

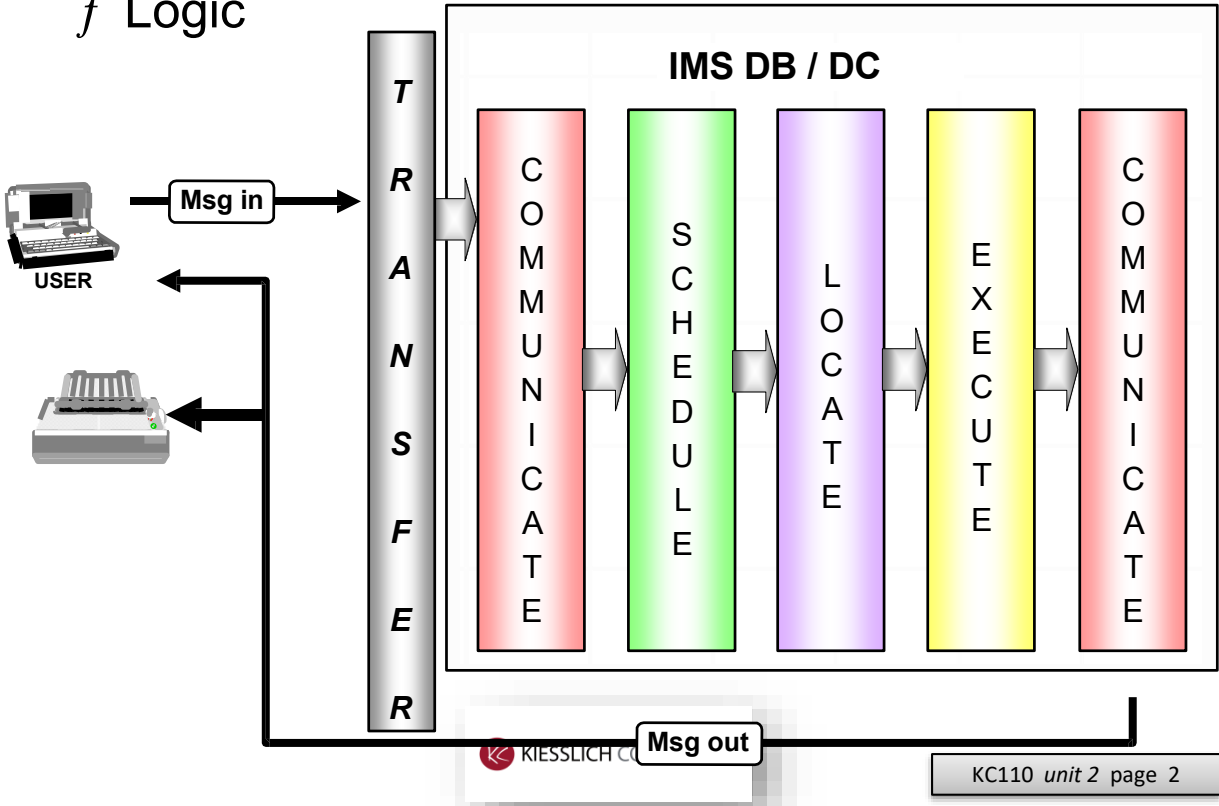
Unit 2

Transaction Flow and Message Flow

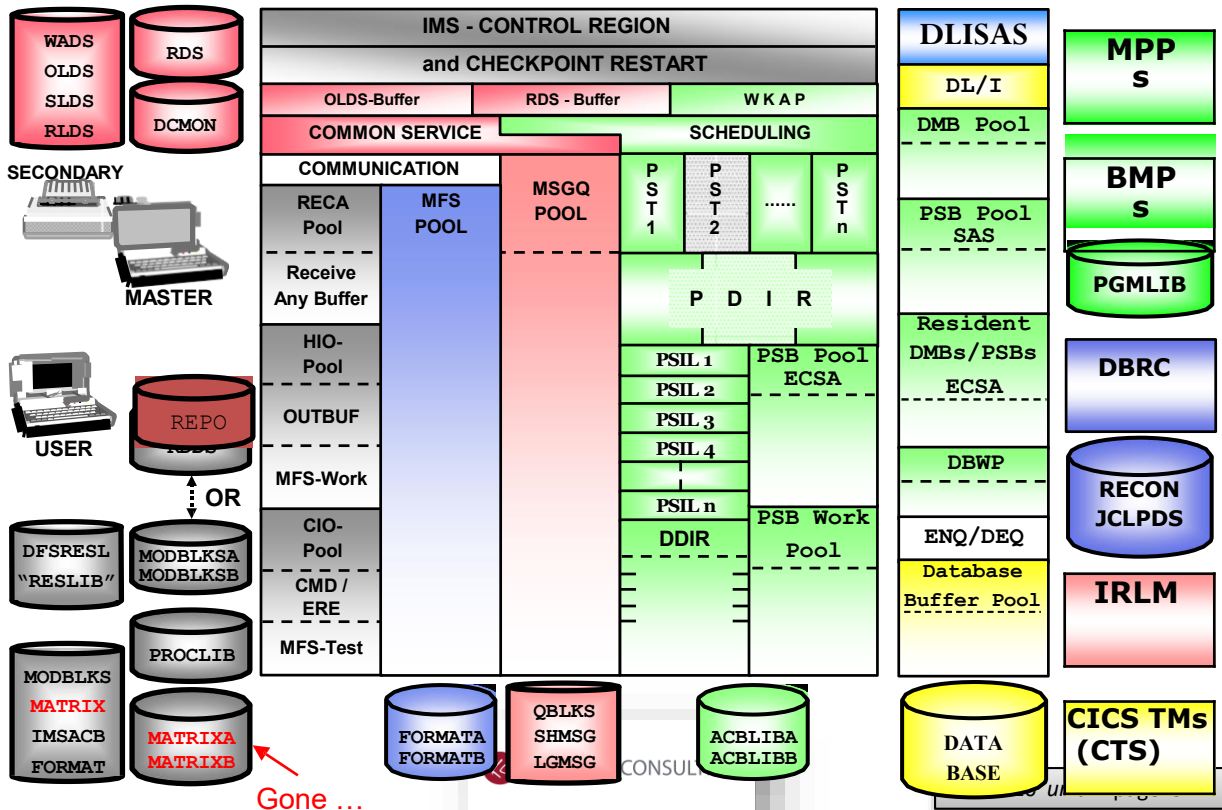
(applies to DB/DC and DCCTL environments)

Message processing overview

f Logic



IMS Transaction Flow overview



First : Starting IMS itself (1 of 2)

- /S ... PROC / ASID (as started task) - Or via JCL with JOB card
- Specify type of (Re)Start (IMS master terminal or z/OS console):
 - Input only permitted if AUTO=N parameter is used on start command
 - Otherwise, IMS will select either Warm or Emergency restart as appropriate
- ✓ COLD Start (/NRE CHKPT 0 [FORMAT ALL,...])
- ✓ WARM Start (/NRE [FORMAT ALL,...])
 - The *FORMAT* keyword parameter is rarely used on Warm starts
- ✓ EMERGENCY Restart (/ERE [FORMAT ALL,...])
 - The *FORMAT* parameter is one of numerous available for emergency restarts
 - See here:
- [/ERESTART command - IBM Documentation](#)

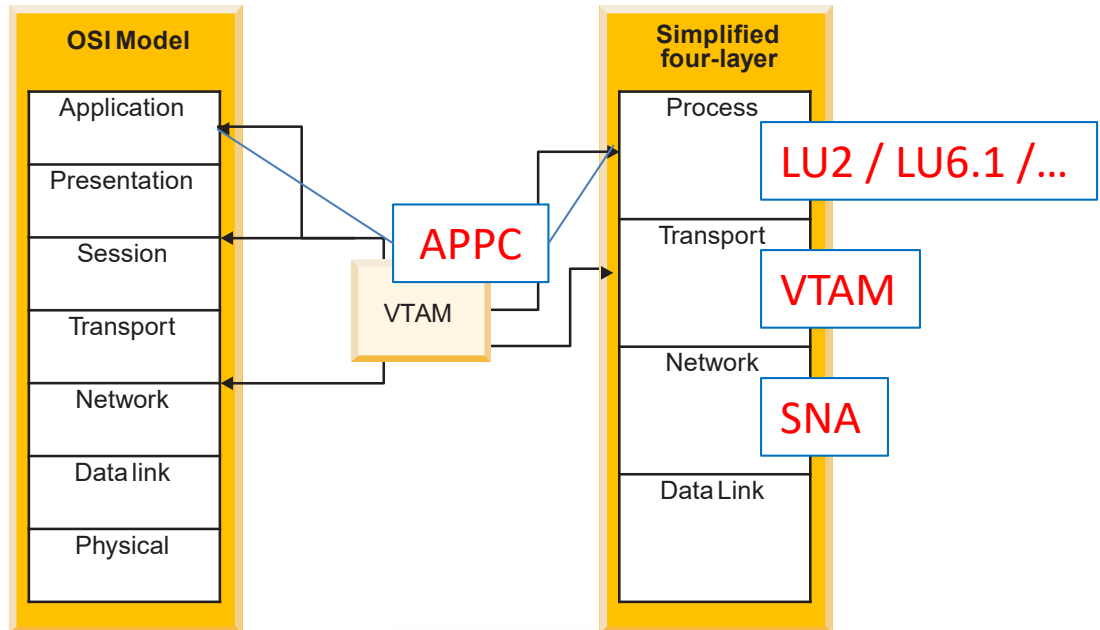
... Starting IMS (2 of 2)

... Then this will happen:

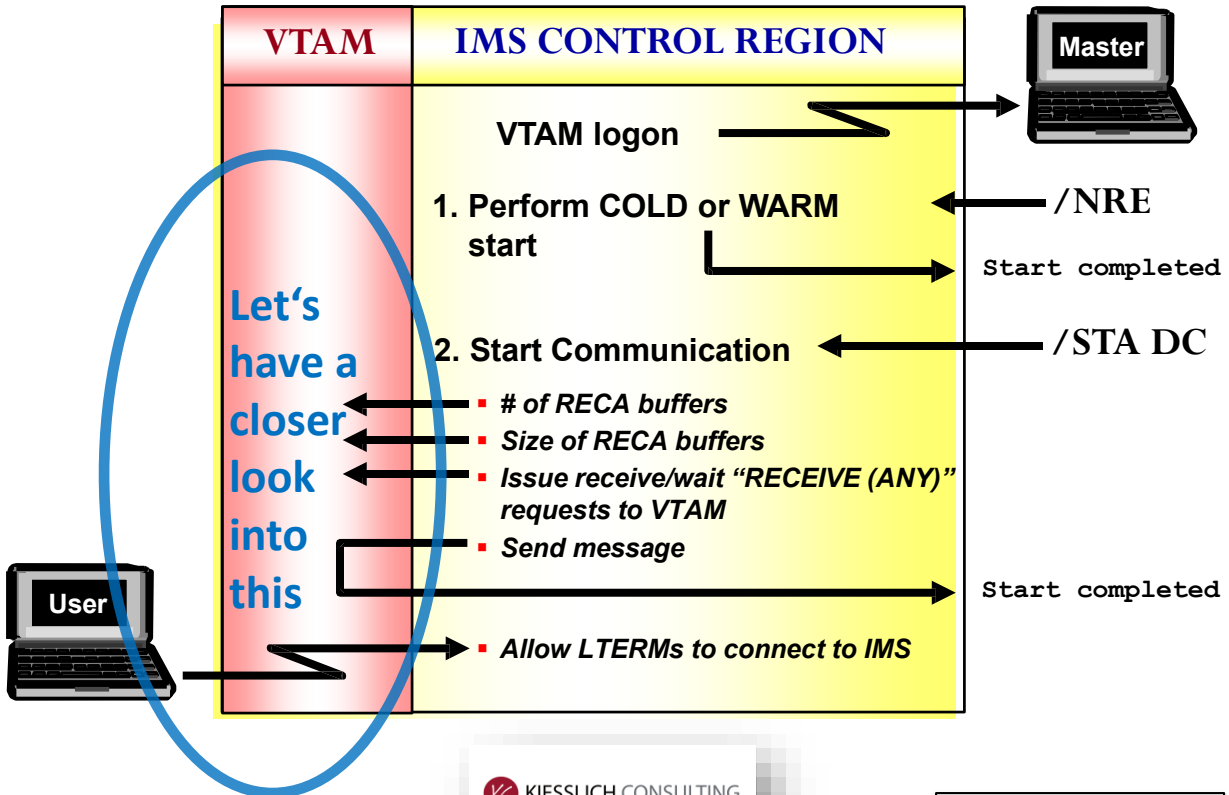
- IMS starts DLISAS and DBRC automatically based on PROC names (specified in parameters of the startup procedure in use)
- IMS initializes modules and storage (code, ctlblks and pools)
 - If Parameter RES=Y is specified, *RESident* PSBs and DMBs are also loaded
- Depending on MAXREGN= parameter in SYSGEN or PST= parameter in startup JCL (or parameter member) the blank PSTs are established
 - Up to MAXPST= parameter in execution JCL additional PSTs can be established dynamically and will be released at end of dependent region
- A first System Checkpoint is written to log
- IMS is *almost* ready to process work (input)
 - We still need to start dependent regions and connections to VTAM and IMS Connect (or other subsidiary ASIDs as ODBM,...)



IMS VTAM correlated to OSI



Start of communication [VTAM]



Input Message handling [VTAM]

RecAny Buffer (1 of 2)

E C S A

VTAM - IOBUF

M e s s a g e

M e s s a g e

M e s s a g e

M e s s a g e

Let's
have a
closer
look
here ...

IMSCTRL EPA

RECA POOL

RECA - Buffer

RECA - Buffer

RECA - Buffer

"RECEIVE ANY"
is a VTAM
Macro /
Request
issued by IMS.
The Receive Any
Buffer Address
(&size)
are **parameters** of
the request
(See VTA;M RPL)

if RECA not
available
GETMAIN

... when a RECA buffer is freed

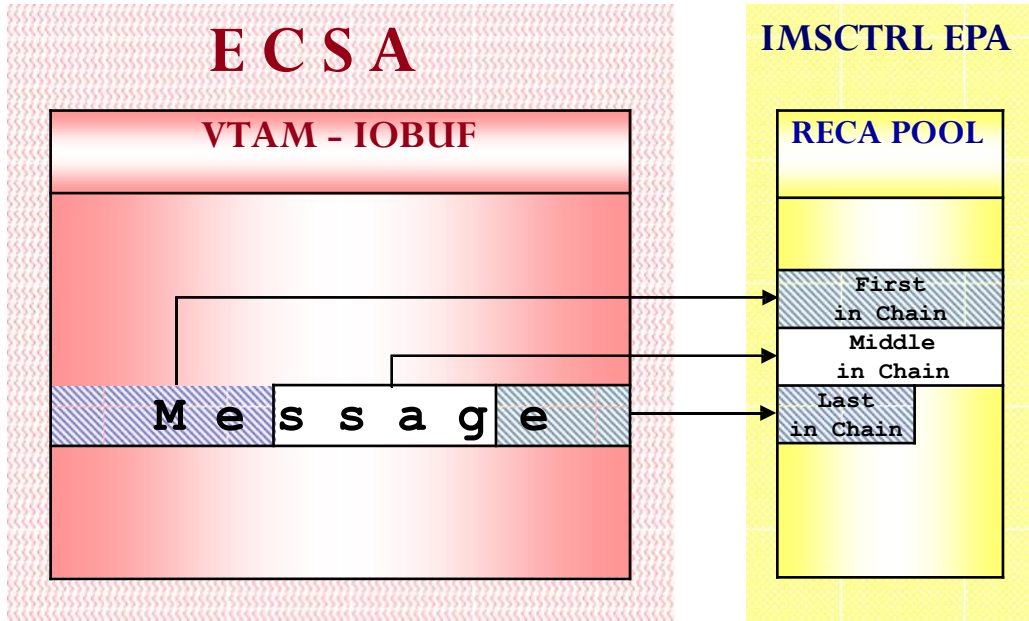
ESP229



KISSLICH CONSULTING

Input Message handling [VTAM]*

RecAny Buffer (2 of 2)



* OTMA Input Msg handling follows later



KISSLICH CONSULTING

IMS Message types

- The IMS "Communication Analyzer" classifies messages as:

TRANSACTIONS:

TRANcode	text
1	8 9 n

- Originate from terminal .. Or origin source (TPIPE, OTMA YTIB, APPC TIB) - most IMS/TM activity
- Can be originated from another application via *program-switch (PTP)*

MESSAGE-SWITCHES:

LTERMname	text
1	8 9 n

- Allows terminals to communicate with each other

COMMANDS:

/	COMMAND-VERB	text
1	2	n

- Mostly entered by Automation or *Master Terminal*
- Commands can be entered from user terminals
- Also from (authorized) APPLICATIONs (AOI)



IMS Message type examples

- Transactions:
 - Example 1:
 - User input
 - SALARY 123456
 - IMS Transaction Output
 - "Salary for John Doe (123456) is \$xxx.xx"
 - Example 2:
 - User input
 - ADDINV PART 34567
 - IMS Transaction Output
 - "Inventory for red wagons = 1492 units"
- Message Switch:
 - User input
 - SALSTERM SALES DEPARTMENT MEETING AT 3:30 PM
 - IMS TERMINAL Output at terminal ACTGTERM (NODE with LTERM called SALSTERM by name)
 - SALES DEPARTMENT MEETING AT 3:30 PM
- Command examples:
 - /CHE FREEZE
 - /STA REGION IMSMSGx

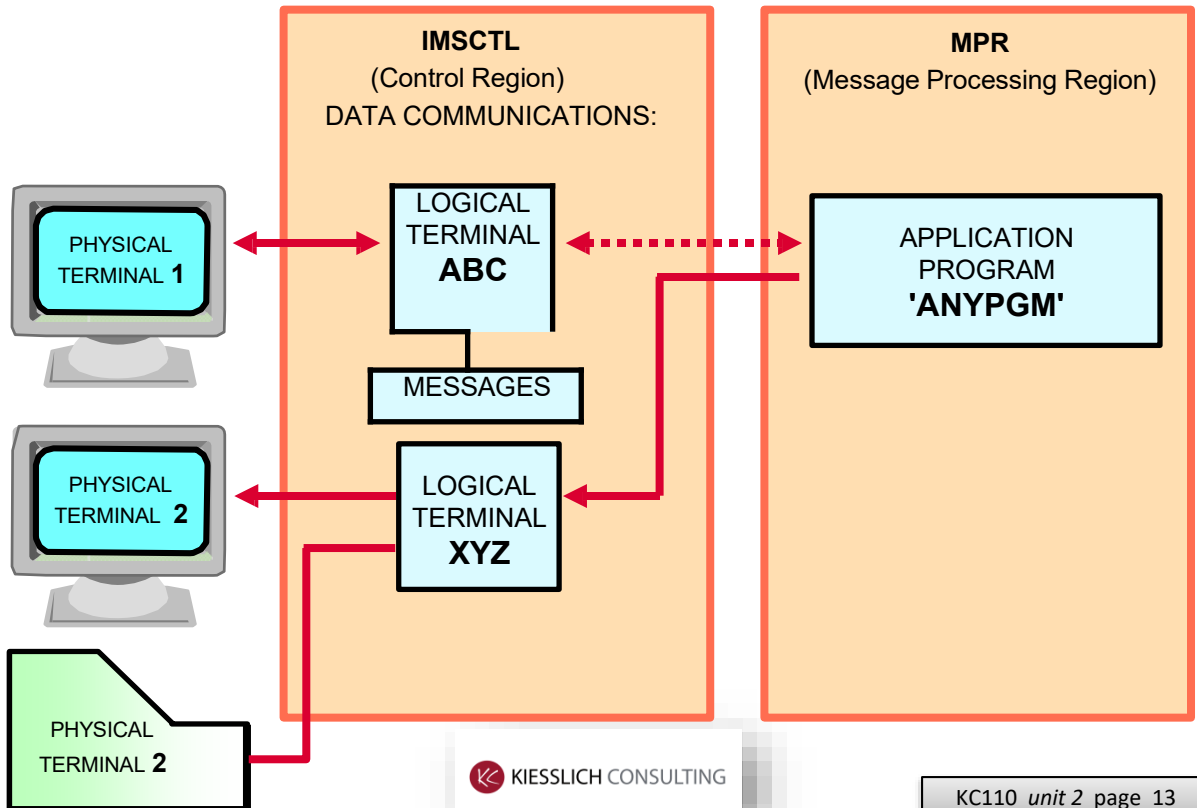
IMS Message destinations

- Transaction-type and Message Switch-type transactions:
 - Processed by the IMS Queue Manager (component of IMS) and placed onto the IMS message queue
 - Messages and therefore IMS message queues have two types of destinations:
 - **SMB** (Scheduler Message Block) destinations are the Queue Manager's representation of messages associated with a transaction code
 - **LTERM** (Logical TERMinal) destinations - more details later
 - IMS SYSGEN coding will determine whether a destination is an LTERM or an SMB
 - For example, "PAYROLL" could have been defined as an LTERM in IMSA and a Transaction (SMB) in IMSB
- Output messages from Application processing programs are also queued to LTERM destinations

[No IMS commands are placed into the message queues; commands are not considered as IMS Msgs → special command handler task]

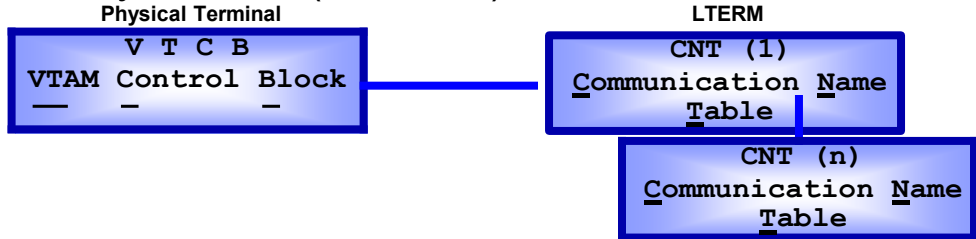


IMS Terminal concept [VTAM]



IMS Terminal related CBs – Static Defined [VTAM]

Statically Defined (IMSGEN) Devices

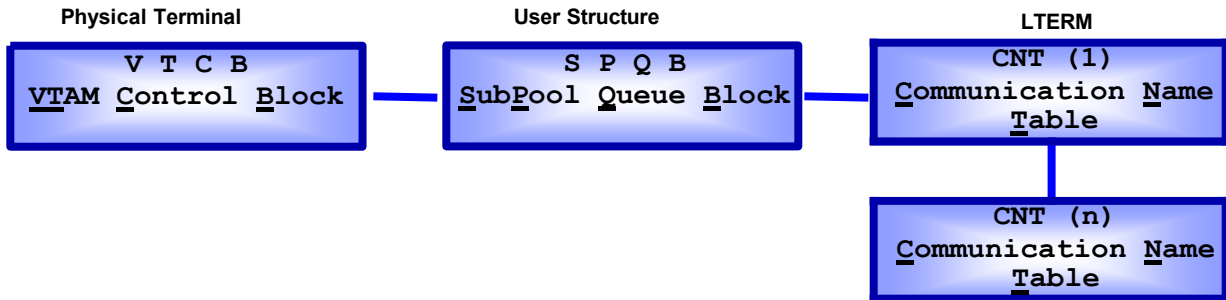


- There is one VTCB for each VTAM Terminal (*NODE*):
 - The VTCB is actually a collection of control blocks as : CLB, CTB, CRB, CCB and CIB
 - The name is specified in the IMSGEN *TERMINAL* macro
- The CNT control block represents an IMS LTERM:
 - The name of an LTERM/CNT is specified through the IMSGEN *NAME* macro
 - Multiple CNTs can be associated with each VTCB (CLB)
- The relationship between Physical terminal and LTERM can be changed by the /ASSIGN IMS Operator command



IMS Terminal related CBs: Dynamic (1 of 2) [VTAM]

Dynamically Defined (ETO) Devices



- The purpose of dynamic VTCBs and LTERMs is the same as for static devices
- The *User Structure* (SPQB) is an added control block type that only exists for ETO devices
 - The SPQB enables additional function such as the ability to associate multiple LTERMs to a signed-on user

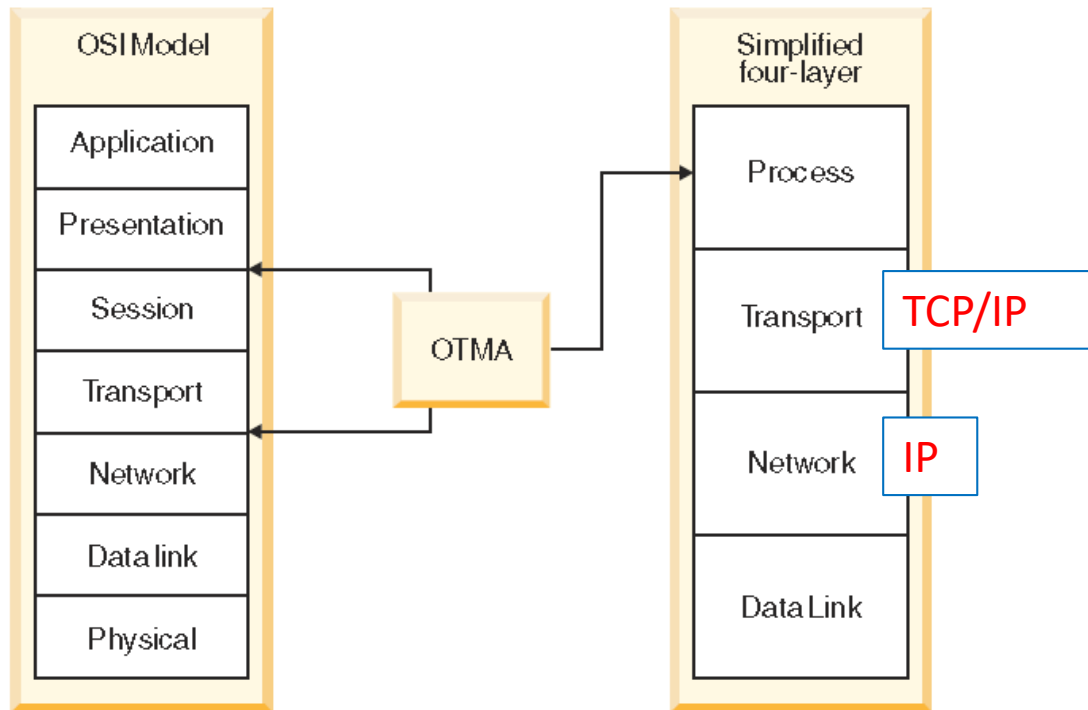


IMS Terminal related CBs: Dynamic (2 of 2) [VTAM]

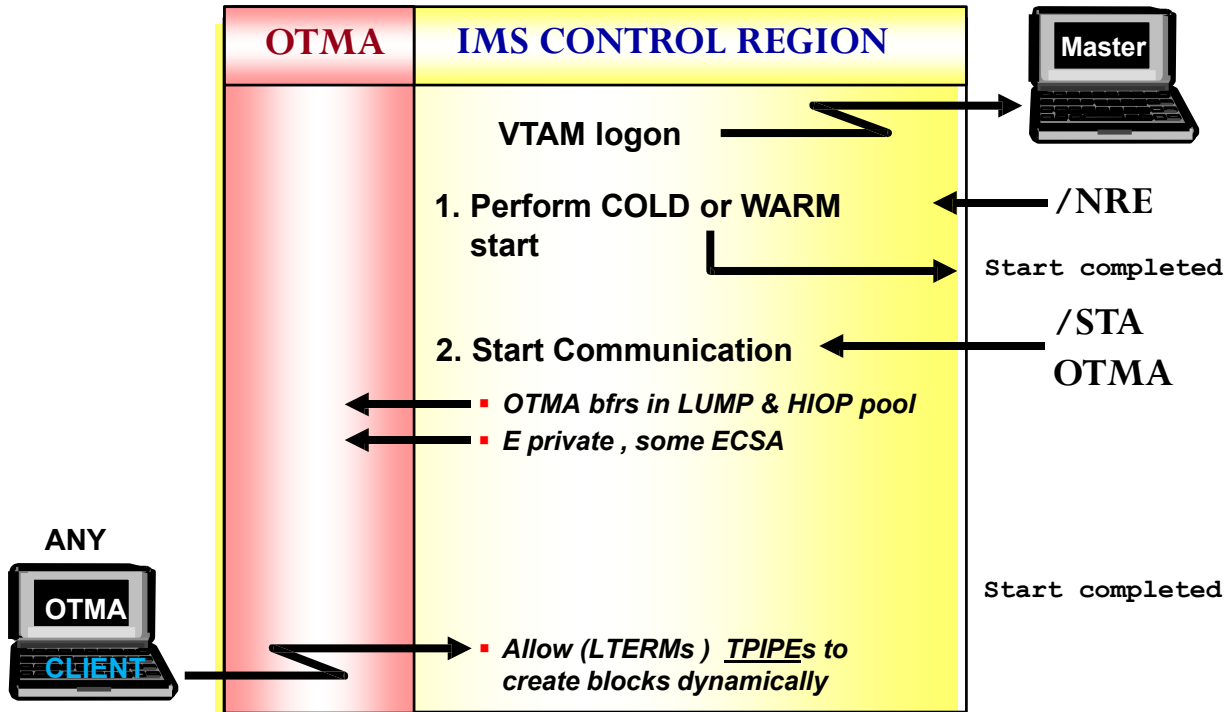
- When ETO : These three types of structures are created only when needed:
 - At user logon or signon time
 - When application output is created and queued
 - When these control blocks were deleted before (deleted when no longer needed -> timeout)
- The names used for these structures depend on installation rules defined through the various types of *ETO Descriptors* (that is, logon, user, device, and MSC) *models* and user exits



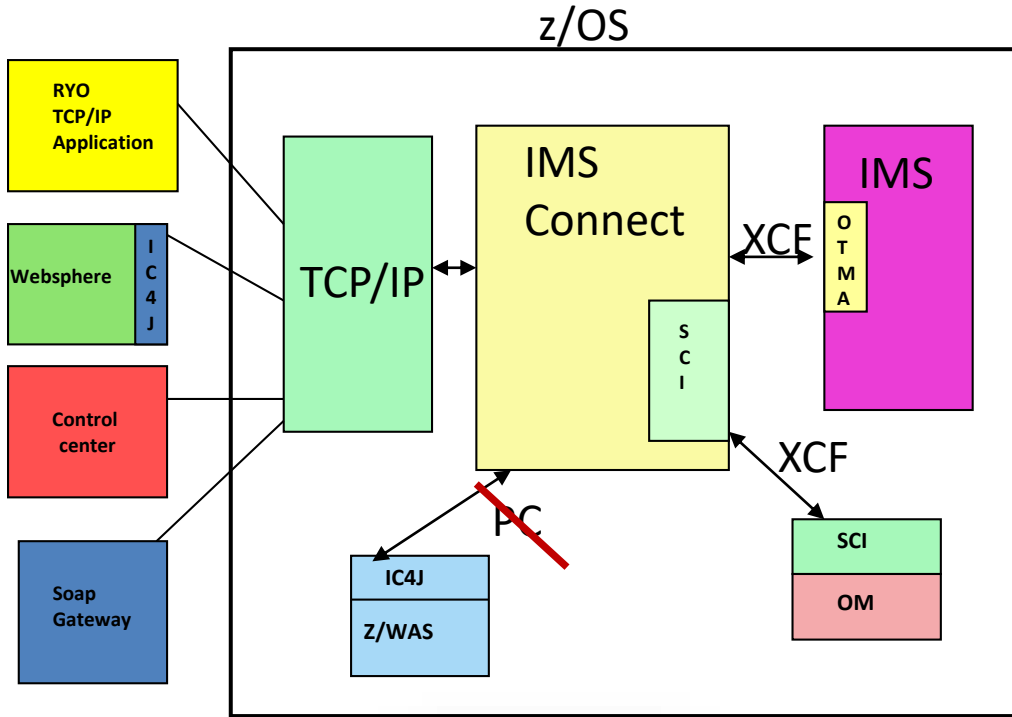
IMS OTMA correlated to OSI



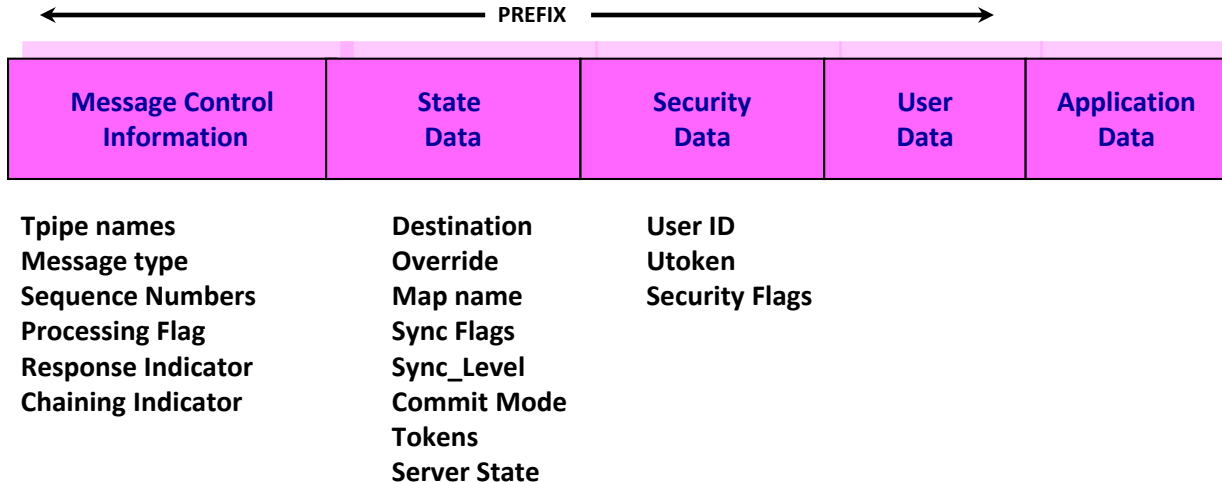
Start of communication [OTMA]



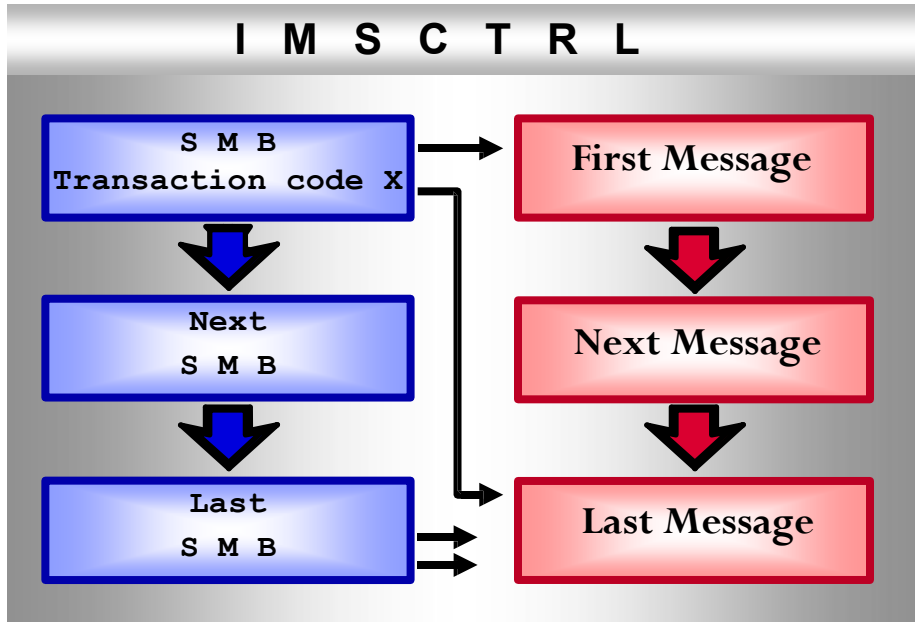
IMS Connect – An OTMA CLIENT



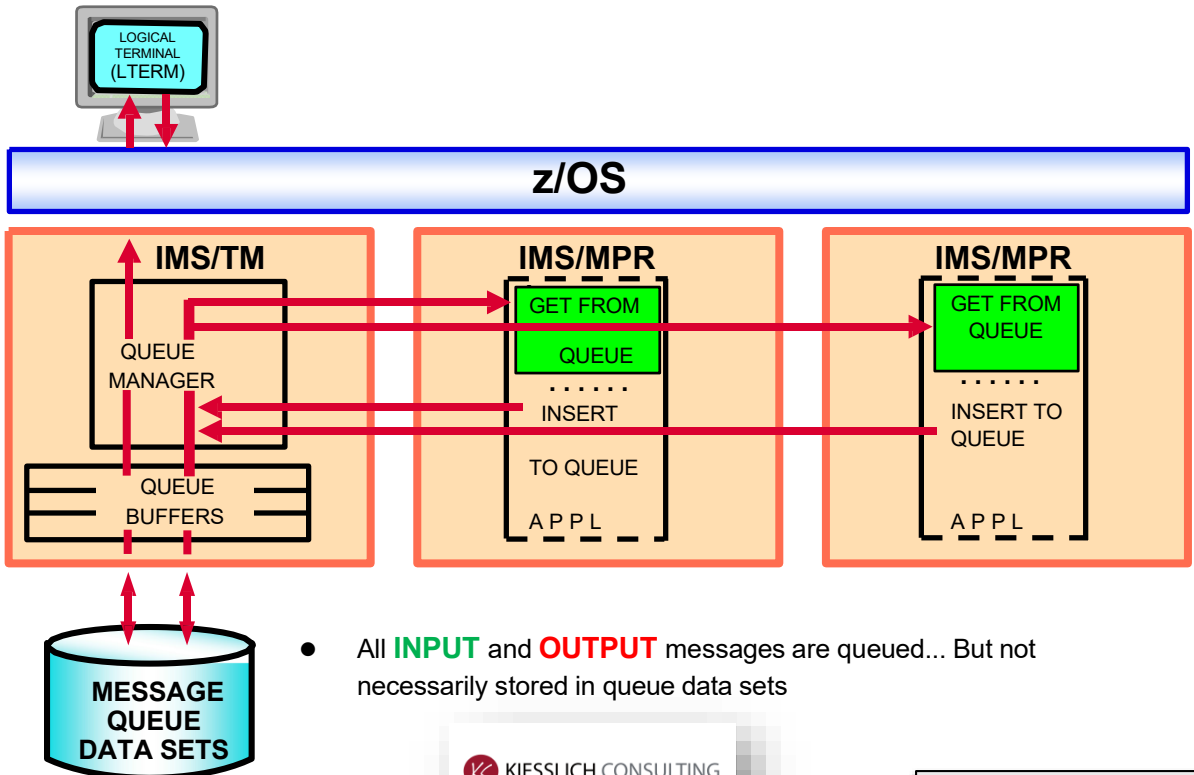
OTMA Message Structure



MSGQ: Input Message queuing (local)



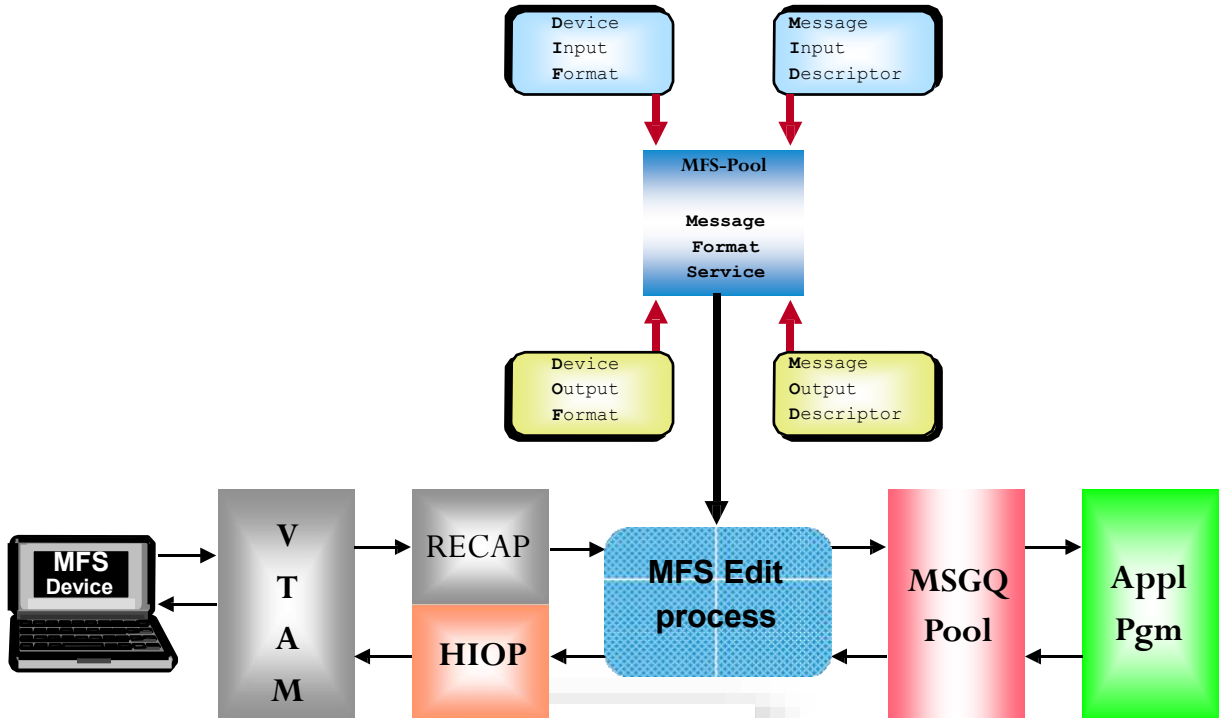
Message queue: **Input** message (1 of 2)



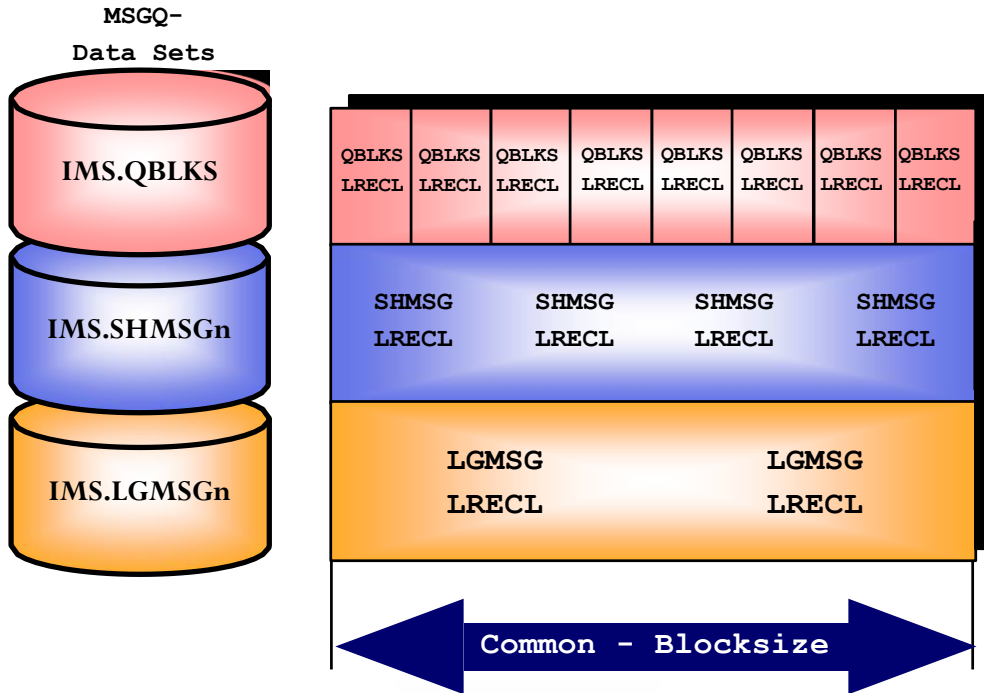
- All **INPUT** and **OUTPUT** messages are queued... But not necessarily stored in queue data sets



Message Format Service (MFS) block relationship



Message Queue datasets and LRECL sizes



MSGQ Pool structure and management (2 of 2)

- Common blocksize (MSGQ buffer size) is a multiple of the LGMSG LRECL
 - There is a slight performance benefit to having same value specified for the MSGQ buffer size = LGMSG LRECL (size=size2 below)
- Definition in MSGQUEUE macro of IMS Gen:

`BUFFERS=(nbr,size), RECLNG=(size1,size2)`

 - Startup override parameter values can be specified as well
 - Shared Queue parms only in Startup specifications – NOT in IMS Gen
- Up to 10 SHMSG and LGMSG data sets can be allocated:
 - *Queue Manager Concurrent I/O* added as performance enhancement in order to enables parallel I/O operations between multiple Message Queue Data sets
 - LRECL and BLKSIZE will be the same for all the data sets of a single queue data set type (that is, all SHMSG data sets have same LRECL)
 - Same SPACE allocation **should** be specified for the same data set type



Transaction processing: Conversational

