 8.3.19**.

Prototype link between GSAM DBD and JCL dataset

Inside a PSB file, the PCB connects to the database:

```
PCB TYPE=GSAM,DBDNAME=GSAM01,PROCOPT=A,PCBNAME=GSAPCB

*----- PSBGEN -----*
PSBGEN LANG=COBOL,PSBNAME=COBIMSB1
PRINT NOGEN
```

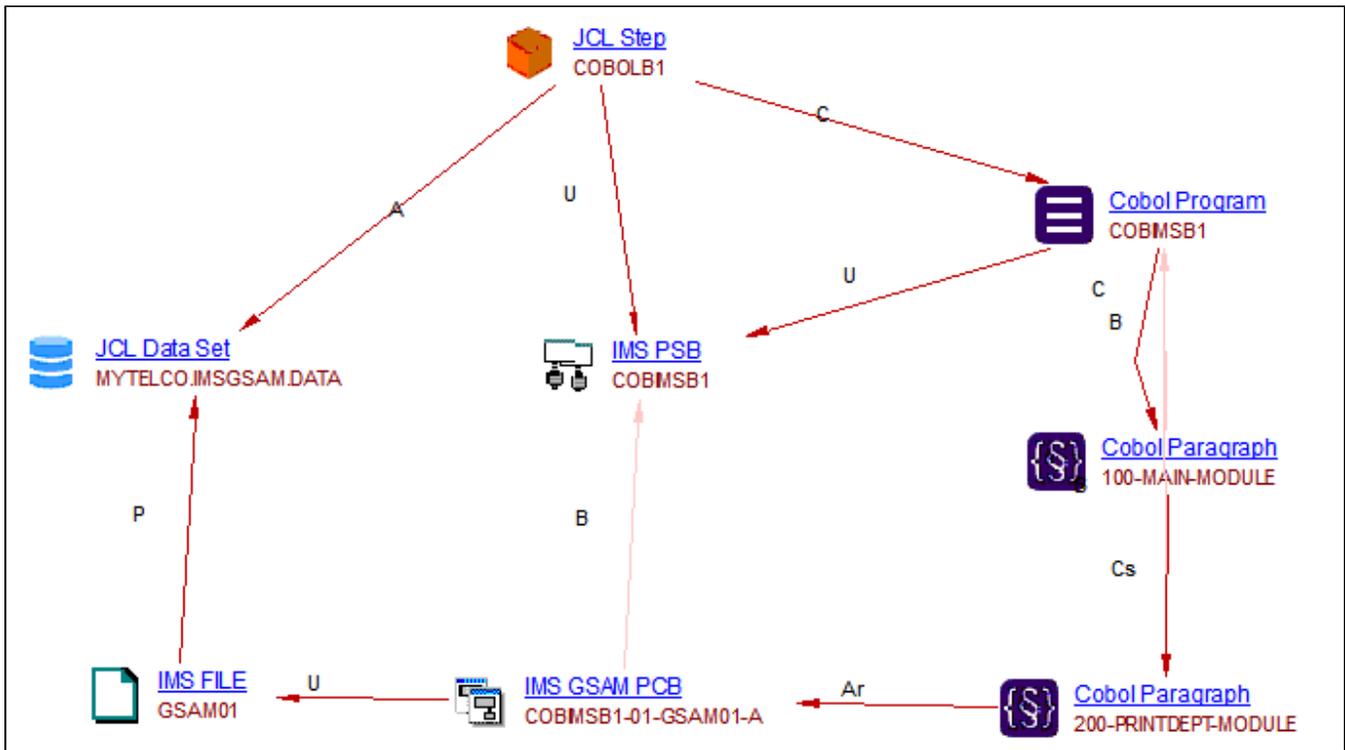
Then we have the definition of the DBD IMS file:

```
DBD NAME=GSAM01,ACCESS=(GSAM,BSAM), X
VERSION='09/22/0812.01'
DSG1 DATASET DD1=GSAM01, X
DD2=GSAM01,RECFM=FB
DBDGEN
FINISH
```

This IMS file calls the dataset of the JCL file, that calls the PSB file. The logic dataset name is set as GSAM01 and this is found when creating the JCL dataset:

```
//COBOLB1 EXEC PGM=DFSRR00,PARM='DLI,COBIMSB1,COBIMSB1'
//STEPLIB DD DSN=SYSCAST.IMS.SDFSRESL,DISP=SHR
//* * IMS DBD AND PSB LIBS
//IMS DD DSN=APPCAST.IMS.PSBLIB,DISP=SHR
// DD DSN=APPCAST.IMS.DBDLIB,DISP=SHR
//* * IMS STATISTICS
//DFSSTAT DD SYSOUT=P
//* * IMS SYSTEM LOG
//IEFRDER DD DUMMY
//GSAM01 DD DSN=MYTELCO.IMSGSAM.DATA,DISP=OLD
```

The Mainframe Analyzer will handle this scenario by linking the **IMS file** to the **JCL dataset** via a **prototype link**:



IMS Transaction Manager file

The Mainframe Analyzer will analyze the IMS Transaction file (*.tra) to resolve two macros:

- [APPLCTN macro](#)
- [TRANSACT macro](#)

Call link between Cobol Program and IMS Transaction

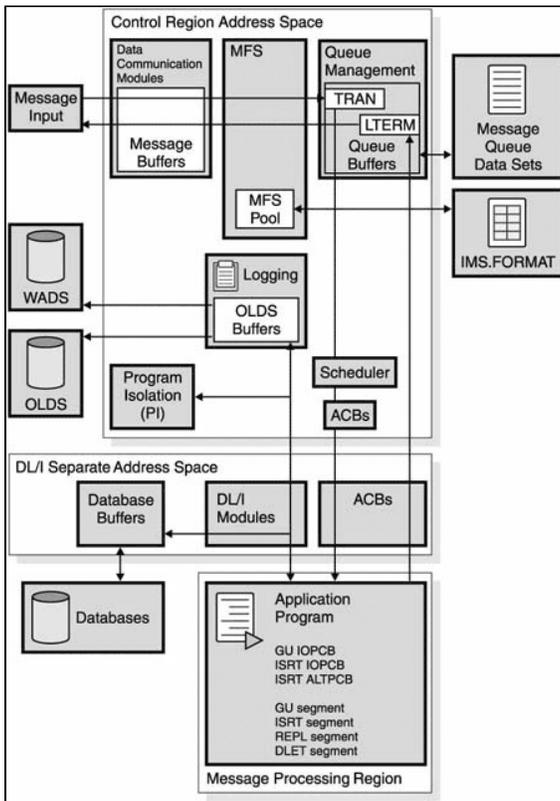
Code example:

```
APPLCTN PSB=IMMBHI,PGMTYPE=TP,SCHDTYP=PARALLEL 06036000
TRANSACT CODE=IMMBHI,MSGTYPE=(MULTSEG,RESPONSE,65), *06037000
PARLIM=0,SEGNO=400, *06038000
MODE=SNGL,PROCLIM=(03,01) 06039000
```

In this example, when the TRANSACT macro is created as code, the Mainframe Analyzer will create an IMS transaction object with the code name, then link it to the Cobol program named as PSB = XXX inside the APPLCTN macro.

IMS Transaction Flow

[Click to enlarge](#)



The input data from the terminal is read by the data communication modules. After editing by MFS, if appropriate, and verifying that the user is allowed to execute this transaction, this input data is put in the IMS message queue. The messages in the queue are sequenced by destination, which could be either transaction code (TRAN) or logical terminal (LTERM).

- **I/O PCB** - in a CICS-DBCTL environment, an input/output PCB (I/O PCB) is needed to issue DBCTL service requests. Unlike other types of PCB, it is not defined with PSB generation. If the application program is using an I/O PCB, this has to be indicated in the PSB scheduling request, as explained in [Format of a PSB](#).
- **Alternate TP PCB(s)** - an alternate TP PCB defines a logical terminal and can be used instead of the I/O PCB when it is necessary to direct a response to a terminal. Alternate TP PCBs appear in PSBs used in a CICS-DBCTL environment, but are used only in an IMS/VS DC or IMS™ environment. CICS applications using DBCTL cannot successfully issue requests that specify an alternate TP PCB, an MSDB PCB, or a GSAM PCB, but PSBs that contain this kind of PCB can be scheduled successfully in a CICS-DBCTL environment. Alternate PCBs are included in the PCB address list returned to a call level application program. The existence of alternate PCBs in the PSB can affect the PCB number used in the PCB keyword in an EXEC DLI application program, depending on whether you are using CICS online, batch programs, or BMPs



Source:

- <https://www.slideshare.net/srinipdn/ims-dc-self-study-complete-tutorial>
- <https://communities.bmc.com/docs/DOC-9973?jsessionid=00BCE1264A8E5C01EEE4938CAC6A80F6.node0>
- <https://www.slideshare.net/IBMIMS/coding-mpp-programs-to-process-ims-transactions-66684760>

Example

From a Cobol file, the Alternat-PCB contains the destination code (IMS Trans code or LTERM code). Example ALT-PCB:

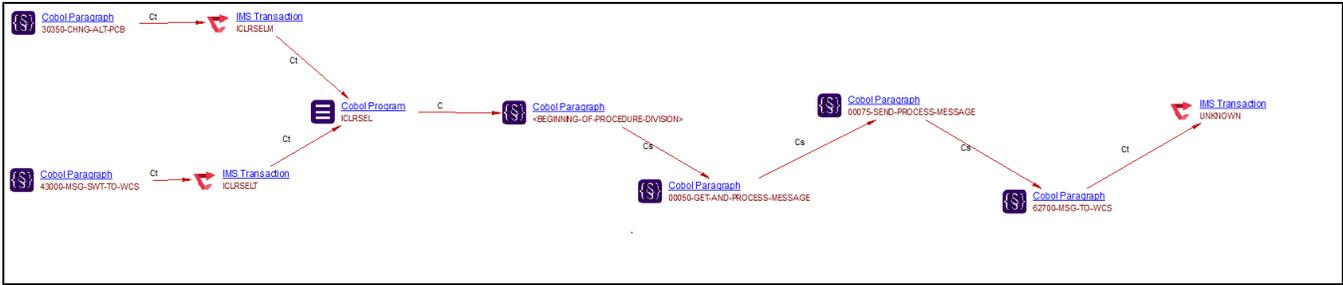
```

01  GNSALTIO-PCB.
02  GNSALTIO-DESTCODE      PIC  X(008) VALUE 'ICLRSELT' .
02  GNSALTIO-IMS-RESERVED  PIC  X(002) .
02  GNSALTIO-STATUS-CODE   PIC  X(002) .
02  GNSALTIO-CURRENT-DATE  PIC  S9(007)  COMP-3 .
02  GNSALTIO-CURRENT-TIME  PIC  S9(007)  COMP-3 .
02  GNSALTIO-MSG-SEQ-NUMBER PIC  S9(008)  COMP .
02  GNSALTIO-MOD-NAME      PIC  X(008) .
02  GNSALTIO-USER-ID       PIC  X(008) .

```

Only the **CHNG DLI** Call could change the value of destination code. The **ISRT DLI** Call will send the message to LTERM or IMS Transaction:

Click to enlarge



IMS MFS Maps file

 Support introduced in CAST AIP 8.3.25.

The Mainframe Analyzer will analyze the IMS MFS Maps file (*.mfs) so that it is possible to find out which Cobol programs use an MFS Map:

- MFS Maps are contained in files with the extension *.mfs.
- FMT macro defines the map (called "format" in IMS vocabulary).
- MSG macro defines MID and MOD messages. MID are those that have the INPUT type and MOD are those that have the OUTPUT type.
- MID and MOD identifiers are specified in the IO-PCB.
- In the MID/MOD structure, there is a field that contains the name of the transaction. This information allows the analyzer to create links between MFS Maps and transactions

As a result:

- See [Mainframe Discoverer](#) for more information about how the discoverer handles *.mfs files.
- .mfs files have been added to the list of files that will be automatically analyzed - see for example [Mainframe - Analysis configuration](#)
- New **object types** will be resolved - see [Mainframe - Analysis results](#):

	IMS Message Format Service
	IMS Message Input Descriptor
	IMS Message Output Descriptor