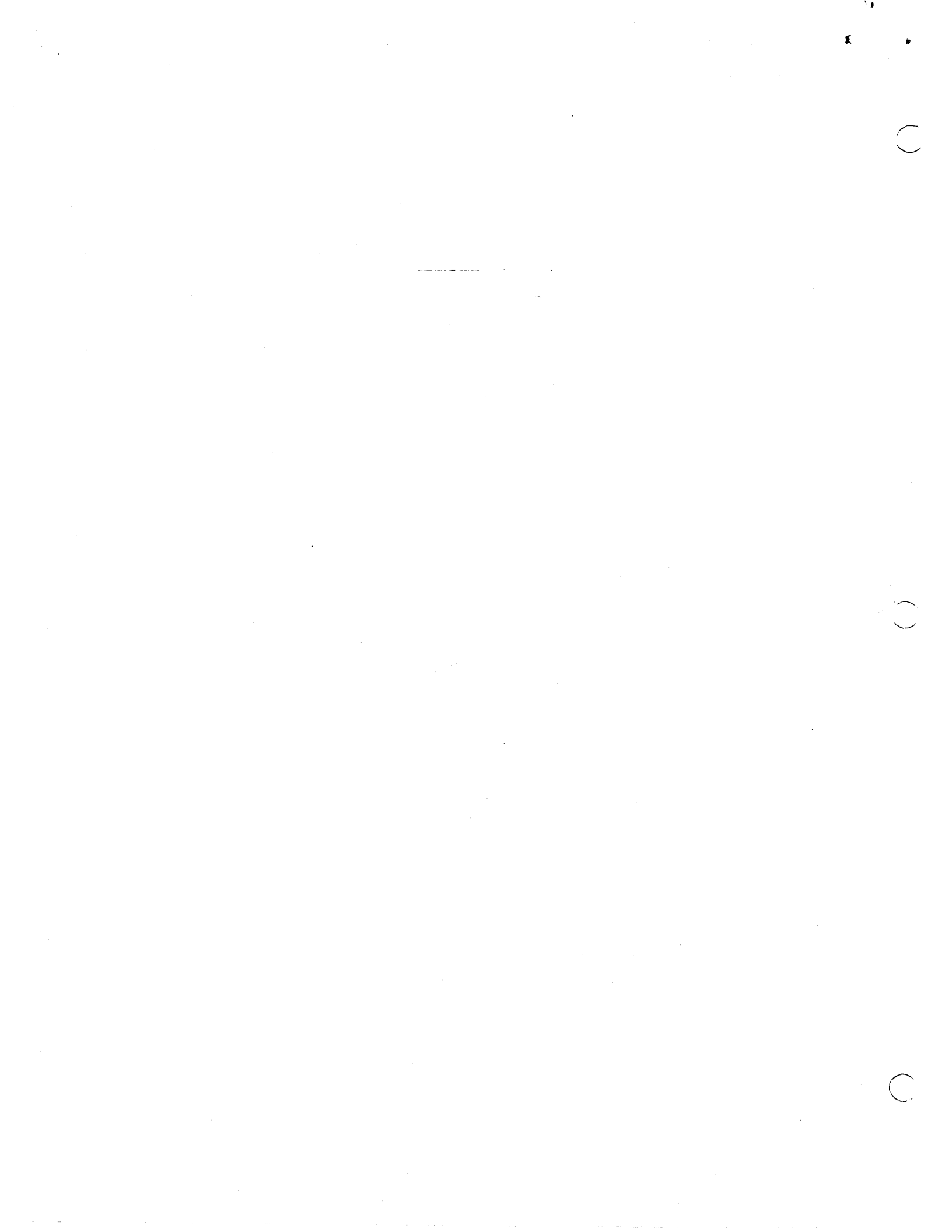


IMS/VIS

INTRODUCTION

P551

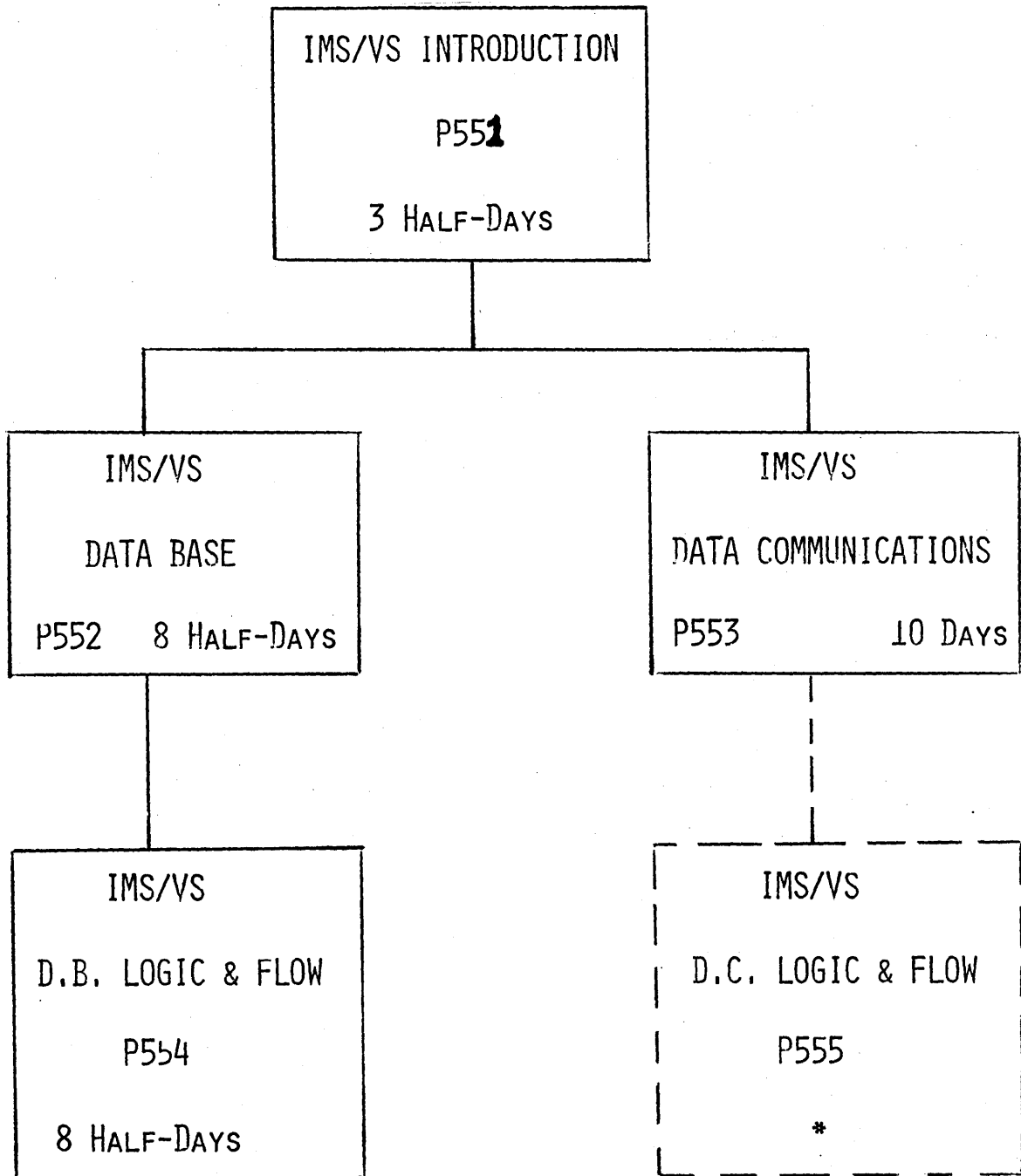


IMS / VS INTRODUCTION

TOPICS:

- IMS/VS TRAINING PLAN AT -STL-
- CONCEPTS
- ARCHITECTURE
- FACILITIES
 - D.B.
 - UTILITIES
 - D.C.
- INTEGRITY
- SYSTEM DEFINITION

STL PROGRAMMING EDUCATION
IMS/VS TRAINING PLAN



* COURSE UNDER DEVELOPMENT

Data Base System

DEFINITION

FROM THE DATA PROCESSING POINT OF VIEW

A DATA BASE IS A NON-REDUNDANT COLLECTION OF INTERRELATED DATA ITEMS PROCESSABLE BY ONE OR MORE APPLICATIONS BY THE INTEGRATION AND SHARING OF COMMON DATA

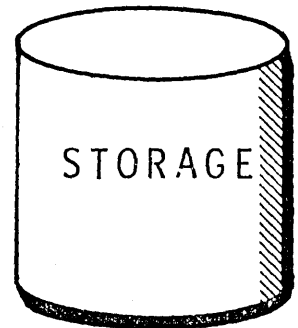
FROM THE USER POINT OF VIEW

A COLLECTION OF EXISTING AND PLANNED DATA AVAILABLE TO THE USER AS IF IT WERE A SINGLE INTEGRATED DATA BASE REGARDLESS OF ACTUAL PHYSICAL DESCRIPTION

DB/DC System

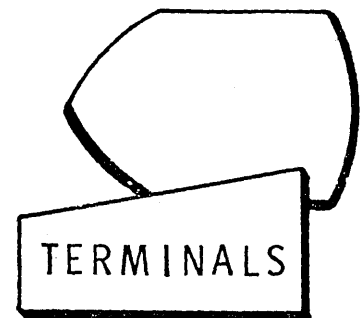
DATA **B**ASE:

MANAGES DATA
CENTRALLY FOR
MANY APPLICATIONS



DATA **C**OMMUNICATIONS:

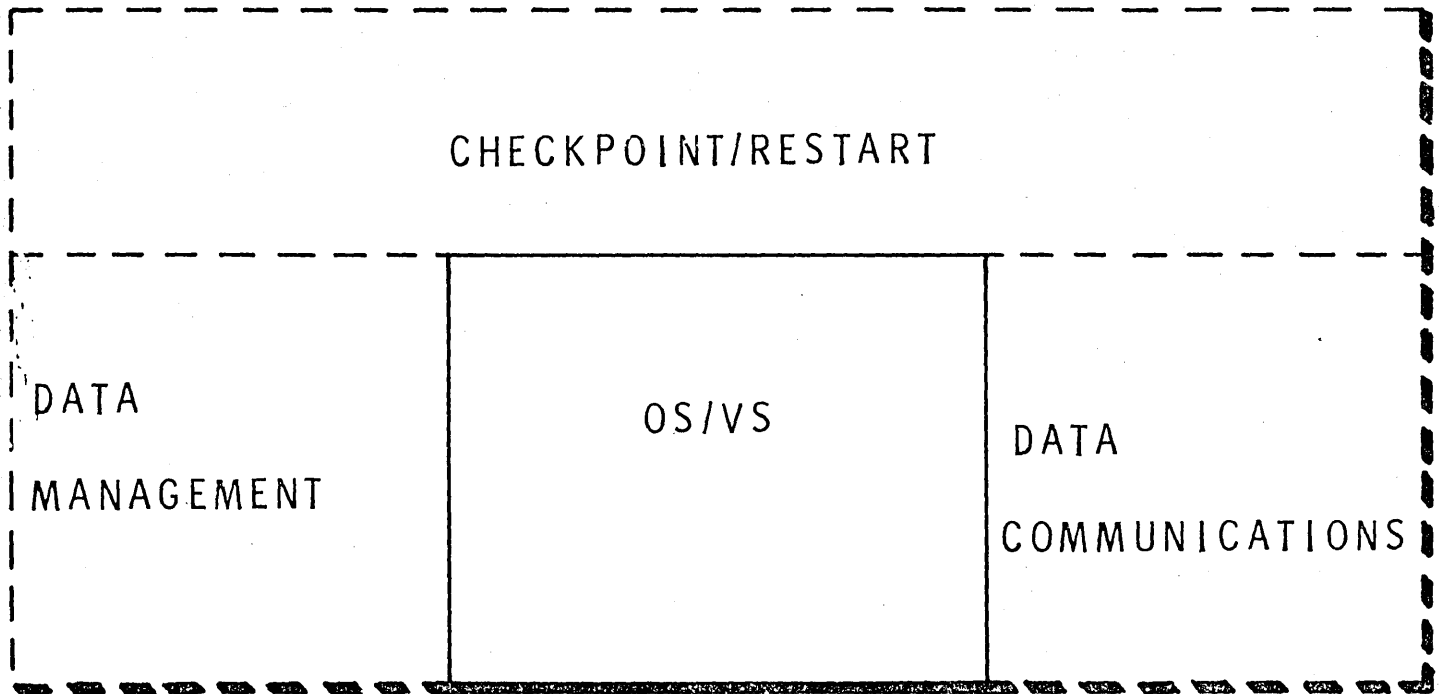
MAKES DATA
AVAILABLE TO
TERMINAL USERS



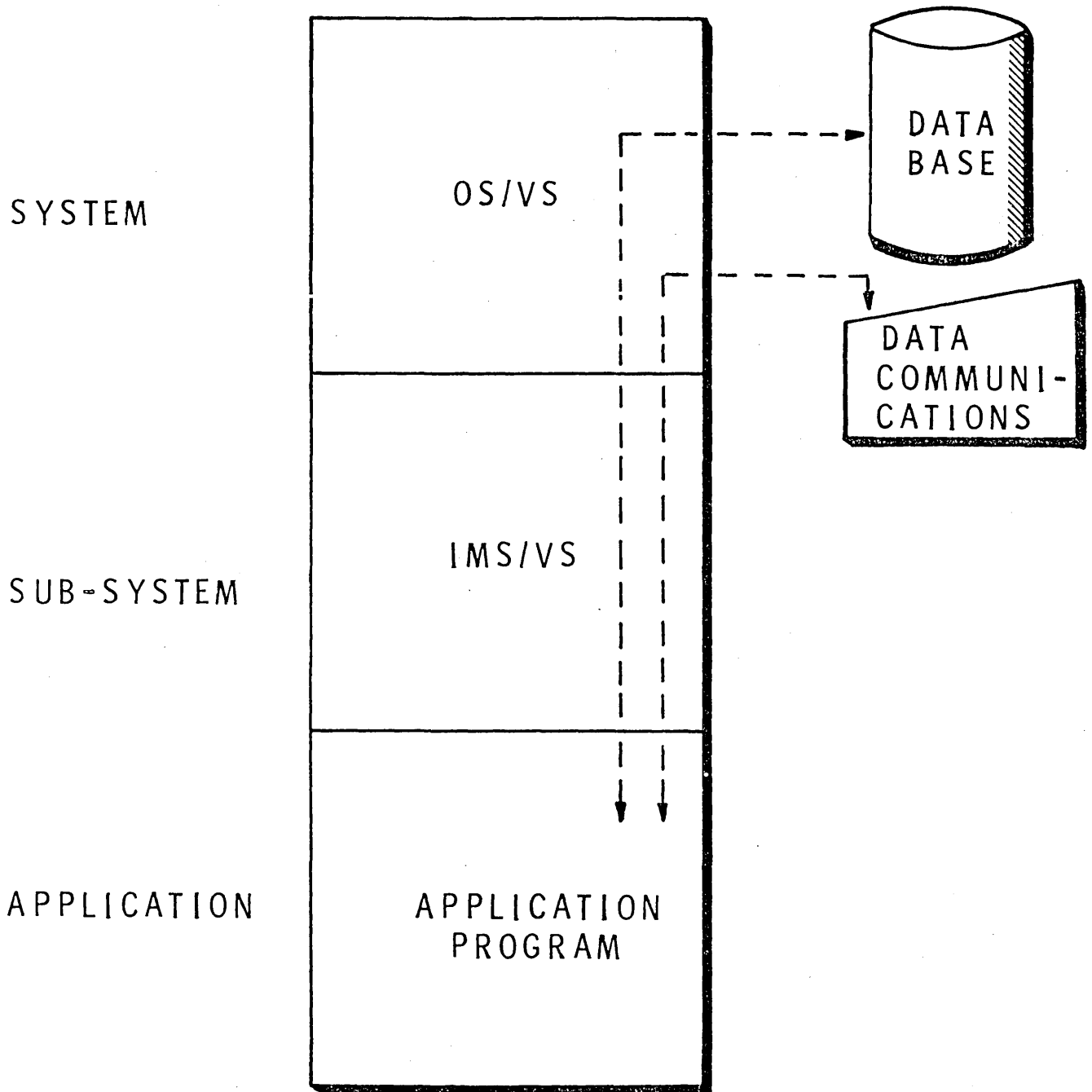
I M S / V S A R C H I T E C T U R E

- ① APPLICATION PROGRAM INDEPENDENCE FROM:
DATA
SPECIFIC TERMINAL(S)
- ② PROGRAM & DATA BASE INTEGRITY
- ③ COMPLETE DATA BASE MANAGEMENT
- ④ RECOVERY AS A SYSTEM FUNCTION
- ⑤ LOGICAL & PHYSICAL DATA STRUCTURES
- ⑥ LOGICAL & PHYSICAL TERMINALS
- ⑦ CONCURRENT PROCESSING / CENTRALIZED FILES
- ⑧ EXTENSIVE RECOVERY
- ⑨ RESTART
- ⑩ LARGE VOLUME OF DATA / FLEXIBLE STRUCTURES

An Extension Of OS/VS



IMS/VS Is A Sub-System

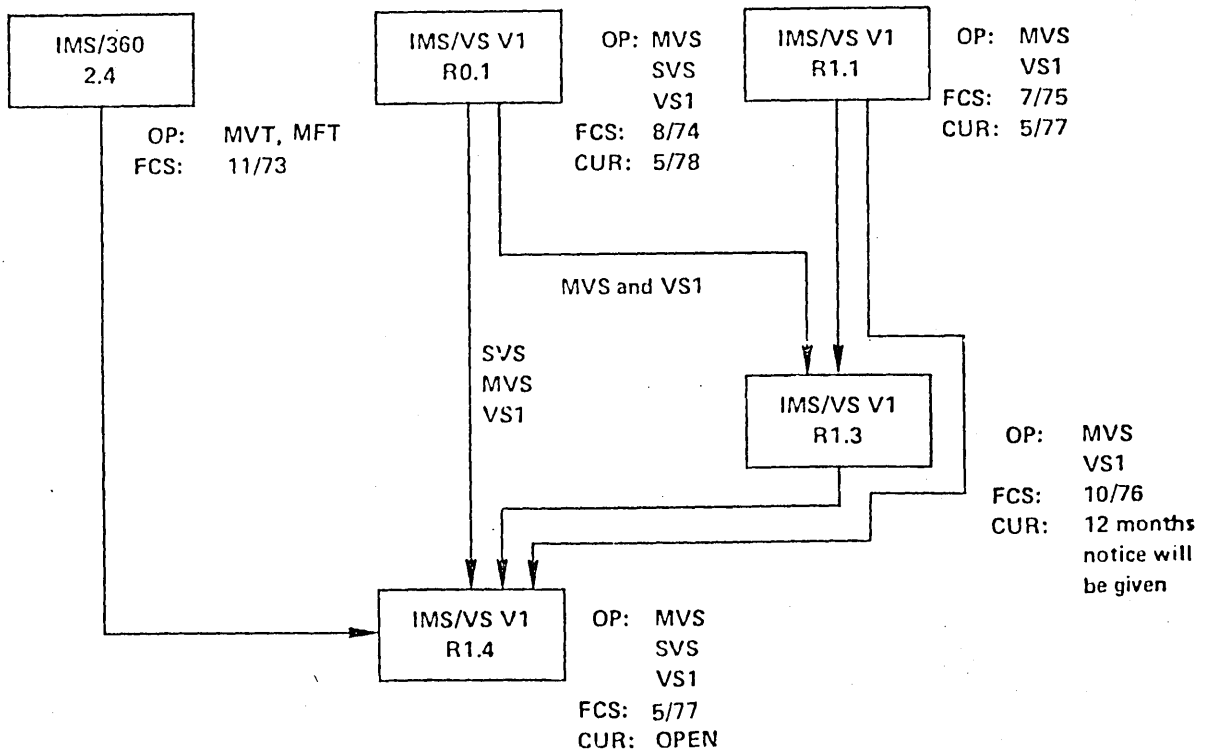


IMS/VS Operating Environment

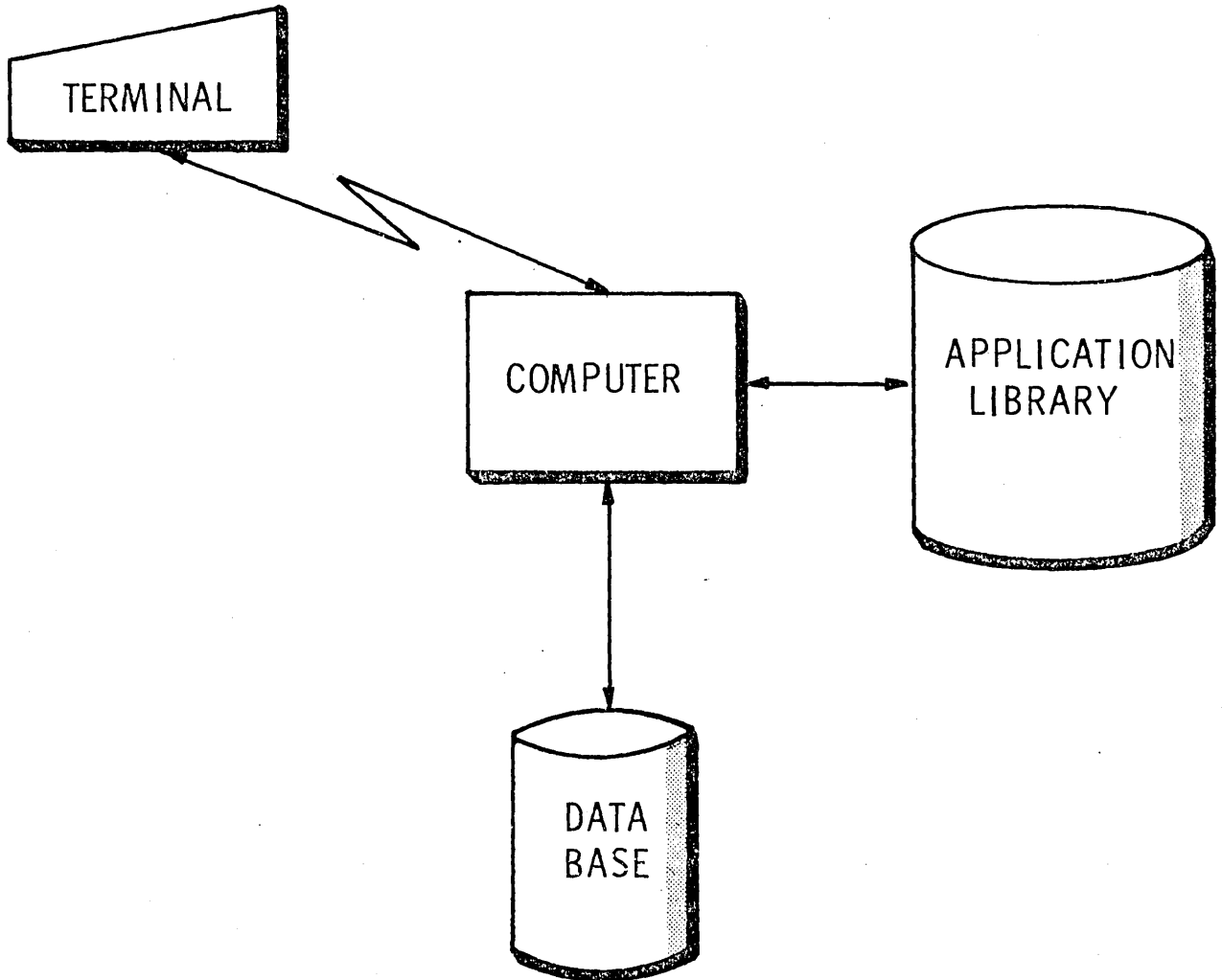
RELEASE - 1.1.4

SUPPORT OS/VS - 1 REL 6
OS/VS - 2 REL 1.7 (SVS)
OS/VS - 2 REL 3.7 (MVS)

MIGRATION



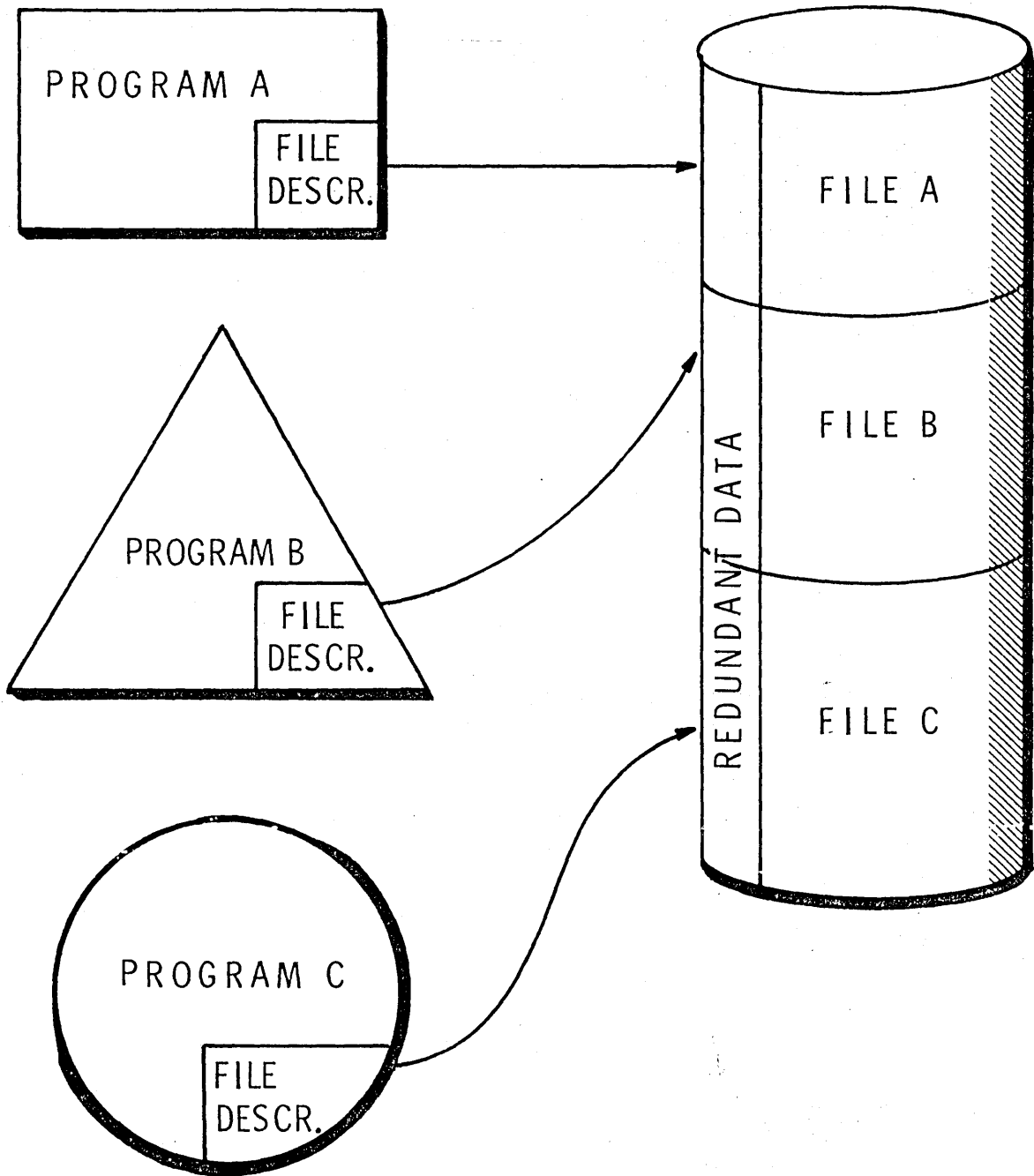
Major Facilities



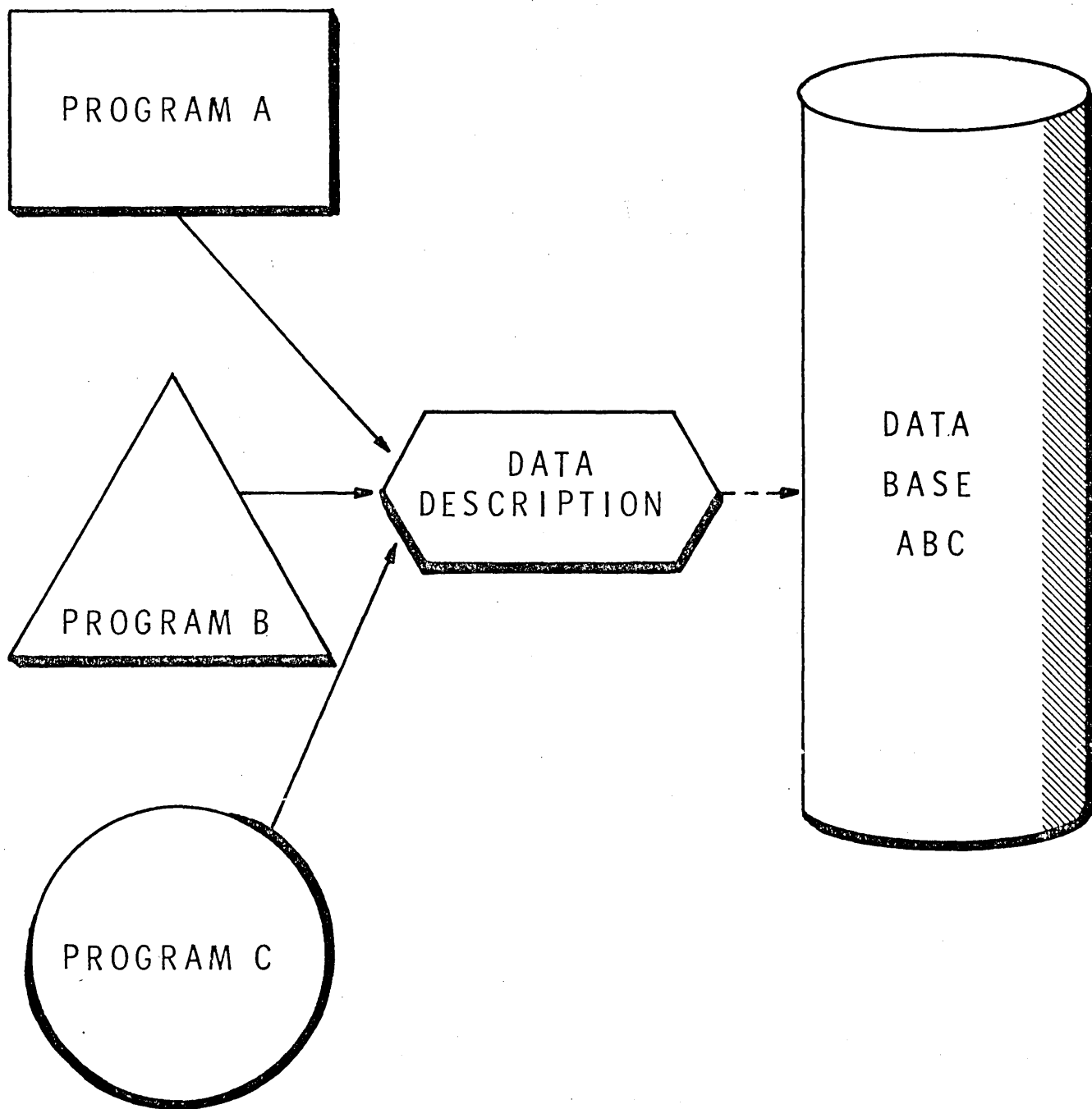
DATA BASE FACILITY

DATA COMMUNICATIONS FACILITY

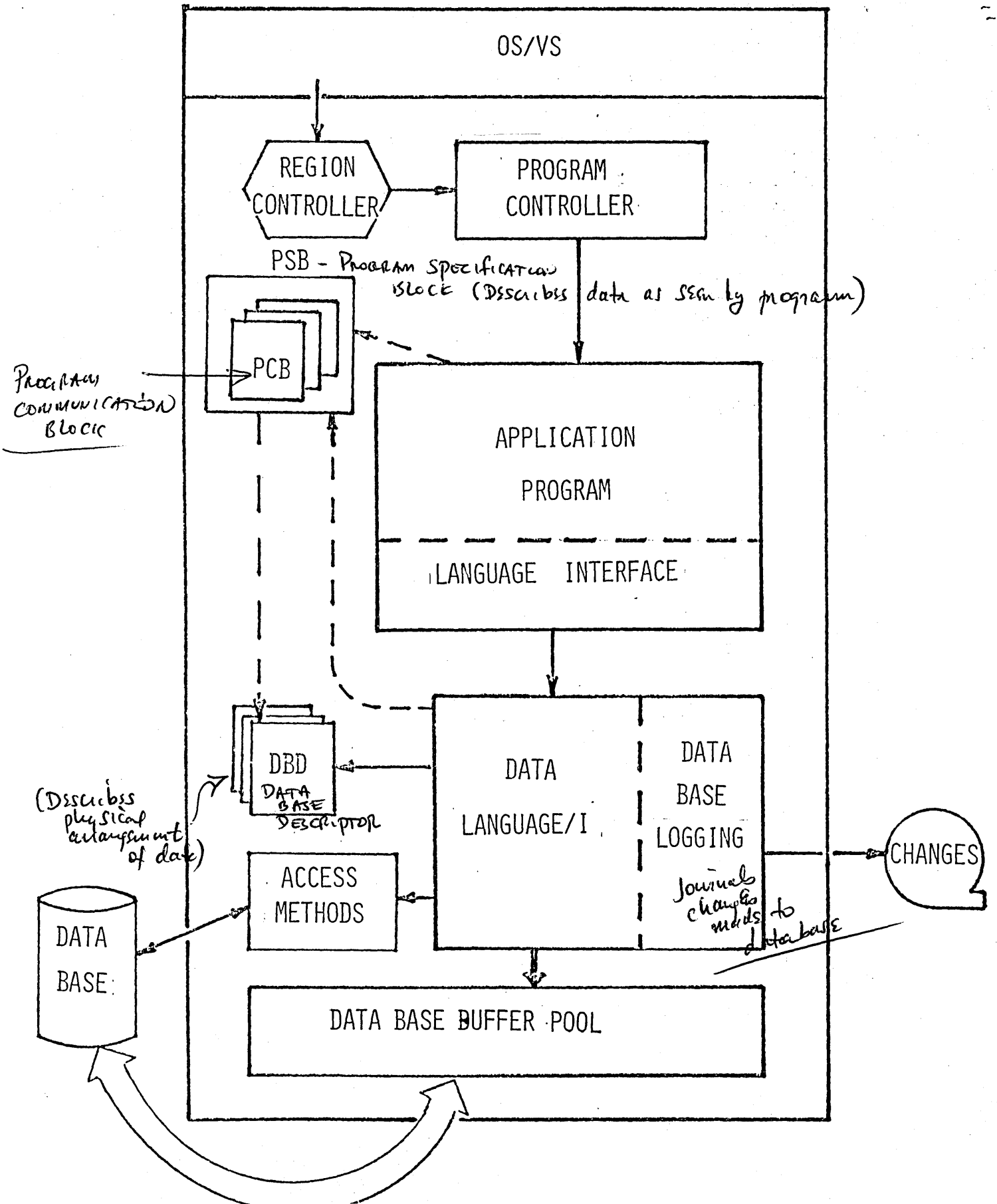
Traditional Approach



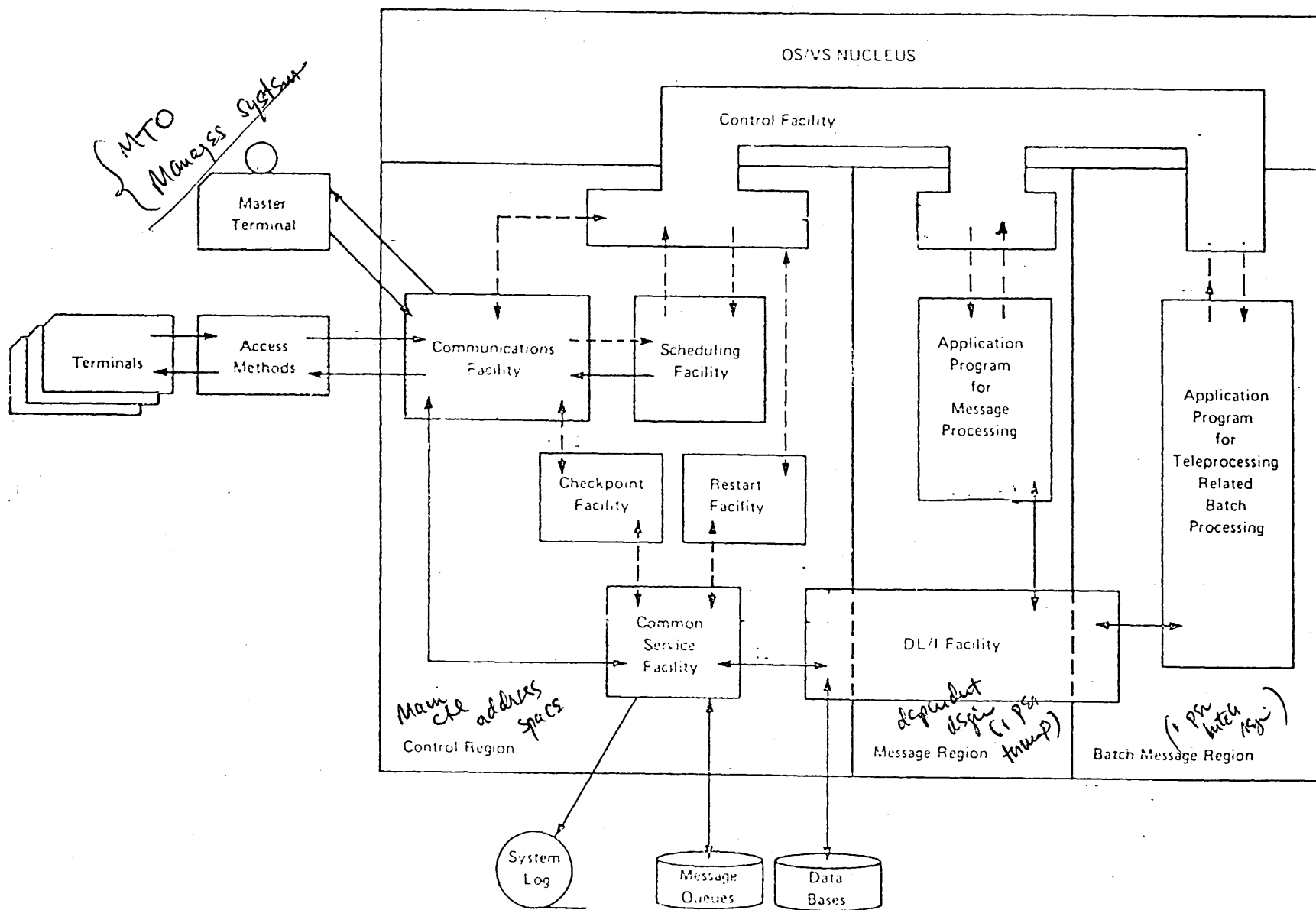
Data Base Approach

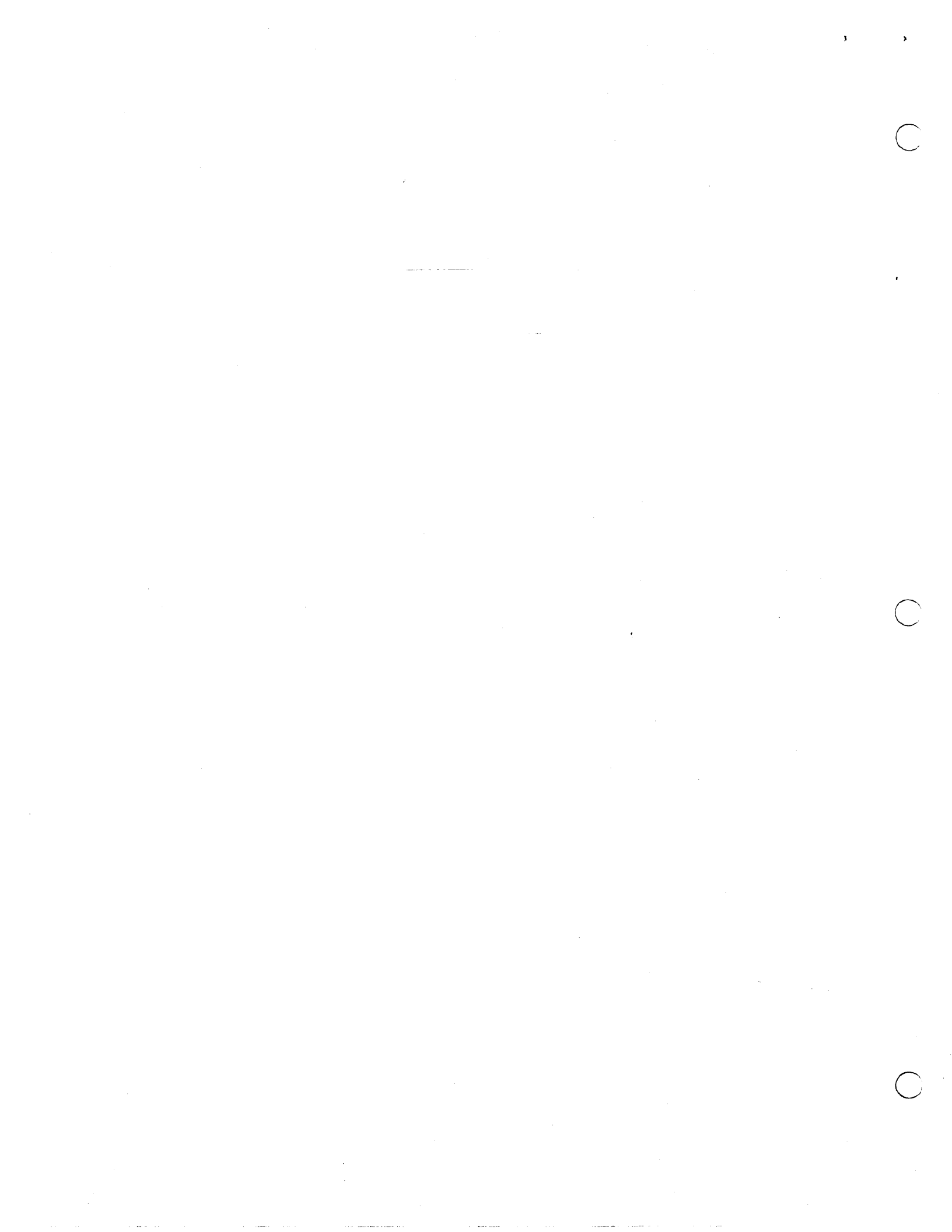


BATCH PROCESSING REGION

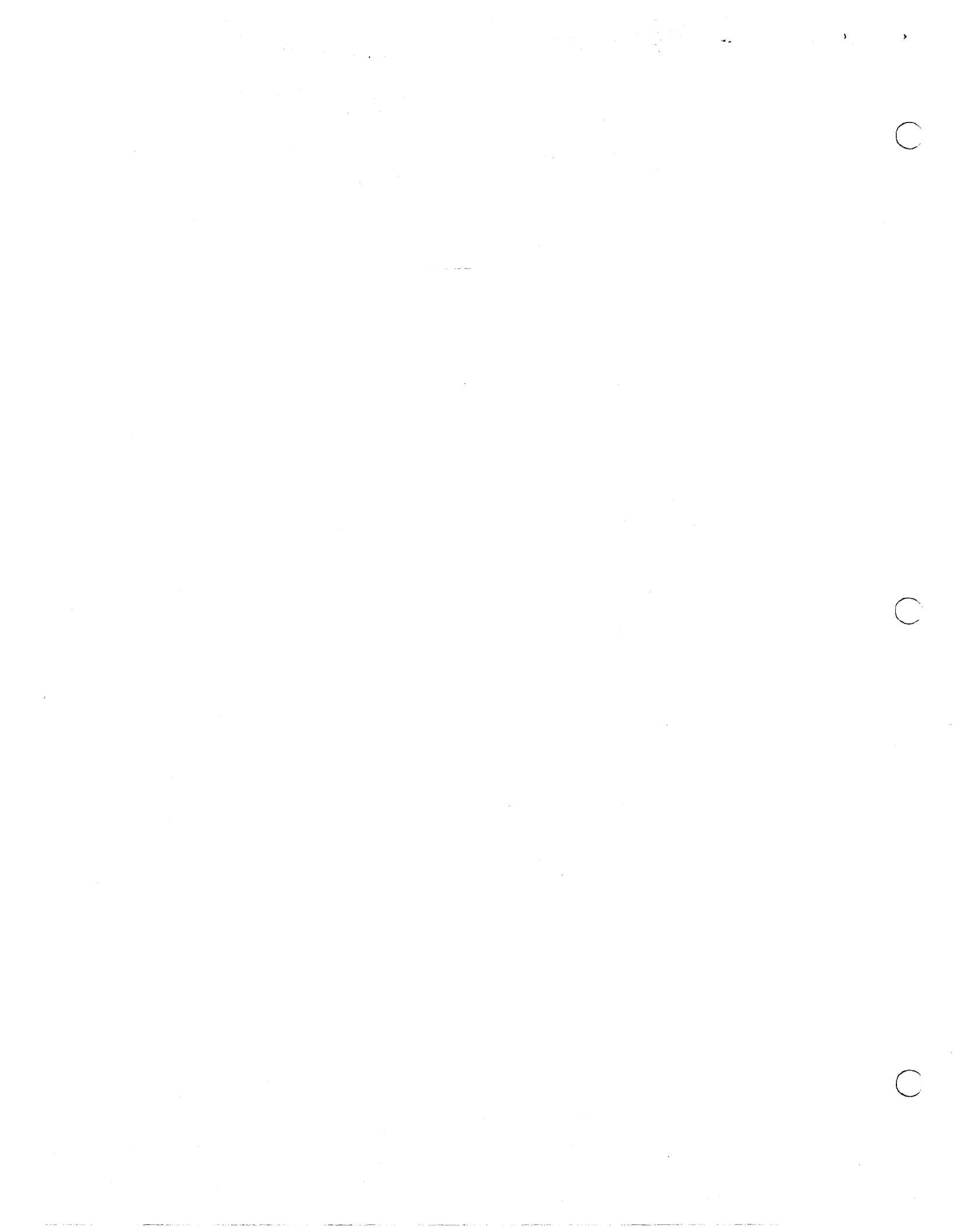


IMS/VS D.C. SYSTEM OVERVIEW





DB facilities



BASIC FEATURES OF 'DATA LANGUAGE/I' (DL/I)

1. DATA INDEPENDENCE -
SEPARATION OF THE LOGICAL DESCRIPTION
OF DATA FROM PHYSICAL DEVICES & ACCESS
2. GROWTH OF DATA BASES -
BOTH VOLUME AND STRUCTURE
3. HIGH LEVEL LANGUAGE SUPPORT - PL/I & COBOL
4. UTILITIES TO -
DEFINE DATA STRUCTURES
RELATE STRUCTURES TO APPLICATION PGMS.
LOAD STRUCTURES
REORGANIZE STRUCTURES
RECOVER STRUCTURES
5. DATA SECURITY AND INTEGRITY -
SEGMENT SENSITIVITY
PROCESSING INTENT
LOGGING AND RECOVERY FACILITIES

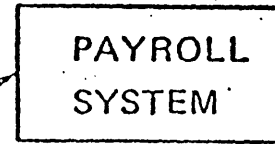
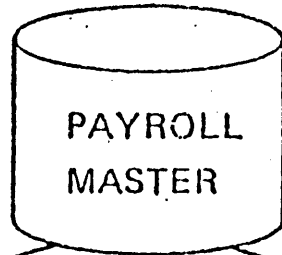
Traditional File Structure

BANK CUSTOMER	DEMAND DEPOSIT	STATEMENT INFO.	LOAN	PAYMENT HISTORY	ADDRESS	TRUST	TRUST BALANCES	TRUST TRANS. HISTORY
------------------	-------------------	--------------------	------	--------------------	---------	-------	-------------------	----------------------------

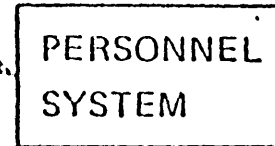
EMPLOYEE	SECURITY	AUTHORIZATION	EDUCATION	EXPERIENCE	PREVIOUS POSITION	PAYROLL	CREDIT UNION	BANK DEPOSIT
----------	----------	---------------	-----------	------------	----------------------	---------	-----------------	-----------------

ADD NEW APPLICATION

EXPAND PAYROLL MASTER ?



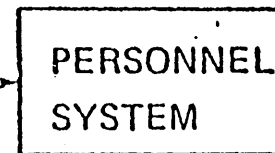
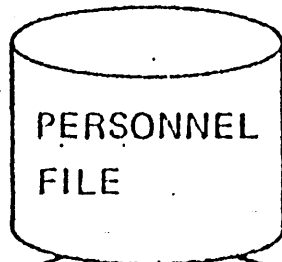
15 PROGRAMS



EMP NO.	NAME	ADDR	DEPT	NO. DEP	YTD SALARY	YTD FIT	YTD FICA	YTD STATE	PERSONNEL INFO
---------	------	------	------	---------	------------	---------	----------	-----------	----------------

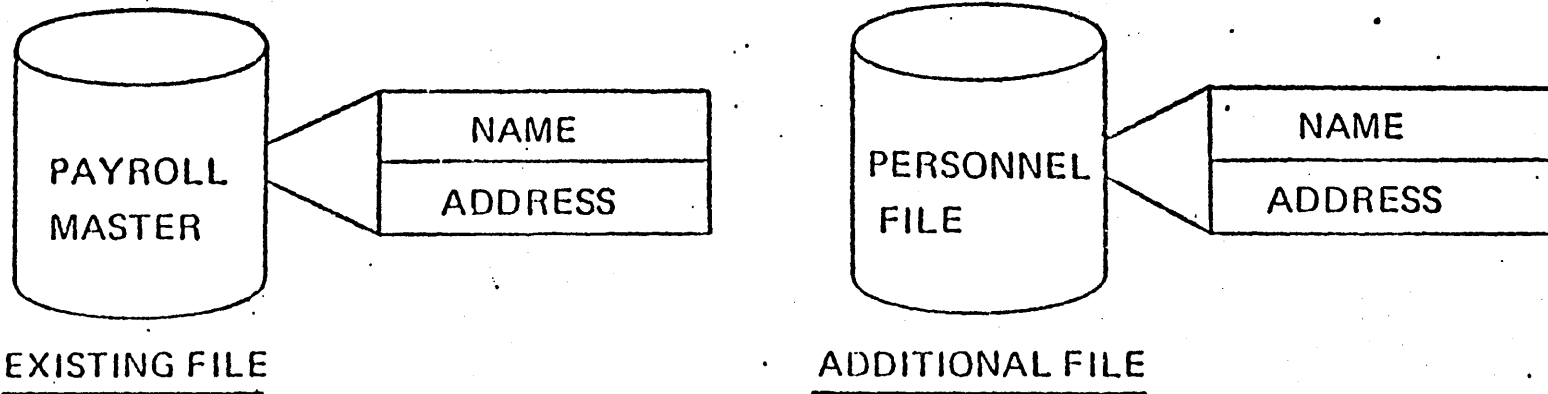
OR

CREATE NEW FILE ?



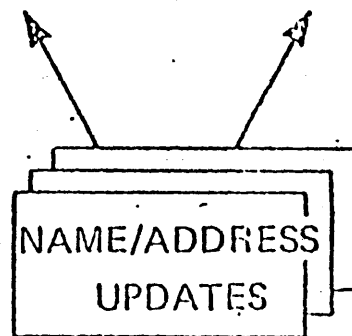
NAME	EMP NO.	ADDR	DEPT	JOB CODE	SKILL CODE	EDUC.	EXPR.
------	---------	------	------	----------	------------	-------	-------

PROBLEMS OF REDUNDANT DATA



• **PROBLEM:**

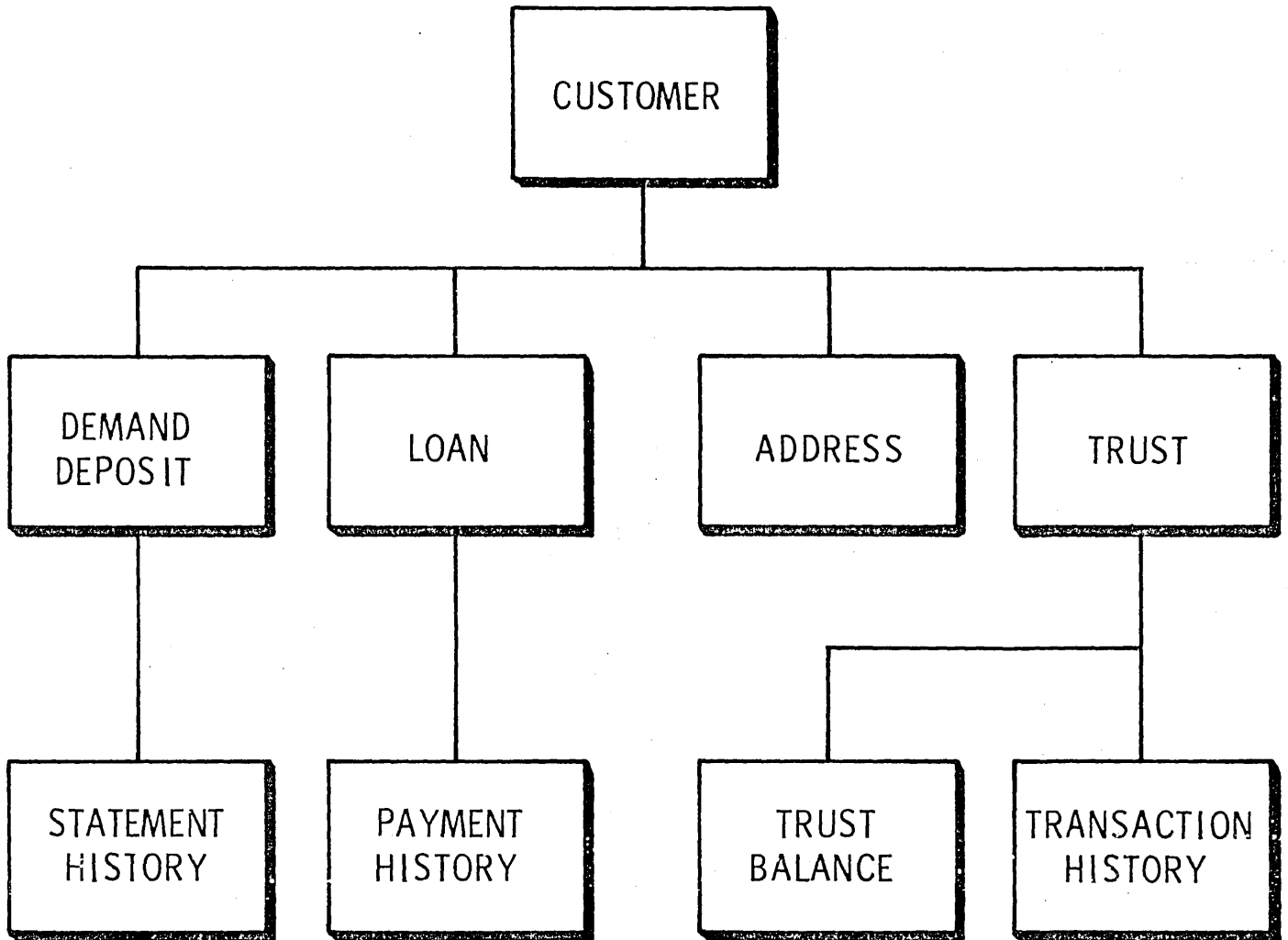
DATA GETS OUT OF SYNCH



UPDATES TO ONE FILE CREATE INCONSISTENCIES BETWEEN FILES

(Hierarchical)

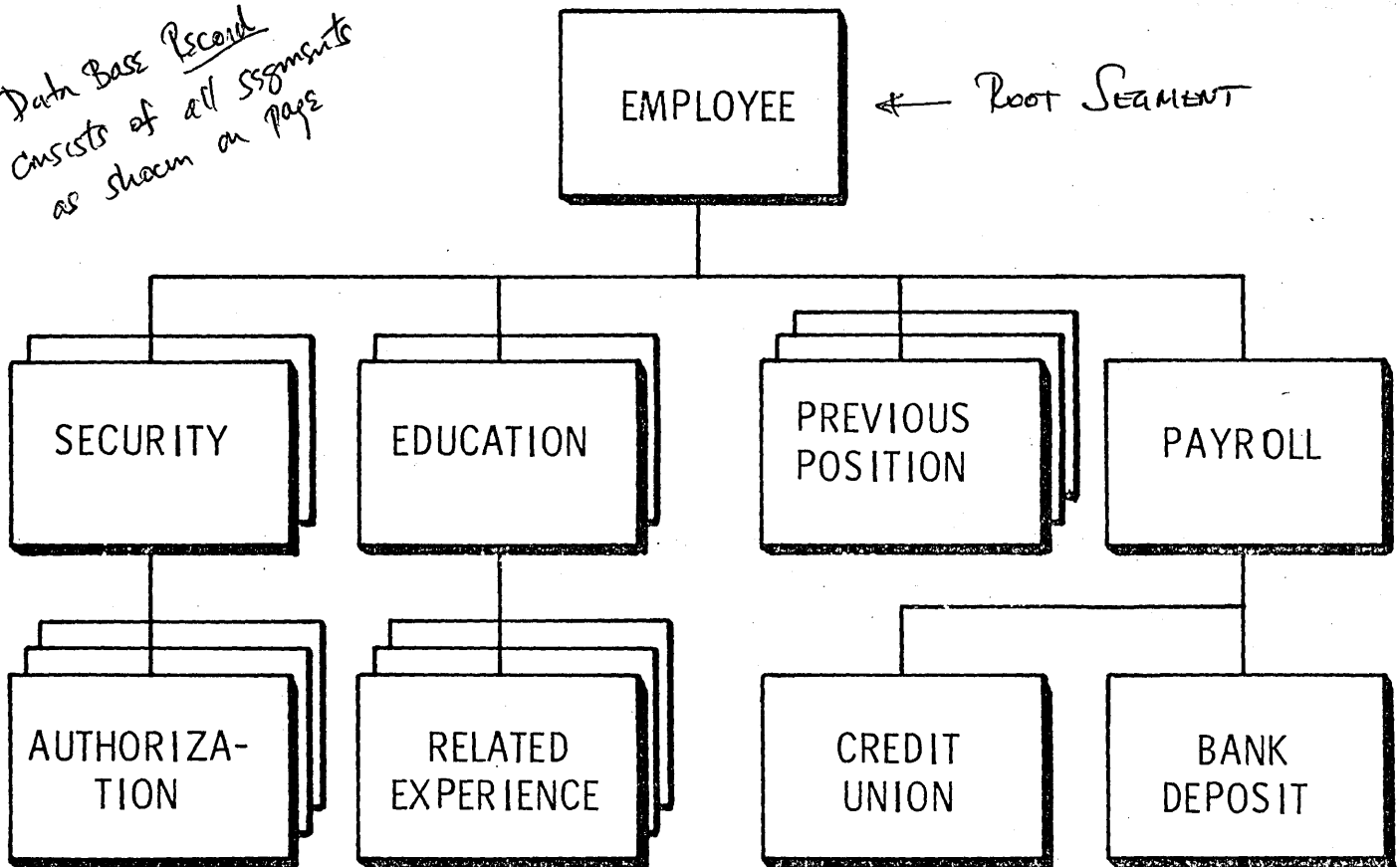
Logical Data Structure



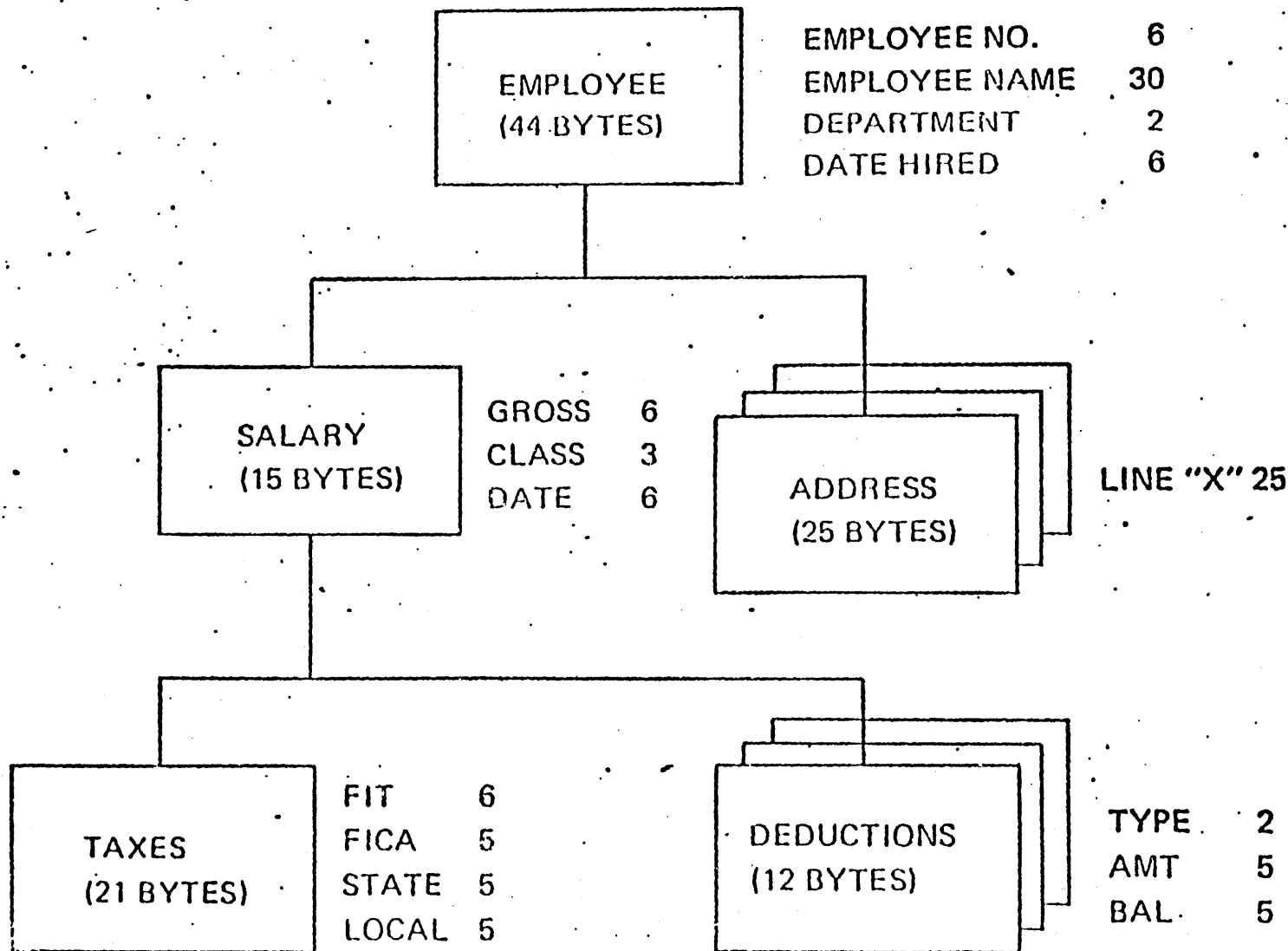
Logical Data Structure

- A DATA BASE CONSISTS OF 1 TO N DATA BASE RECORDS
- A DATA BASE RECORD CONSISTS OF 1 TO N SEGMENTS
- MAXIMUM OF 255 SEGMENT NAMES
- MAXIMUM OF 15 SEGMENT LEVELS < *Actually more, but documented, not yet changed*
- 1 ROOT SEGMENT PER DATA BASE RECORD
- DEPENDENT SEGMENTS -- 0 TO N PER PARENT

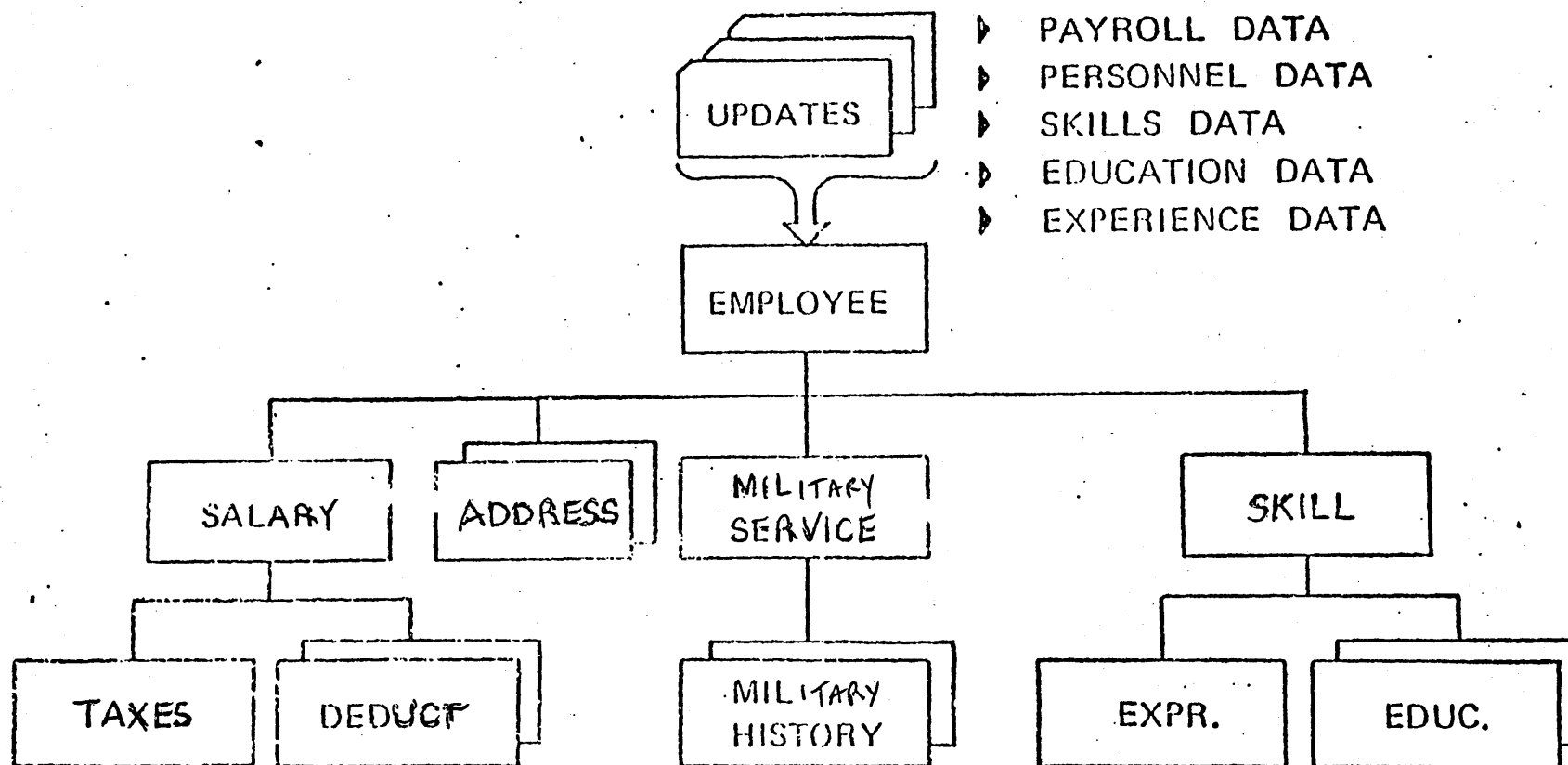
*Data Base Record
consists of all segments
as shown on page*



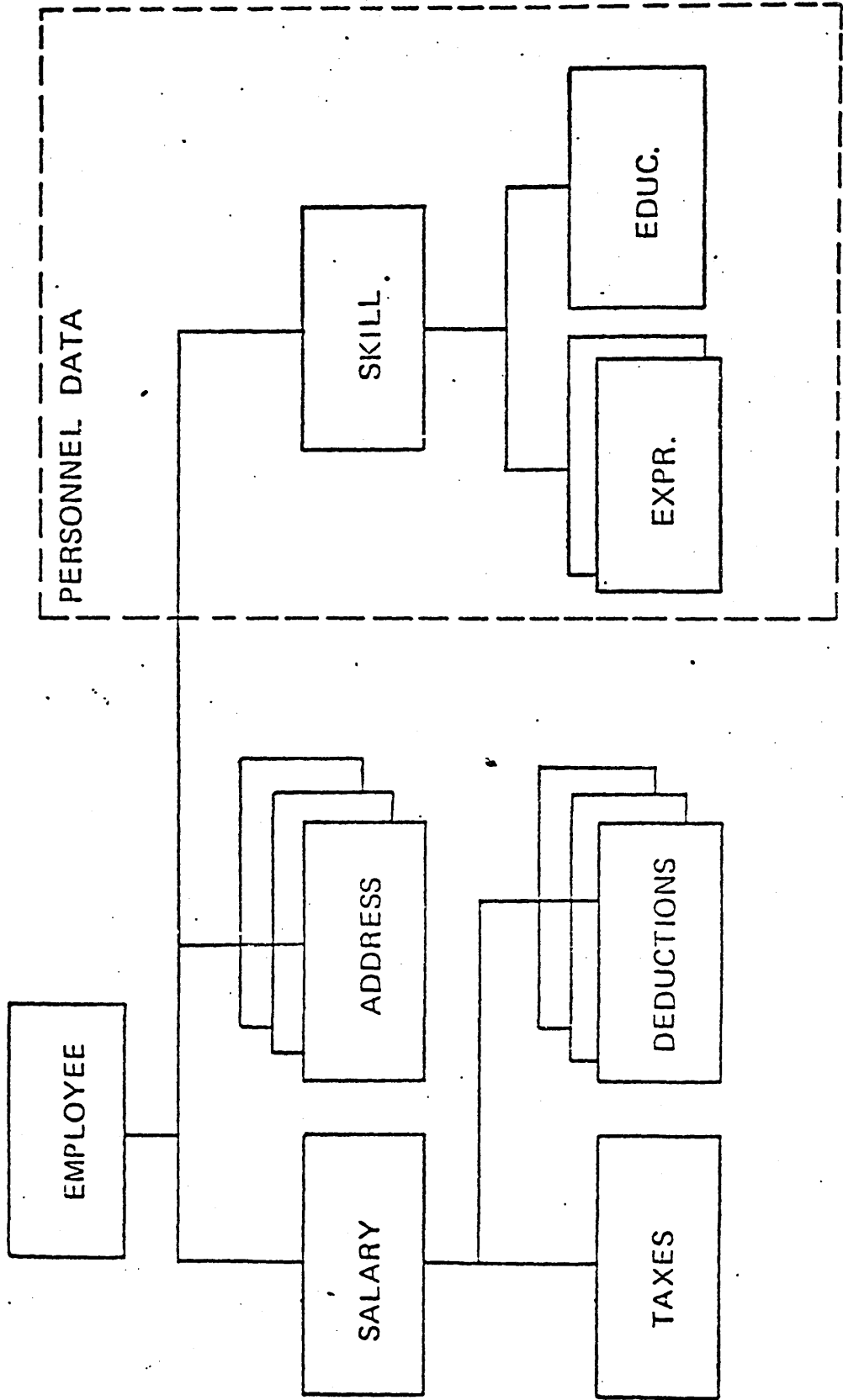
DATA HIERARCHY



UPDATES TO A MULTI-APPLICATION DATA BASE

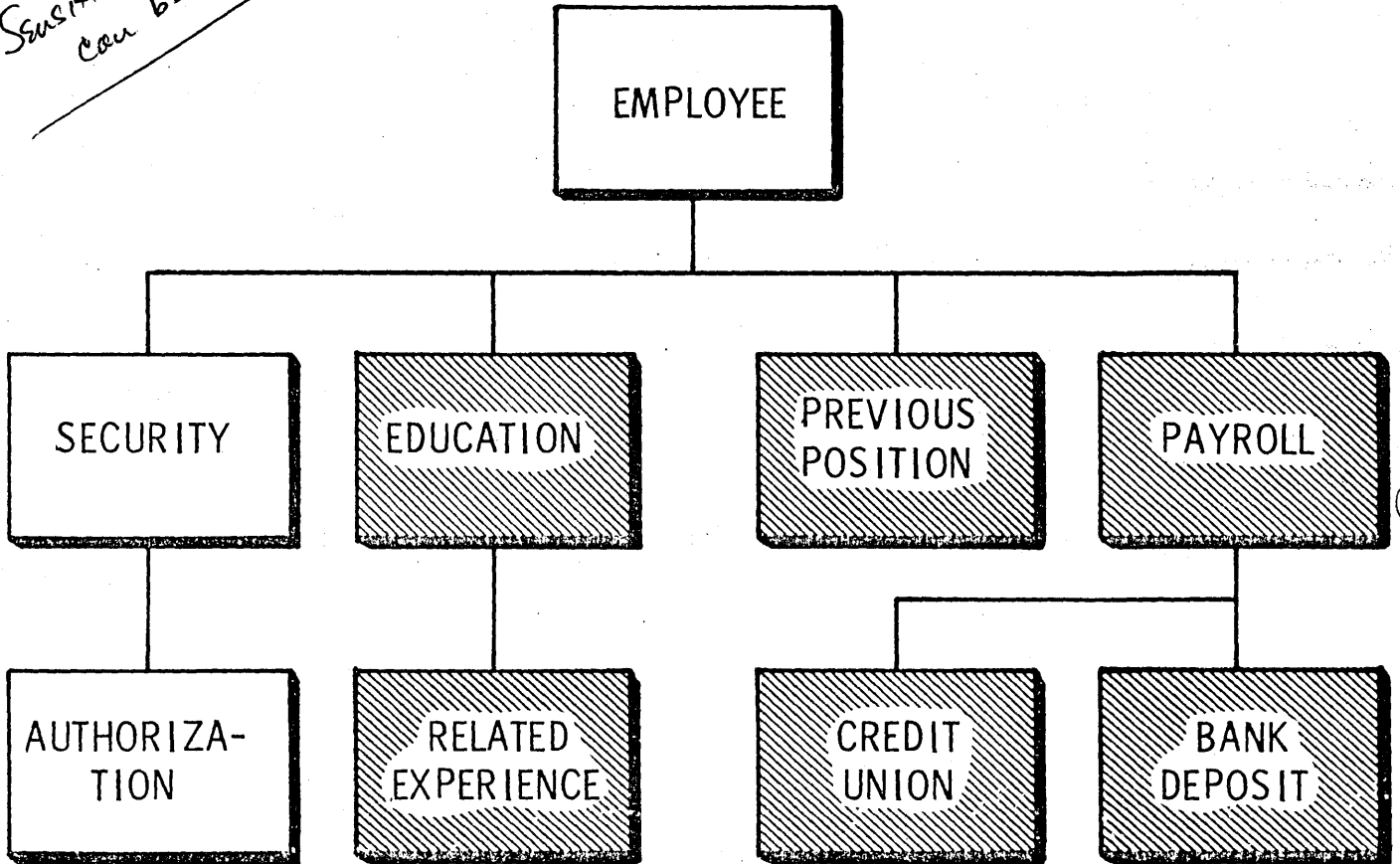


DL/I EASY TO EXPAND

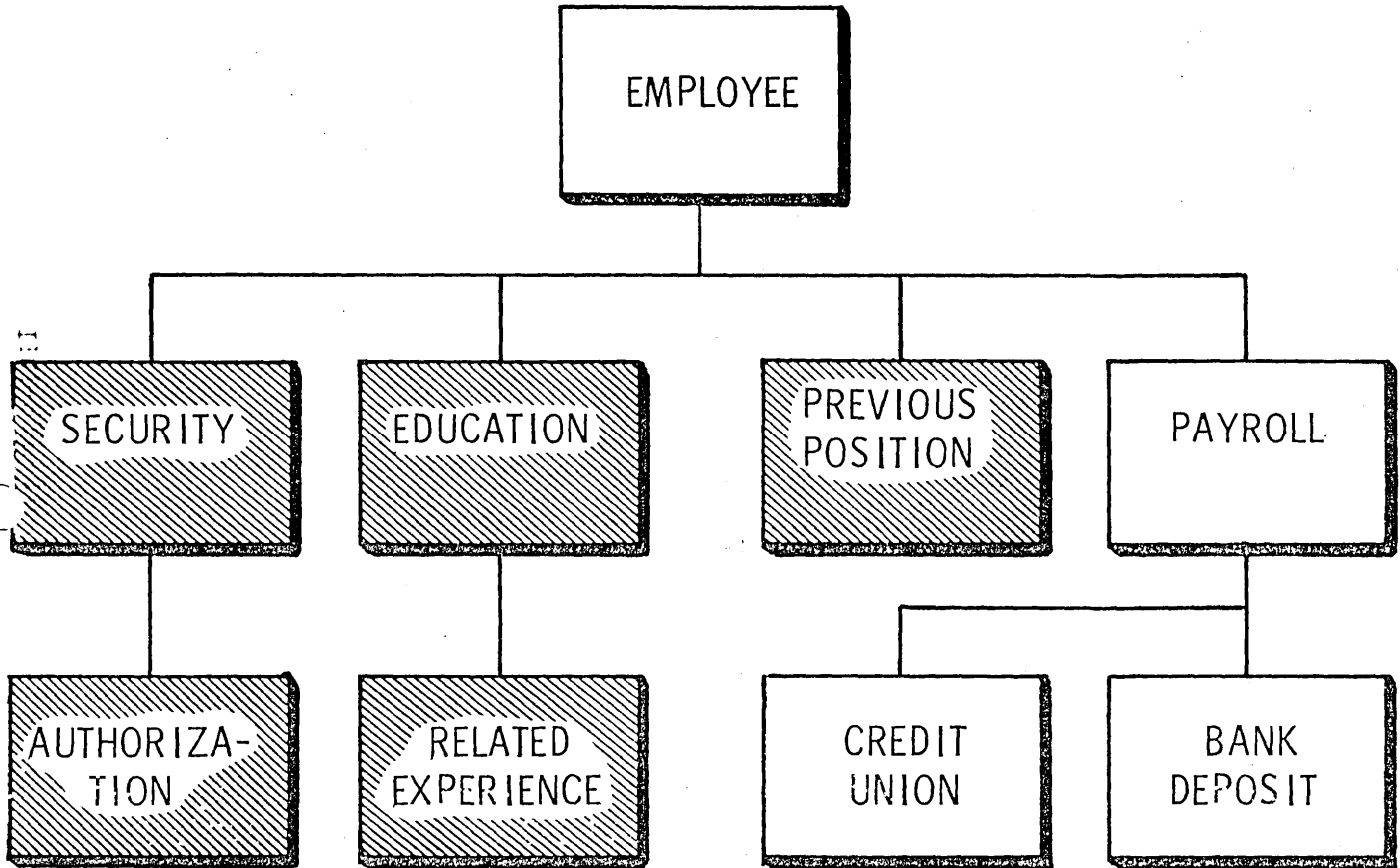


Sensitivity Of User One

Sensitivity = what can be seen



Sensitivity Of User Two



Data Base Operation

- DATA BASE DESCRIPTION
- PROGRAM DESCRIPTION
- DATA BASE CREATION
- DATA BASE PROCESSING

Data Base Description (DBD)

Done by Data Base Administrators

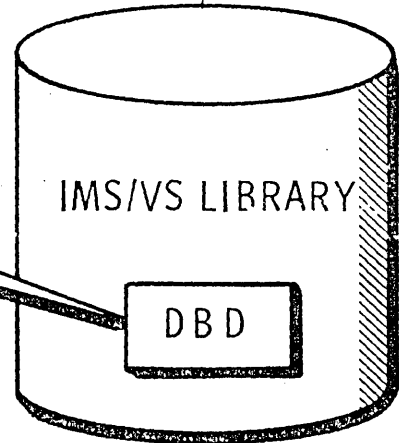
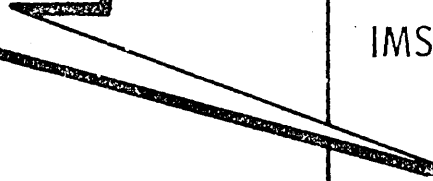
CONTROL
CARDS



DBD
GENERATION
UTILITY



DATA BASE NAME
SEGMENT NAMES AND LENGTHS
FIELD NAMES AND LENGTHS
ACCESS METHOD



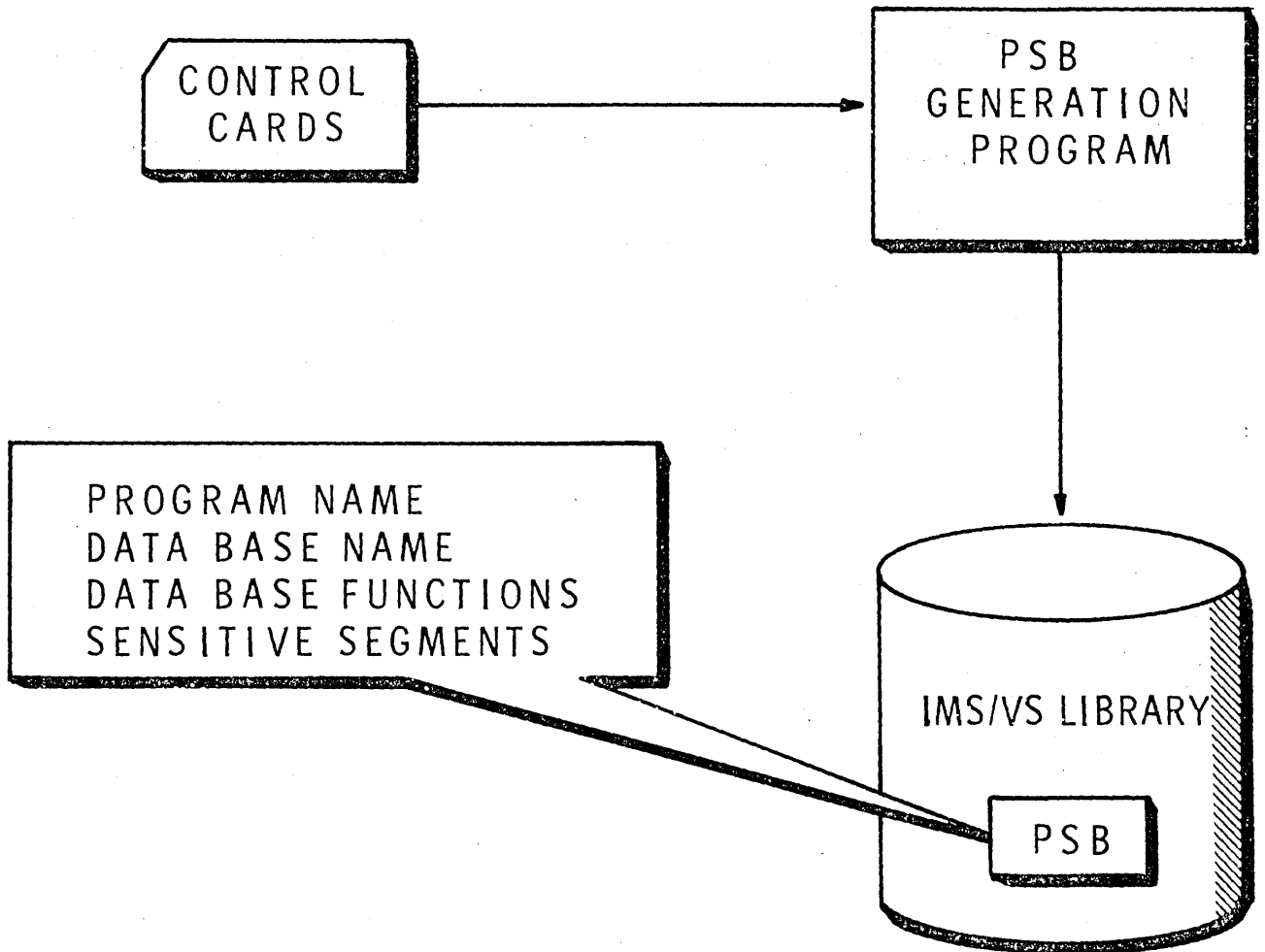
Program Communication
Block gives multiple sensitivities

Program Description

Program Specific
Block

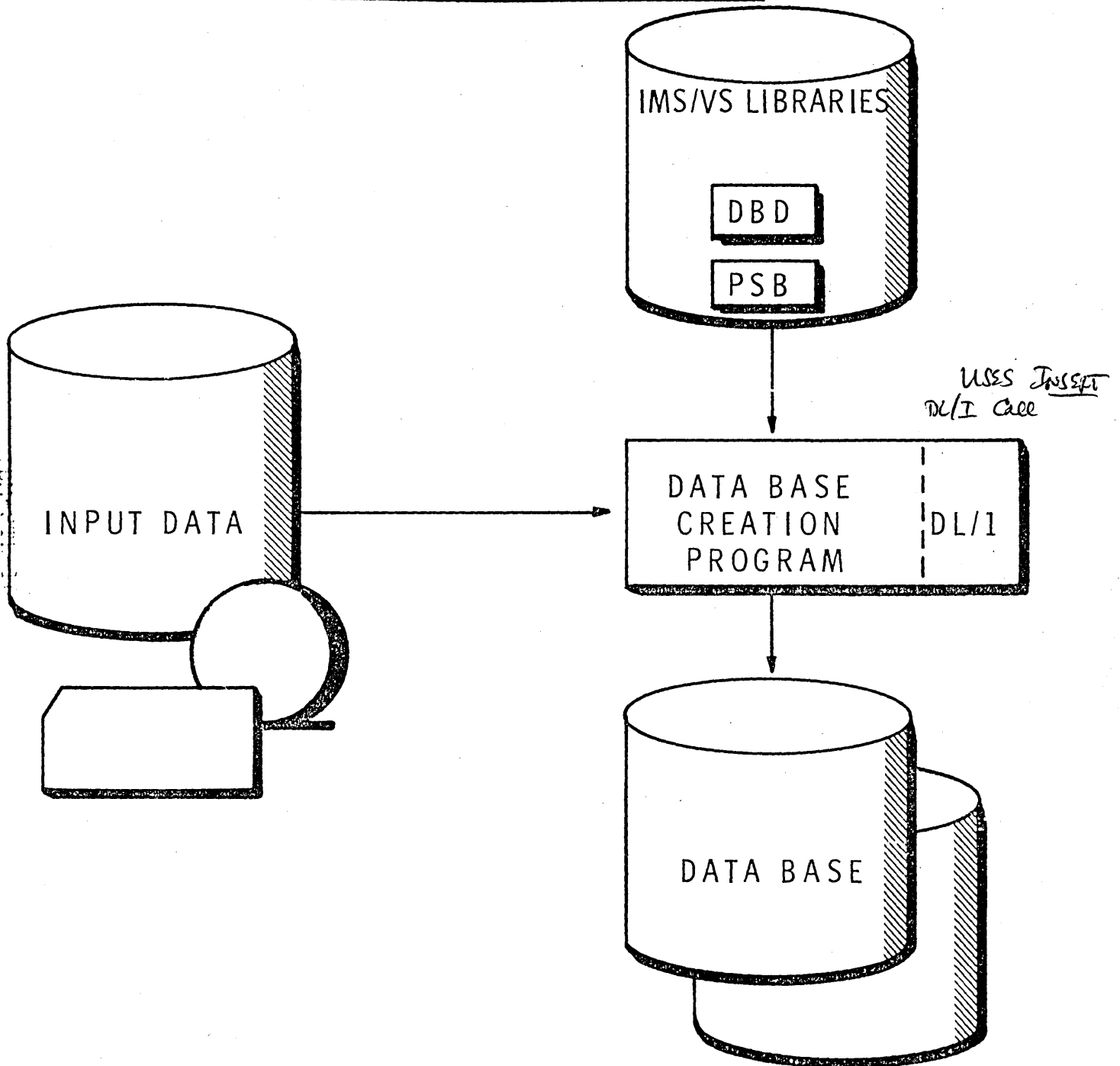
→ (PSB)

Done by Data
Base Administrator

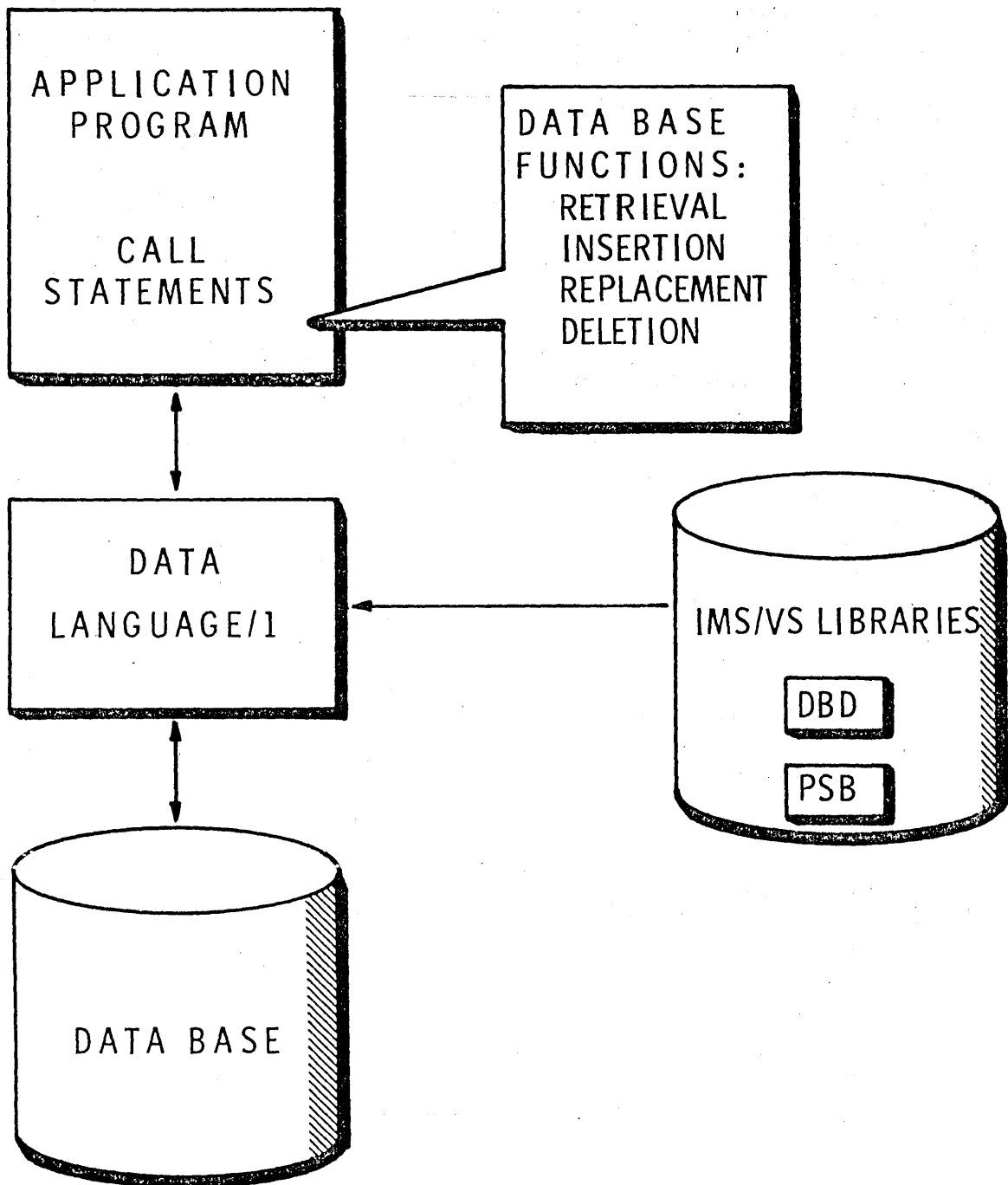


Multiple PSB ~~are~~ required on-line
if multiple message regions
want same view (e.g., 7-7
between Pgm Name & PSB name)

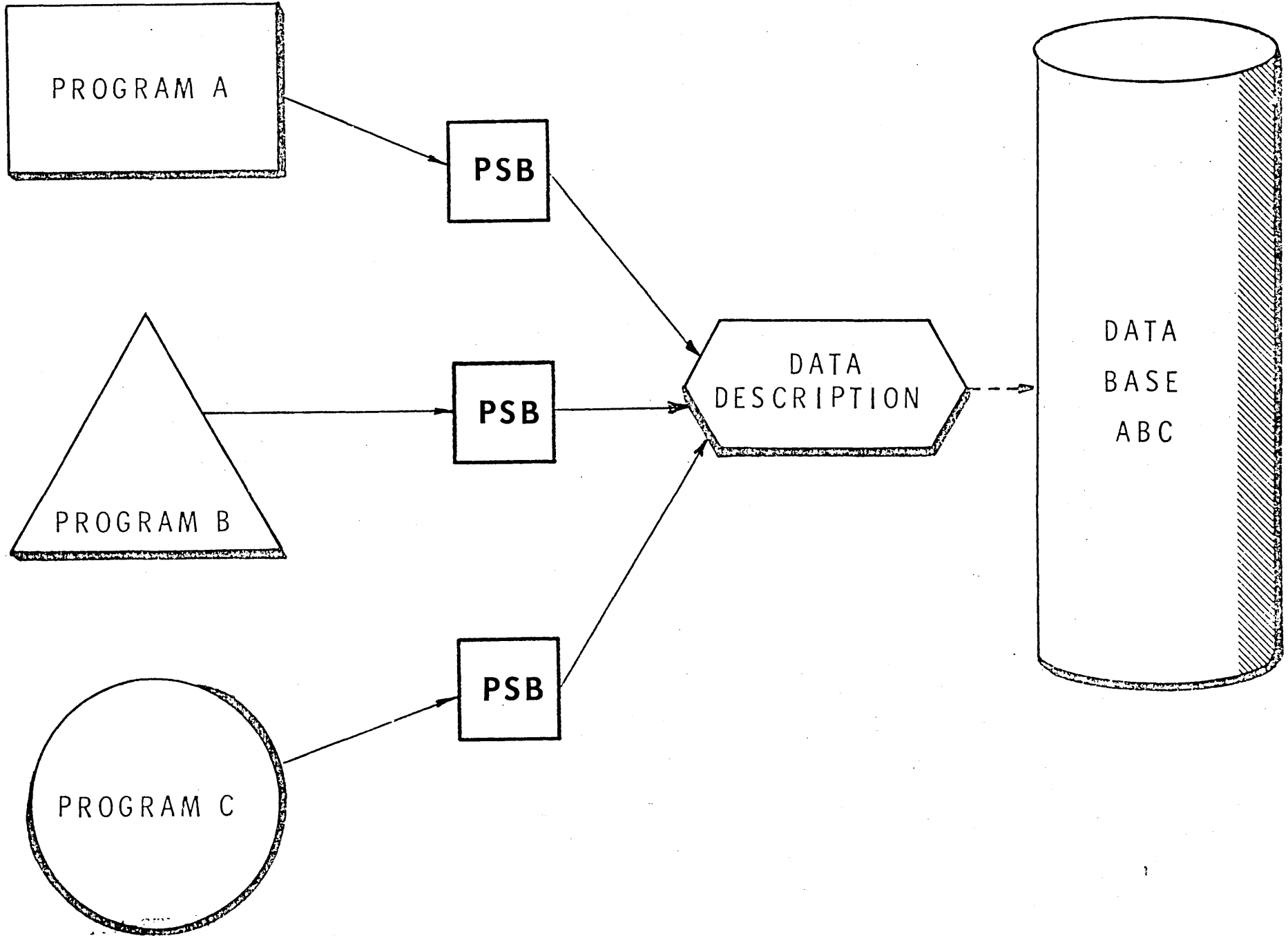
Data Base Creation



Data Base Processing

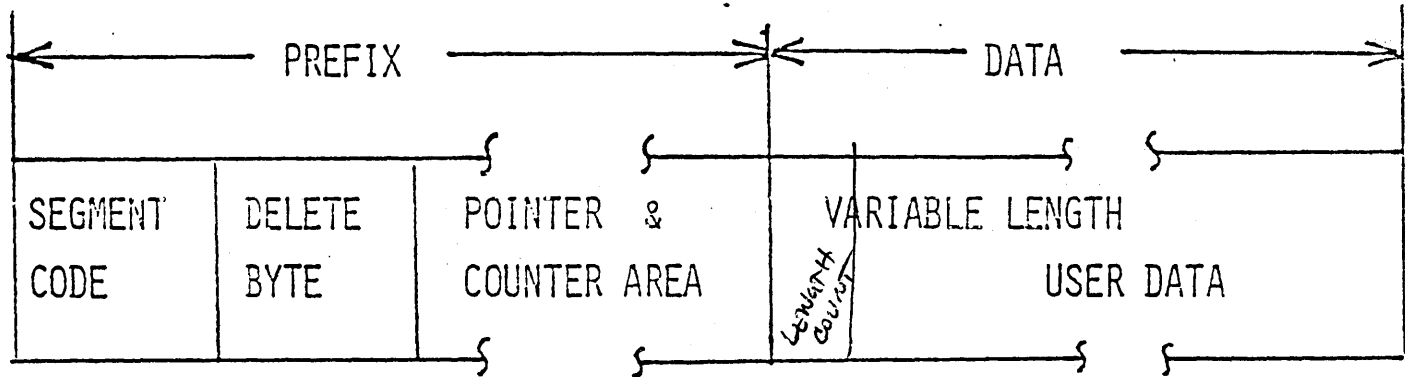
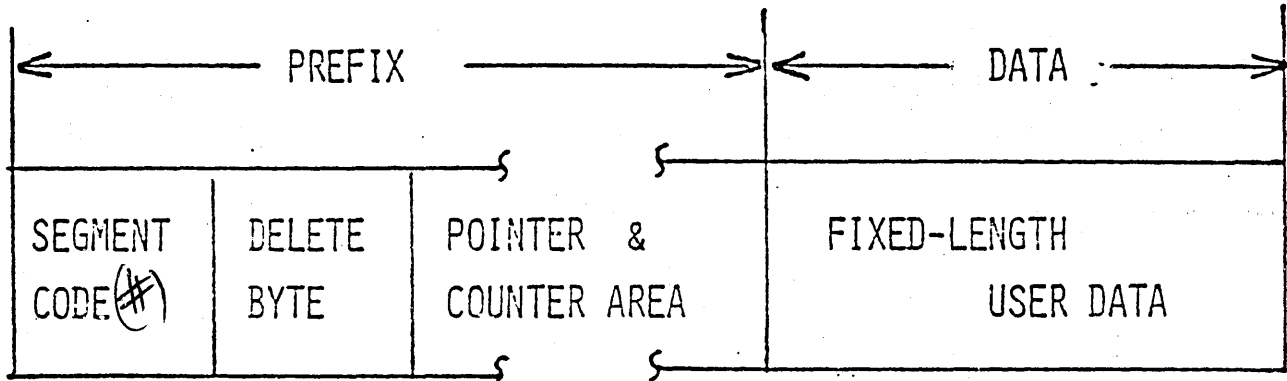


DATA INDEPENDENCE



SEGMENT

FORMATS

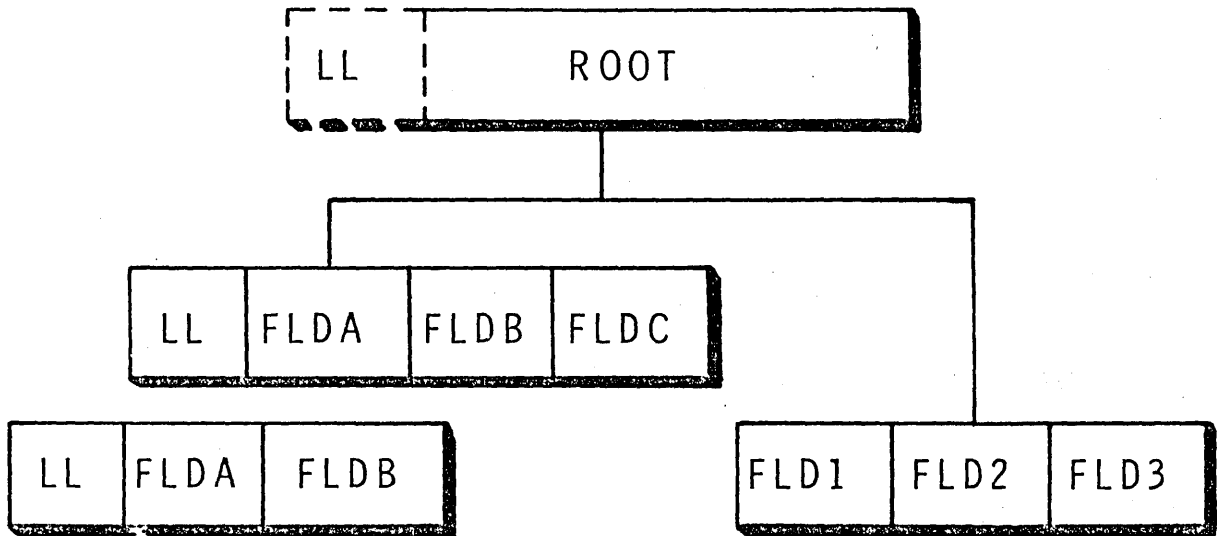


SEG. CODE: 1 - 255 SEG T/PES

POINTERS: FOR HD ORGS (AND HISAM IF SEG = LOGICALLY RELATED TO AN HD SEG)

COUNTERS: LOGICAL RELATIONSHIPS

Fixed And Variable Length Segments



FIXED

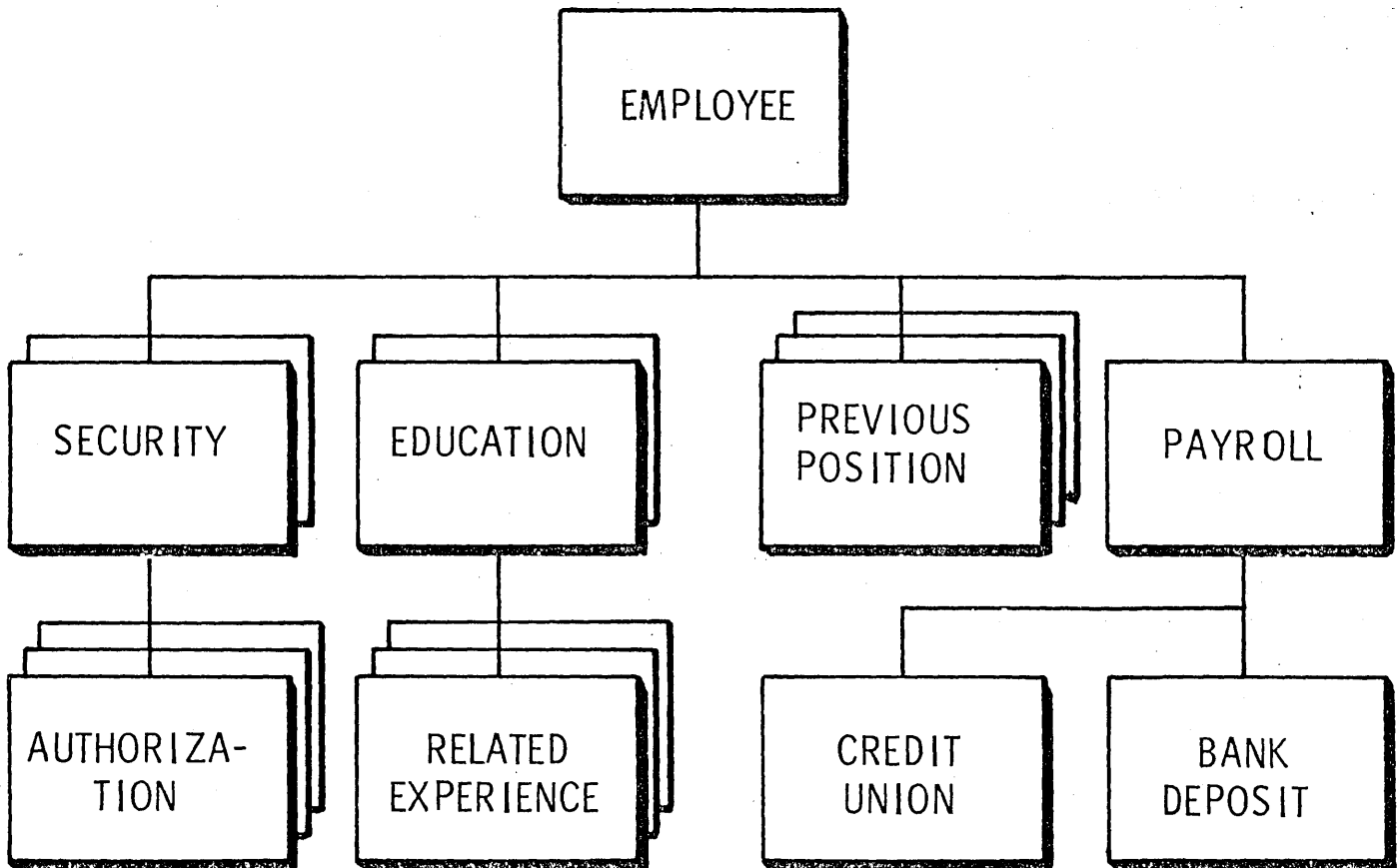
EFFICIENT MAINTENANCE PROCESSING AND
DASD BLOCKING

VARIABLE

EFFICIENT TEXT PROCESSING APPLICATIONS
MULTIPLE OCCURRENCES OF SAME FIELD OR ONE
VARIABLE LENGTH FIELD (I. E., DESCRIPTION)
LL FIELD MUST BE MAINTAINED BY USER

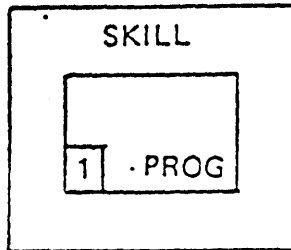
Logical Data Structure

- A DATA BASE CONSISTS OF 1 TO N DATA BASE RECORDS
- A DATA BASE RECORD CONSISTS OF 1 TO N SEGMENTS
- MAXIMUM OF 255 SEGMENT NAMES
- MAXIMUM OF 15 SEGMENT LEVELS
- 1 ROOT SEGMENT PER DATA BASE RECORD
- DEPENDENT SEGMENTS -- 0 TO N PER PARENT

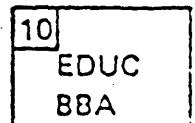
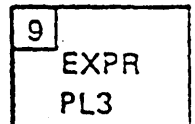
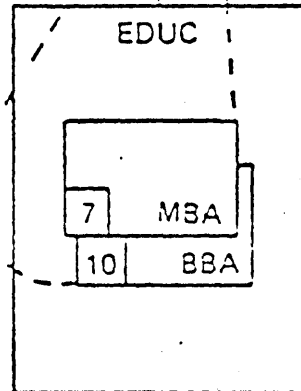
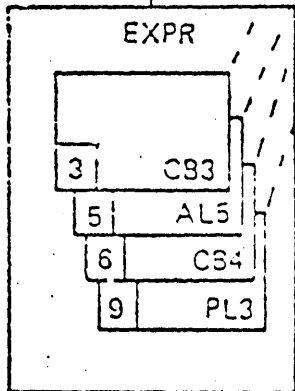
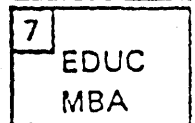
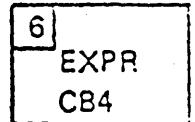
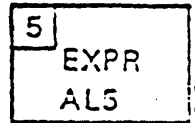
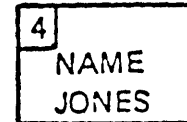
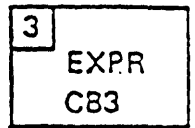
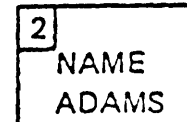
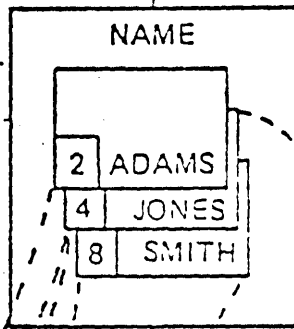
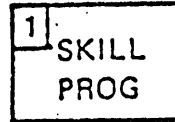


DATA BASE RECORD

HIERARCHICAL
FORMAT



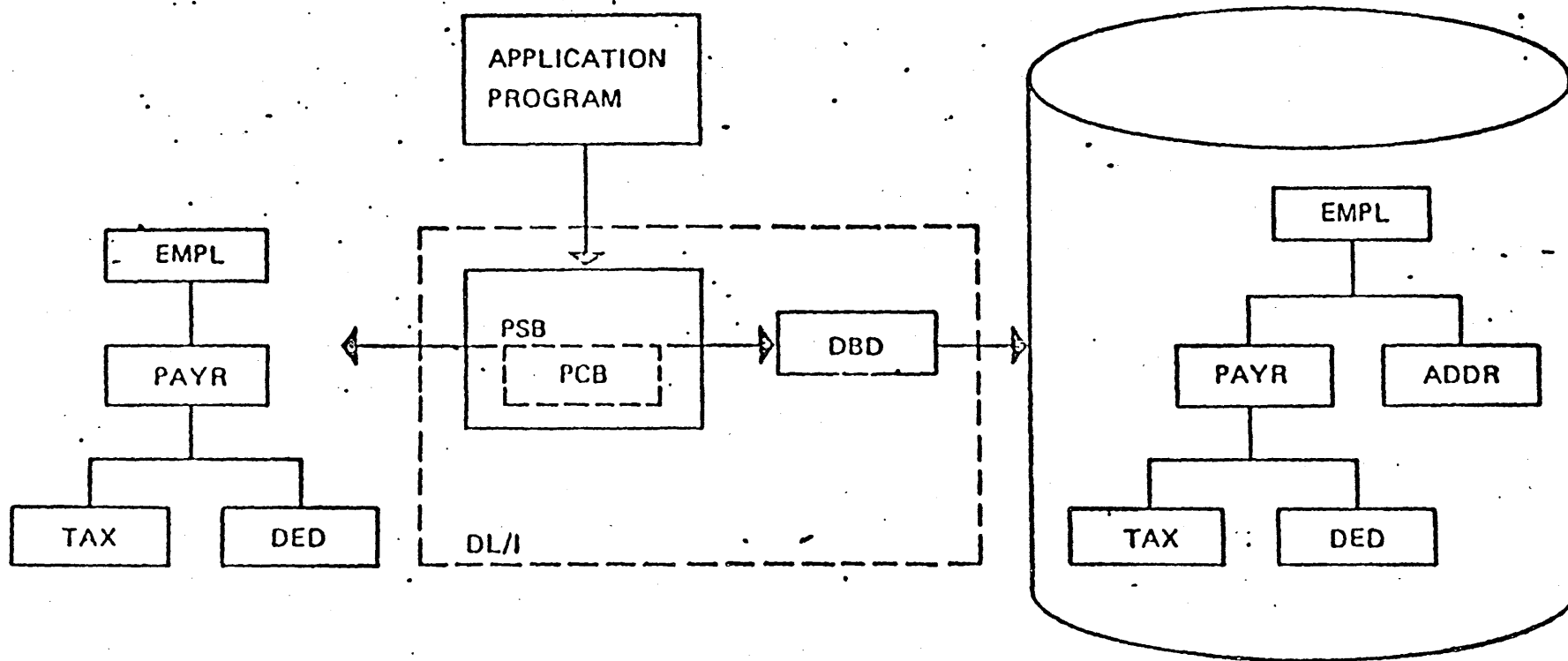
INDENTED
FORMAT



IBM-STL-10



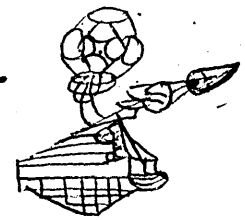
PSB-PCB-DBD. RELATIONSHIP



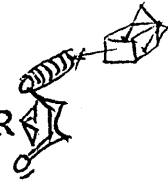
C

C

C



DL/I ADVANTAGES TO THE PROGRAMMER



DATA INDEPENDENCE

- NO FILE DESCRIPTION CODING WITHIN PROGRAM
- NO RE-COMPILATIONS DUE TO CHANGES IN THE PHYSICAL STORAGE OR ACCESS METHOD
- SAME TYPE OF CODING REGARDLESS OF LANGUAGE OR PHYSICAL ACCESS METHOD

DATA SENSITIVITY

- * LESS STORAGE TO RESERVE.
- ⊗ FEWER FIELDS TO DEFINE
- ✈ NO RE-COMPILATIONS DUE TO ADDITION/DELETION OF NON-SENSITIVE DATA ITEMS

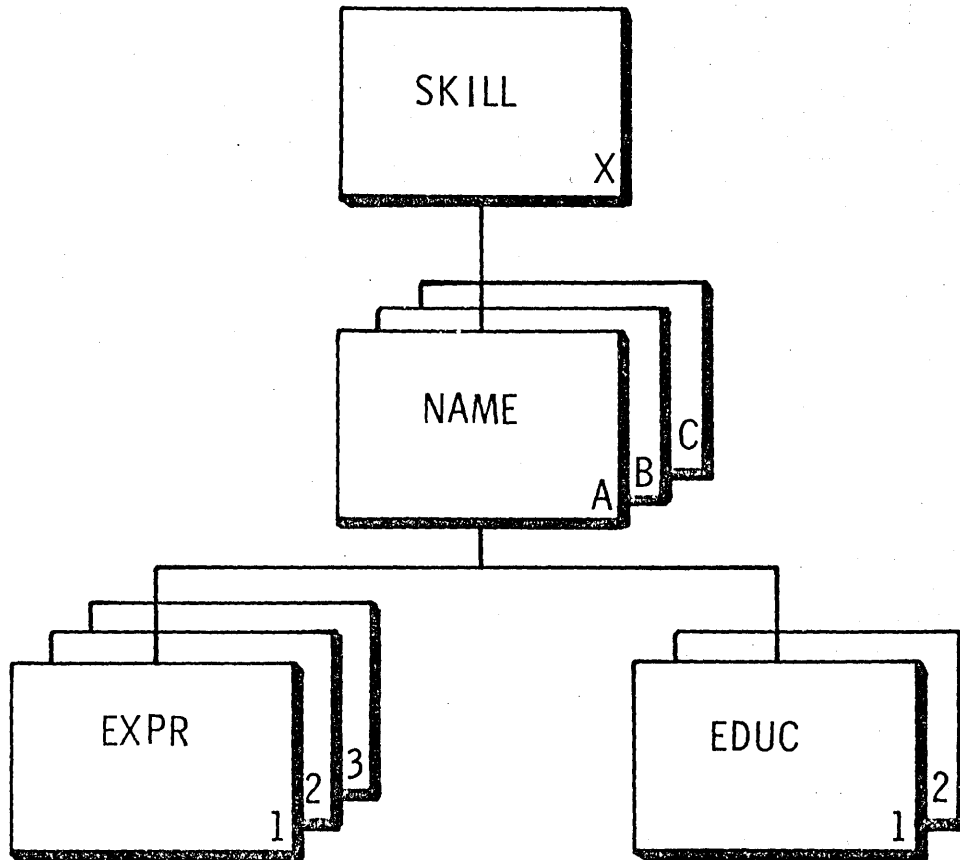
DATA INTEGRATION

- ✈ LESS FILE-RELATED LOGIC TO CODE
- ⊗ ONLY ONE FIELD IN ONE DATA BASE TO MAINTAIN

DATA BASE INTEGRITY

- NO LOGIC OR CODING FOR RECOVERY

Call Functions



GET UNIQUE - *specific occurrence (e.g. name under particular skill)*

GET NEXT - *not specific, follows hierarchy*

GET NEXT WITHIN PARENT - *children returned to segments under immediate predecessor (i.e. parent).*

INSERT

DELETE

REPLACE

Note: Multiplicity of segments under a ~~given~~ given parent not obtainable by using function.

Retrieval Functions

GET UNIQUE

GET NEXT

GET NEXT IN PARENT

GET HOLD $\begin{pmatrix} U \\ N \\ NP \end{pmatrix}$. . . ~~LOCKS~~ LOCKS RECORD
UNTIL DELETE OR REPLACE OR
MOVE TO NEXT RECORD

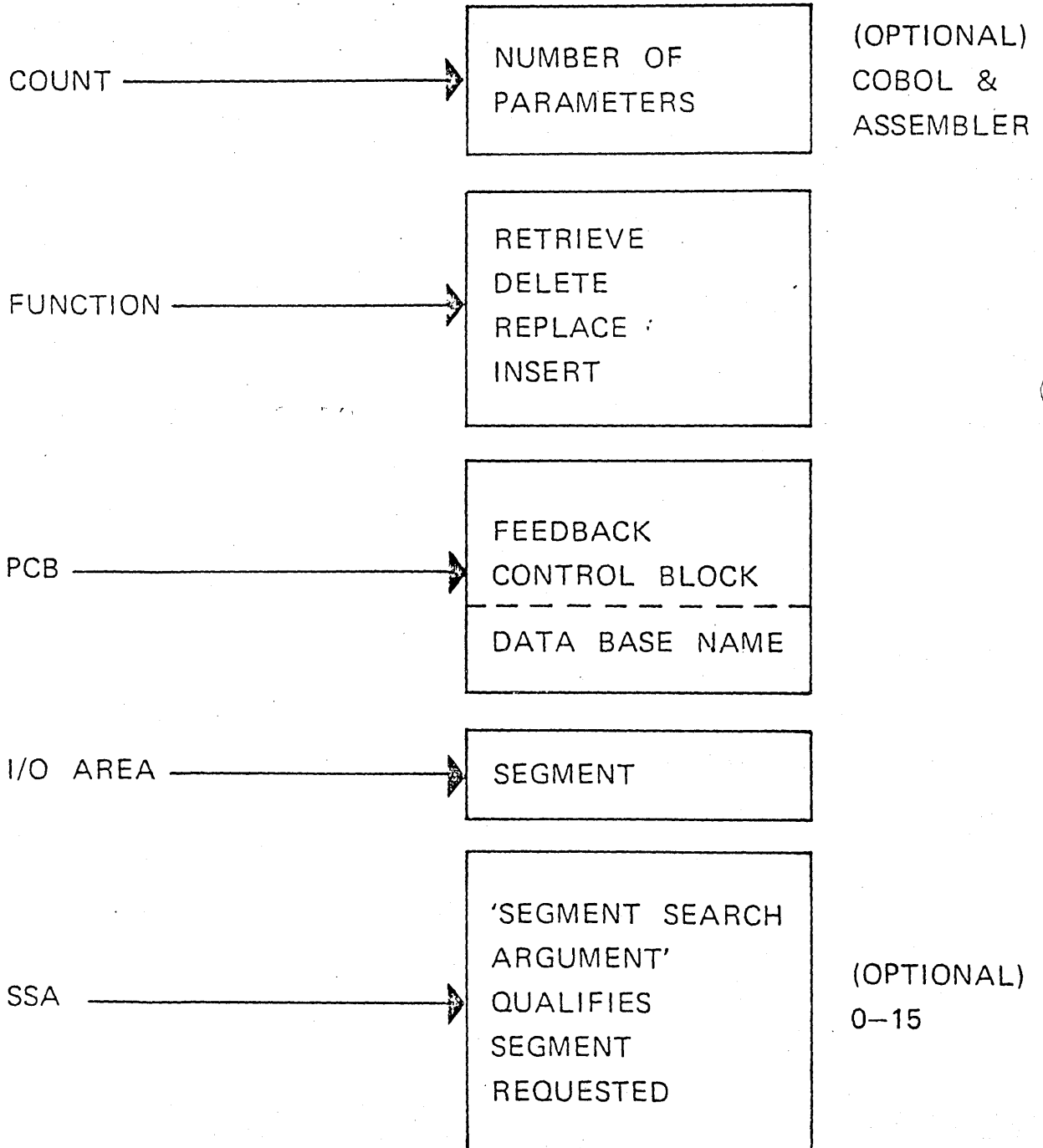
EACH FUNCTION (CALL) RETRIEVES ONE OR

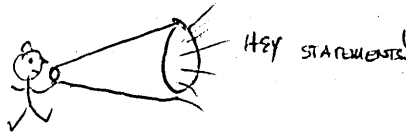
MORE SEGMENTS

PATH CALL - BRINGS IN ALL SEGMENTS IN PATH TO END SEGMENT

DL/I CALLS

CALL 'DL/I'





Call Statements

COBOL

```
CALL 'CBLTDLI' USING FUNCTION, DATA-BASE,  
      IO-AREA, SEARCH-ARG1, SEARCH-ARG2.
```

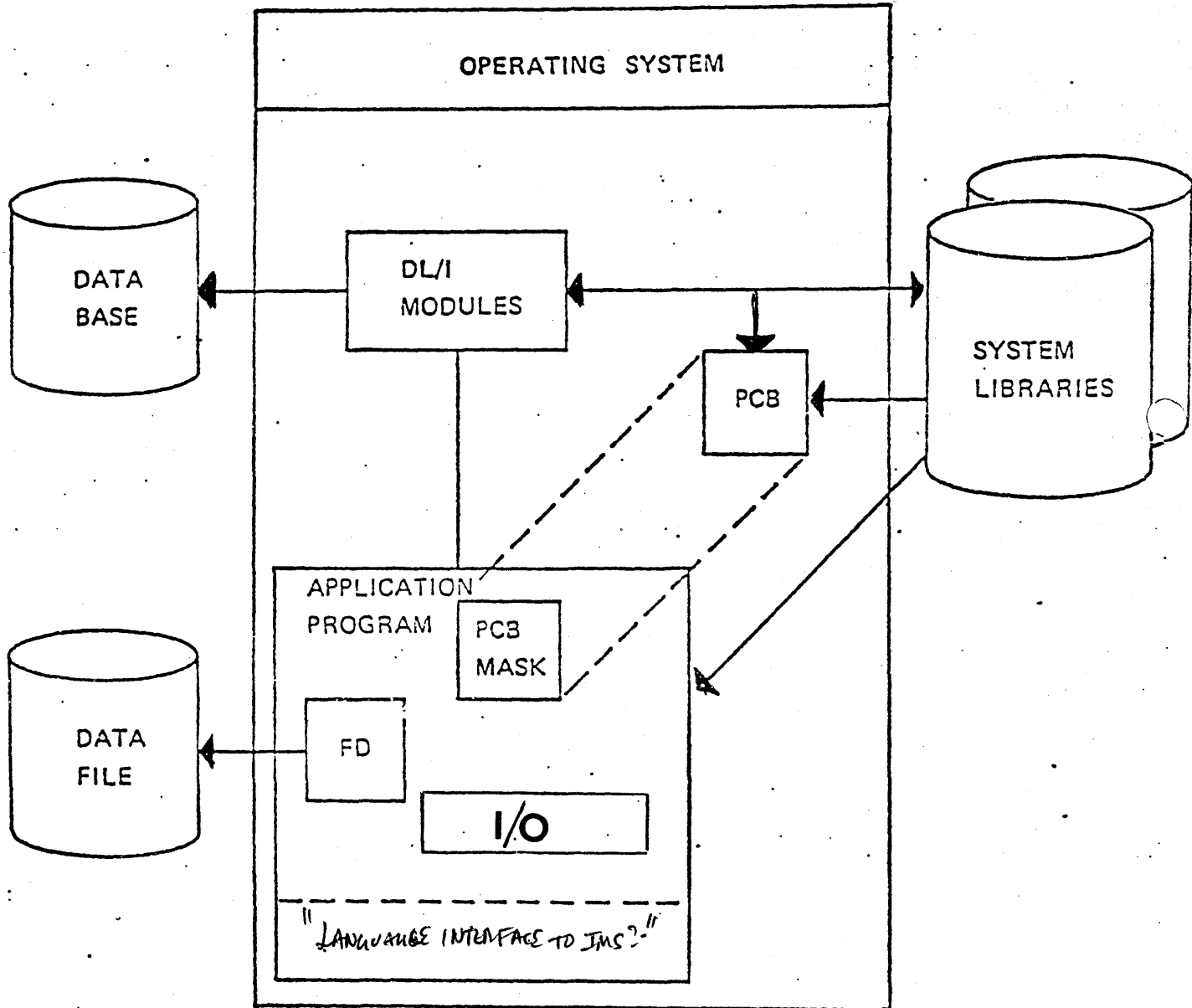
PL/I

```
CALL PLITDLI (PARM_COUNT, FUNCTION, DATA_BASE,  
              IO_AREA, SEARCH_ARG1, SEARCH_ARG2);
```

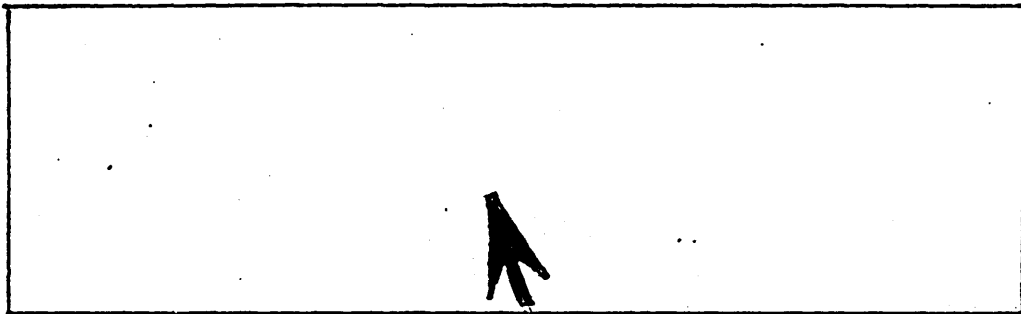
SEARCH ARGUMENTS

```
SEGNAME(FLD-NAME=FLD-VALUE)
```

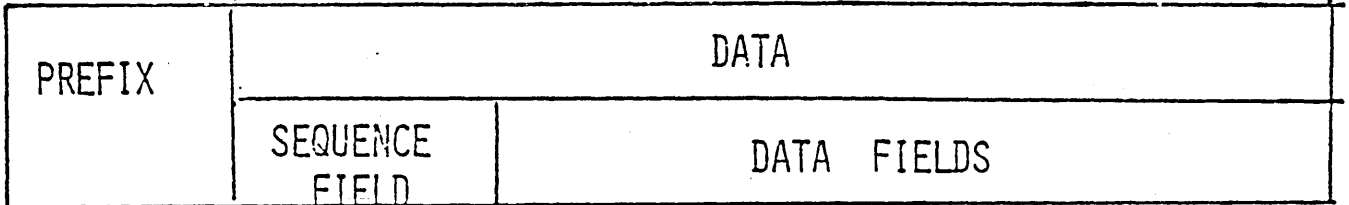
COMMUNICATING WITH DATA BASES



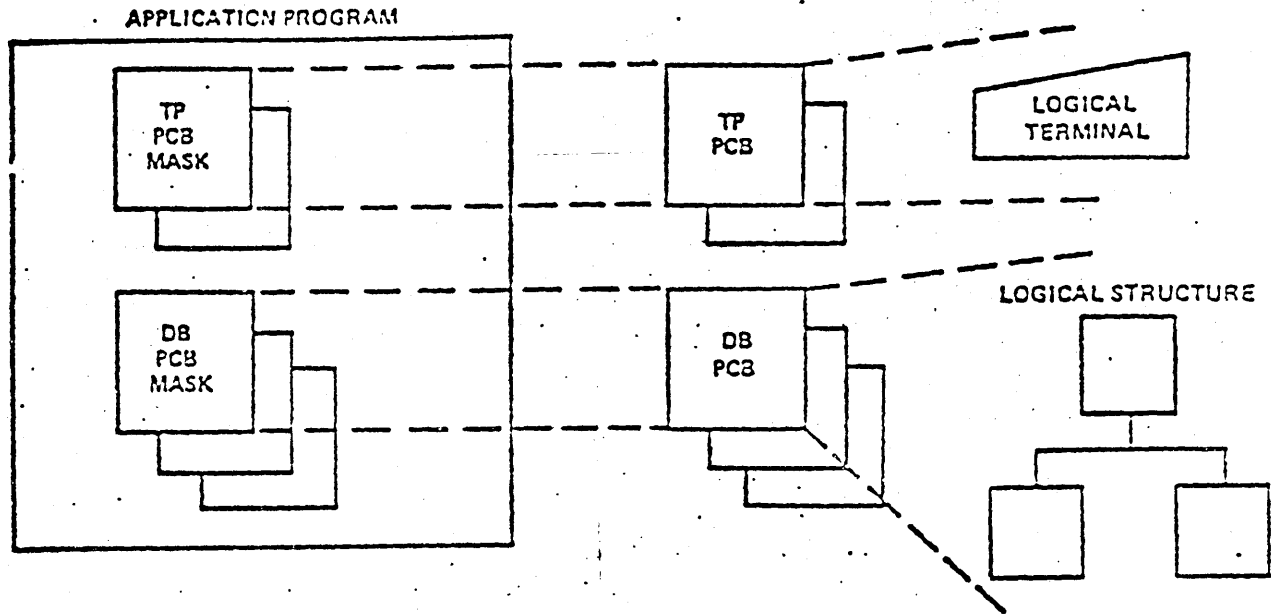
I / O AREA



SEGMENT



PCB's



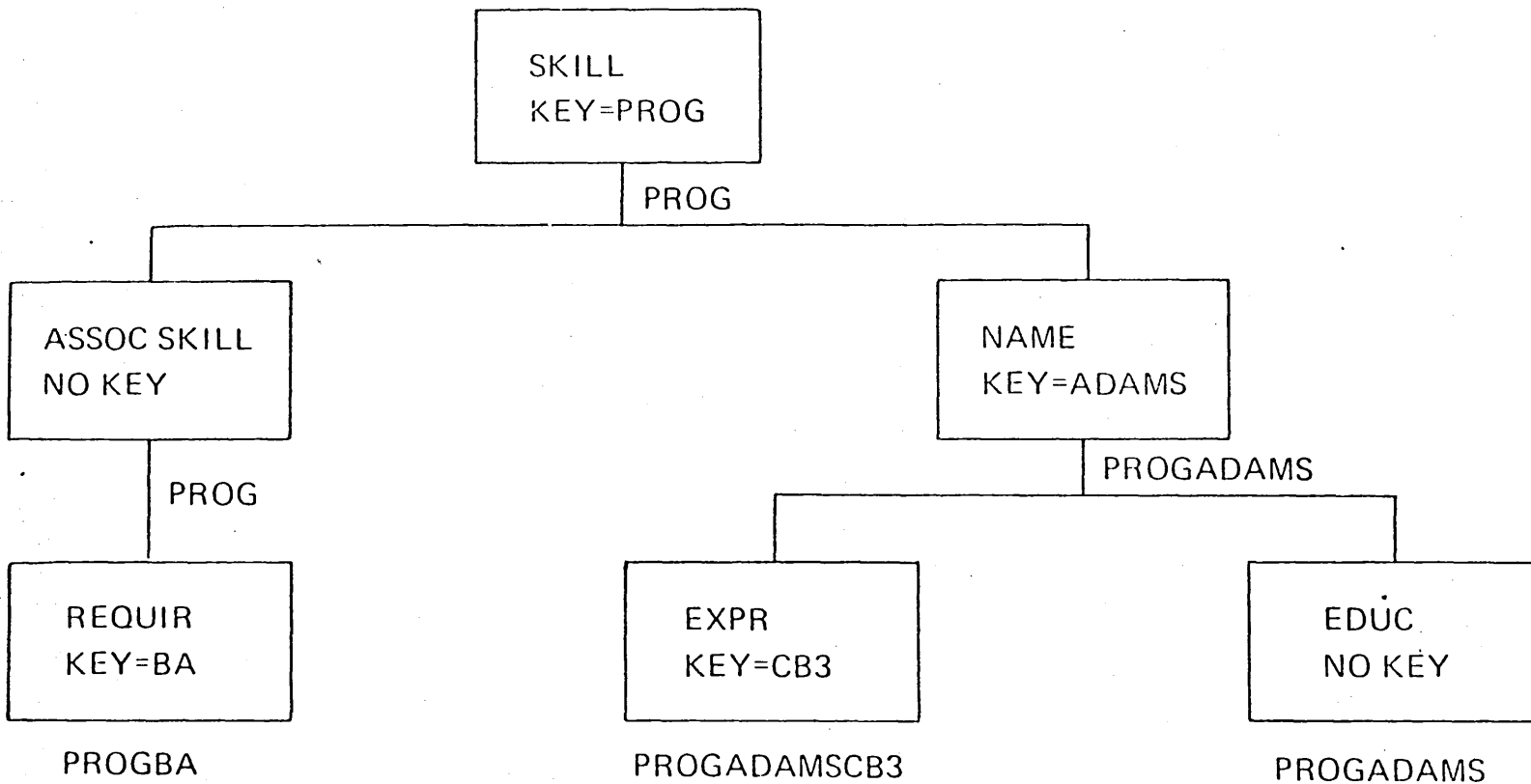
DB

DATA BASE NAME 8 BYTES	
SEGMENT HIERARCHIC LEVEL	-2 BYTES
DL/I STATUS CODE	-2 BYTES
DL/I PROCESSING OPTIONS	-4 BYTES
RESERVED FOR DL/I	-4 BYTES
SEGMENT NAME FEEDBACK AREA 8 BYTES	
LENGTH OF FEEDBACK KEY	-4 BYTES
No. OF SENSITIVE SEGMENTS	-4 BYTES
KEY FEEDBACK AREA 8 BYTES PER KEY	

TP

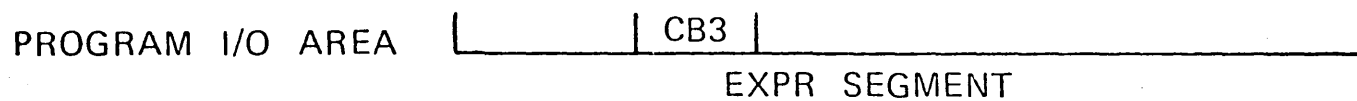
SOURCE/DESTINATION NAME 8 Bytes		
RESERVED FOR IMS/VIS 2 Bytes		
STATUS CODE 2 Bytes		
INPUT PREFIX	CURRENT DATE 4 Bytes	INPUT PREFIX
	CURRENT TIME 4 Bytes	
	INPUT MESSAGE SEQUENCE NUMBER 4 Bytes	
MESSAGE OUTPUT DESCRIPTION NAME 8 Bytes (I/O PCS only)		

CONCATENATED KEYS



RETRIEVAL OF EXPR SEGMENT

PCB KEY FEEDBACK AREA - "PROGADAMSCB3"



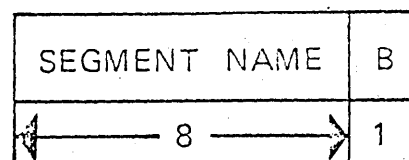
SEGMENT SEARCH ARGUMENT (SSA)

MAY BE USED TO QUALIFY A CALL IN TWO WAYS:

1 IDENTIFY THE SEGMENT TYPE

(FOR EXAMPLE, RETRIEVE ALL
EMPLOYEES
WITHIN DATA
BASE)

UNQUALIFIED SSA

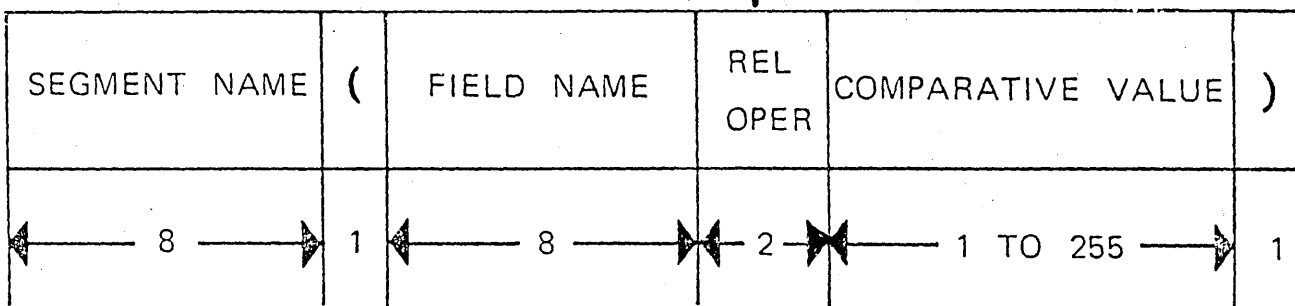


GN EMPLOYEEb

2 IDENTIFY THE SPECIFIC OCCURANCE OF A
SEGMENT TYPE

- | | | | |
|----|-------|----|-------|
| b= | OR EQ | T= | OR NE |
| b> | OR GT | => | OR GE |
| b< | OR LT | =< | OR LE |

QUALIFIED SSA



(FOR EXAMPLE, RETRIEVE EMPLOYEE 123456)

GN EMPLOYEE(EMPYNObbb=123456)

COMMAND CODES
EXTEND THE FUNCTION OF SSA

*SSA FORMAT
WITHOUT COMMAND CODES*

SEGMENT NAME	BEGIN QUAL	QUALIFICATION STATEMENT			END QUAL
	(FIELD NAME	RO	COMPARATIVE VALUE)
8 BYTES	1	8	2	1 TO 255	1

*SSA FORMAT
WITH COMMAND CODES*

SEGMENT NAME	COMMAND CODES		BEGIN QUAL	QUALIFICATION STATEMENT			END QUAL
	*	CODES	(FIELD NAME	RO	COMPARATIVE VALUE)
8 BYTES	1	VARIABLE	1	8	2	1 TO 255	1

- * IN 9th POSITION INDICATE ONE OR MORE COMMAND CODE(S) TO FOLLOW

- COMMAND CODE(S) TERMINATED BY :

a) (IF SSA QUALIFIED

b)) IF SSA UNQUALIFIED

COMMAND CODE DEFINITIONS

CODE	MEANING
F	START WITH THE FIRST OCCURRENCE OF THIS SEGMENT-TYPE UNDER ITS PARENT IN ATTEMPTING TO SATISFY THIS LEVEL OF THE CALL.
L	RETRIEVE THE LAST OCCURRENCE OF THIS SEGMENT-TYPE UNDER ITS PARENT, (IF UNQUALIFIED); OR THE LAST OCCURRENCE WHICH SATISFIES THE QUALIFICATION.
	NOTE: 'F' AND 'L' IGNORED AT ROOT LEVEL.
P	SETS PARENTAGE AT SPECIFIED LEVEL.
D	PATH CALL: ALLOWS RETRIEVAL OR INSERTION OF MULTIPLE SEGMENTS IN A HIERARCHICAL PATH WITH A SINGLE CALL.
N	NEGATES THE REPLACE FUNCTION FOR THE SPECIFIED LEVEL FOLLOWING A PATH RETRIEVAL CALL.
U	TREATS THIS SSA AS IF IT WERE QUALIFIED ON THE KEY FIELD OF THE CURRENT SEGMENT.
V	SAME AS 'U', EXCEPT THAT IT ALSO APPLIES TO ALL HIGHER LEVELS.
C	DATA CONTAINED IN COMPRESSED FORM WITHIN QUALIFICATION STATEMENT IS CONCATENATED KEY OF NAMED SEGMENT.
-	NULL COMMAND CODE.

BOOLEAN OPERATORS
"METHOD OF USING UP TO 8 QUALIFIERS FOR AN SSA"

BOOLEAN STATEMENT											
SEGMENT	COMMAND	BEGIN	QUALIFICATION				END QUAL.	QUALIFICATION			END
NAME	CODES	QUAL.	STATEMENT	FIELD	RO	COMPAR- ATIVE VALUE) OR * OR + <i>AND &</i>	FIELD	RO	COMPAR- ATIVE VALUE)
8	1	VAR	(8	2	1 to 255	1	8	2	1 to 255)

QUALIFICATION STATEMENTS SEPARATED BY CONNECTORS

AND (*OR&)

OR (+ORI)

E.G. LIST ALL EDUC SEGMENTS WHERE PERSON HAS MAJORED
 IN FINANCE OR HAS RECEIVED A MBA DEGREE

EDUCbbbb(MAJOR bbbb=FINANCE + DEGREE bb b = MBA)

BOOLEAN OPERATOR RULES

BOOLEAN SETS

- 1) ALL BOOLEAN STATEMENTS CONNECTED BY "AND" OPERATORS ARE A "SET" OF QUALIFICATION STATEMENTS.
- 2) AN "OR" OPERATOR BETWEEN TWO SETS OF QUALIFICATIONS BEGINS A NEW "SET" OF QUALIFICATION STATEMENTS.

EXAMPLES:

- A) SALARY (PAYRATE > 1000 * PAYRATE < 2000)
- B) INVEN (COST => 2000 + BACKORDR > 10)

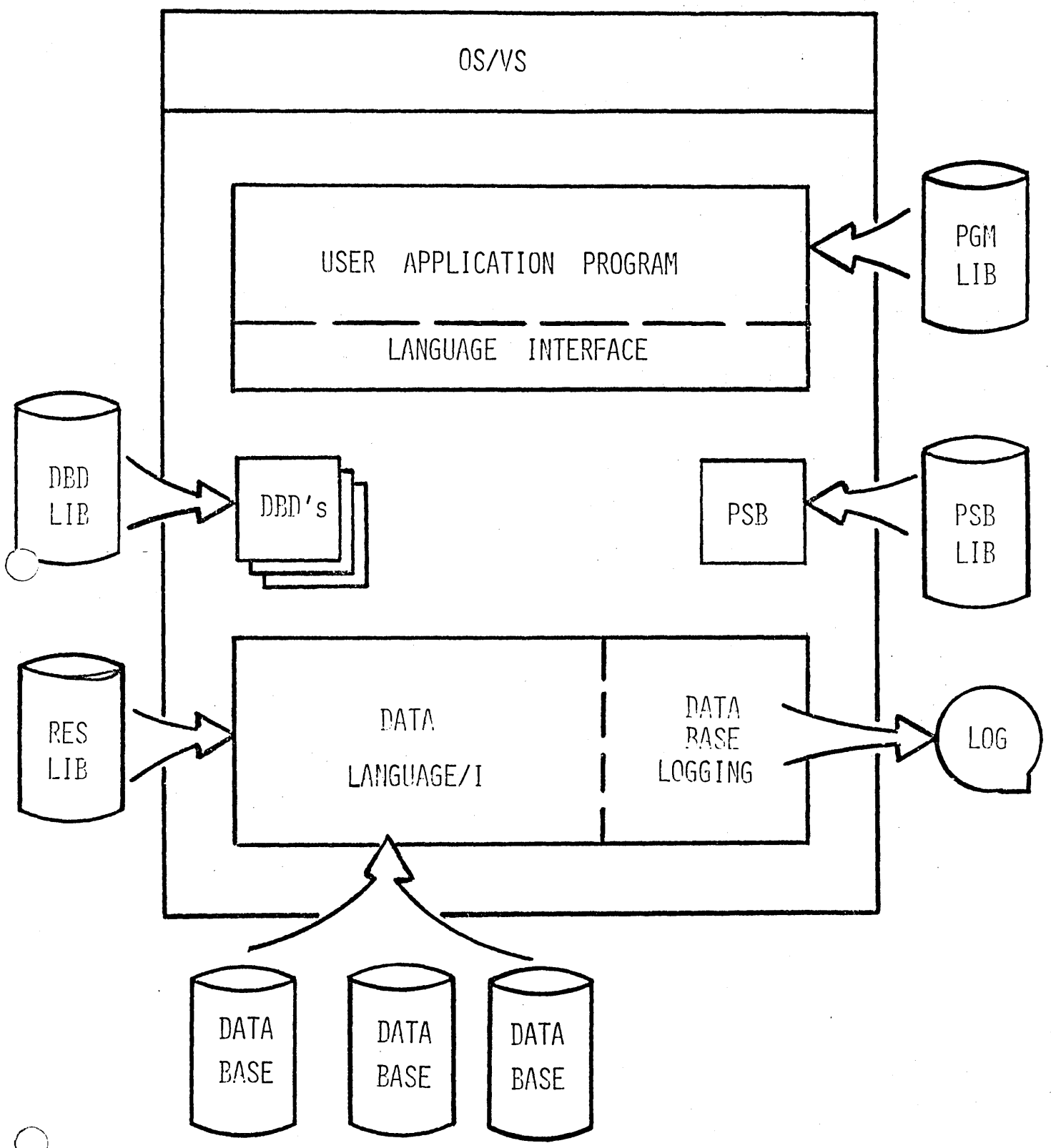
INITIAL POSITION

- 3) IF ANY SET OF QUALIFICATION STATEMENTS DOES NOT CONTAIN AT LEAST ONE STATEMENT QUALIFIED ON THE ROOT KEY, SEARCH WILL START AT THE BEGINNING OF THE DATA BASE.
- 4) IF ALL SETS HAVE AT LEAST ONE STATEMENT QUALIFIED ON THE ROOT KEY, THE LOWEST KEY FIELD VALUE IS USED FOR INITIAL POSITION.

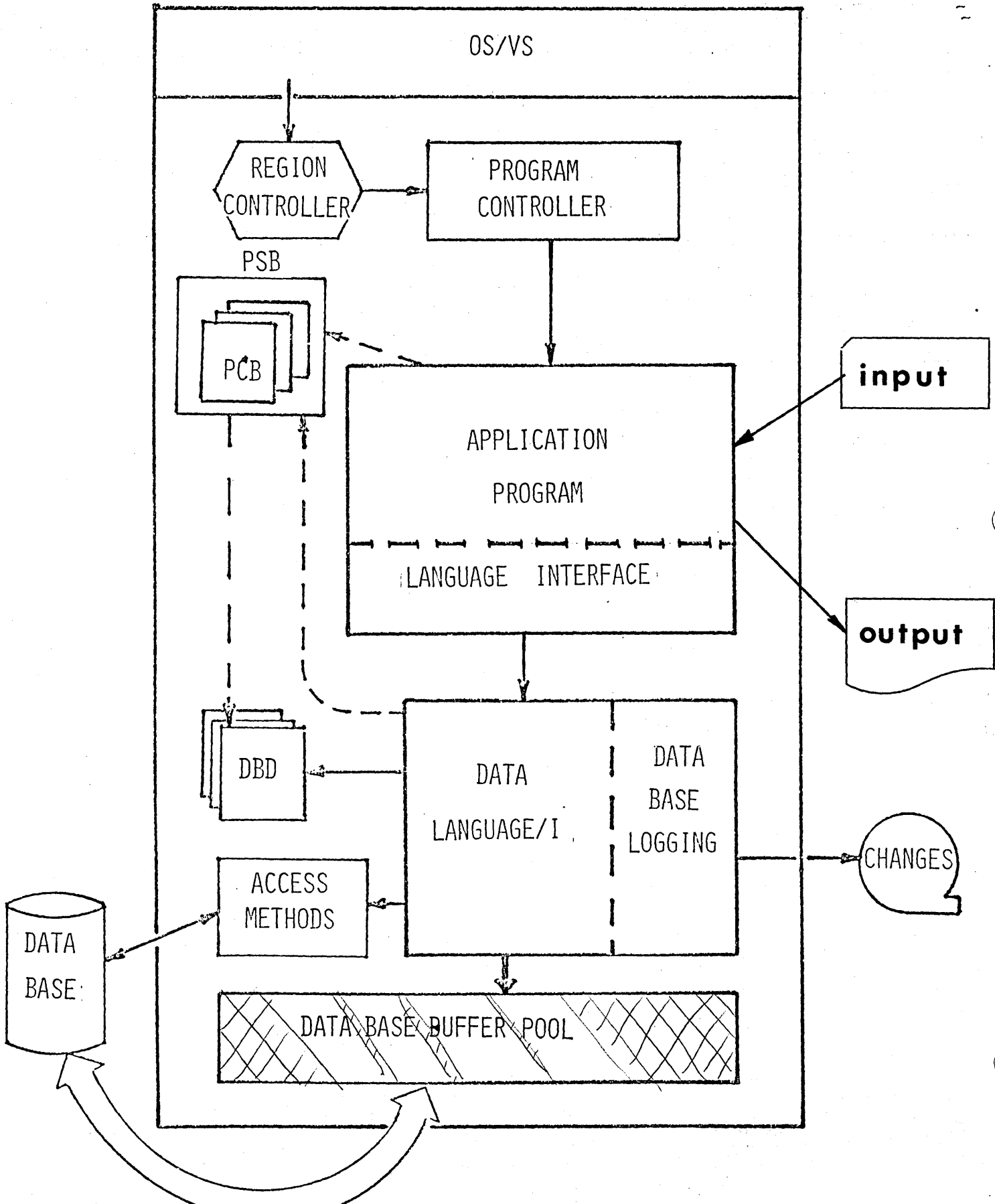
EXAMPLES:

- A) NAME (EMPLNO > 50260 * EMPLNO < 98430)
- B) SKILL (SKCLASS > OPERATOR + SKCLASS => CLERK)

IMS BATCH REGION



BATCH SYSTEM FLOW



I M S Access Methods

Part 1:

File

Organization

SEQUENTIAL

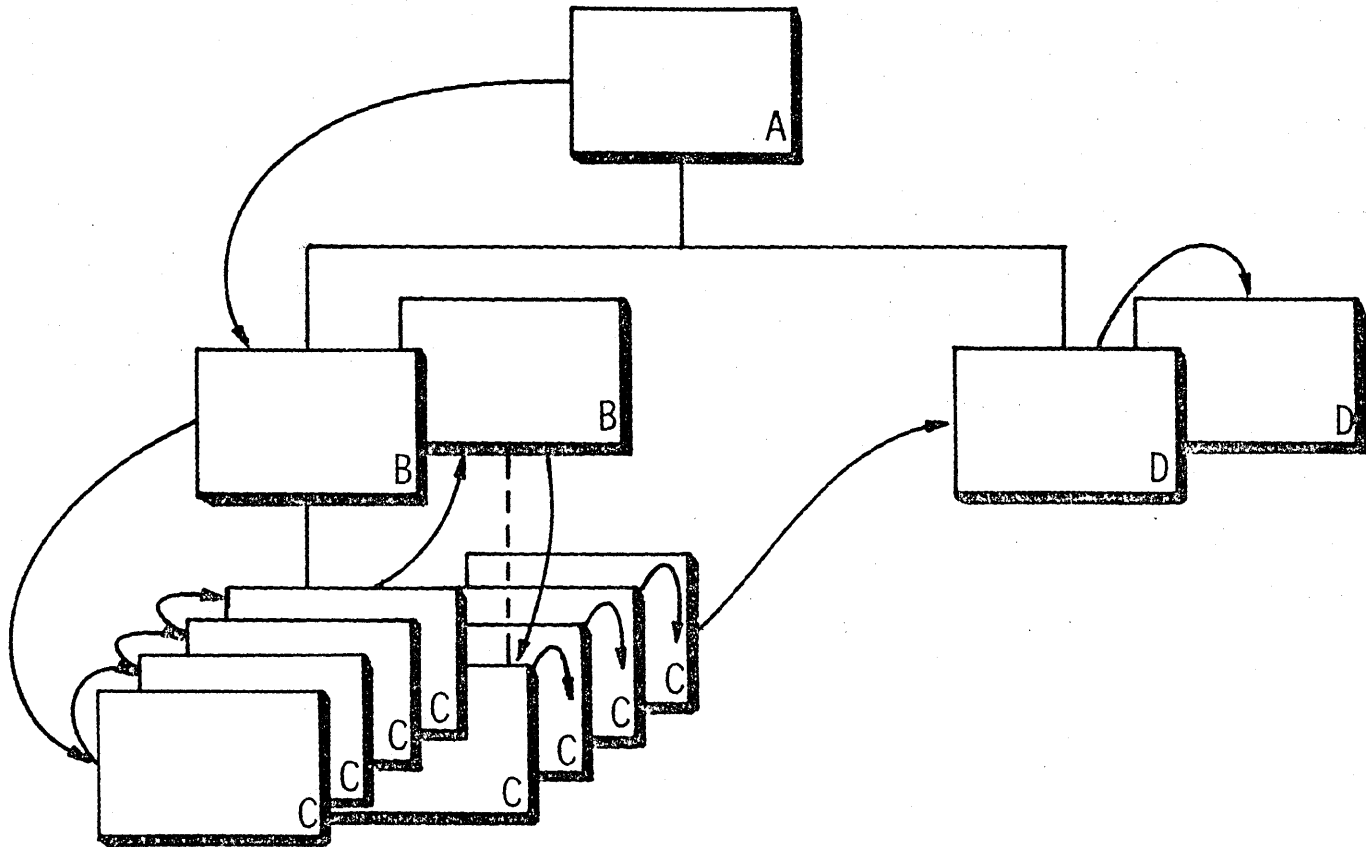
WITHOUT INDEX

WITH INDEX

DIRECT

WITHOUT INDEX

WITH INDEX

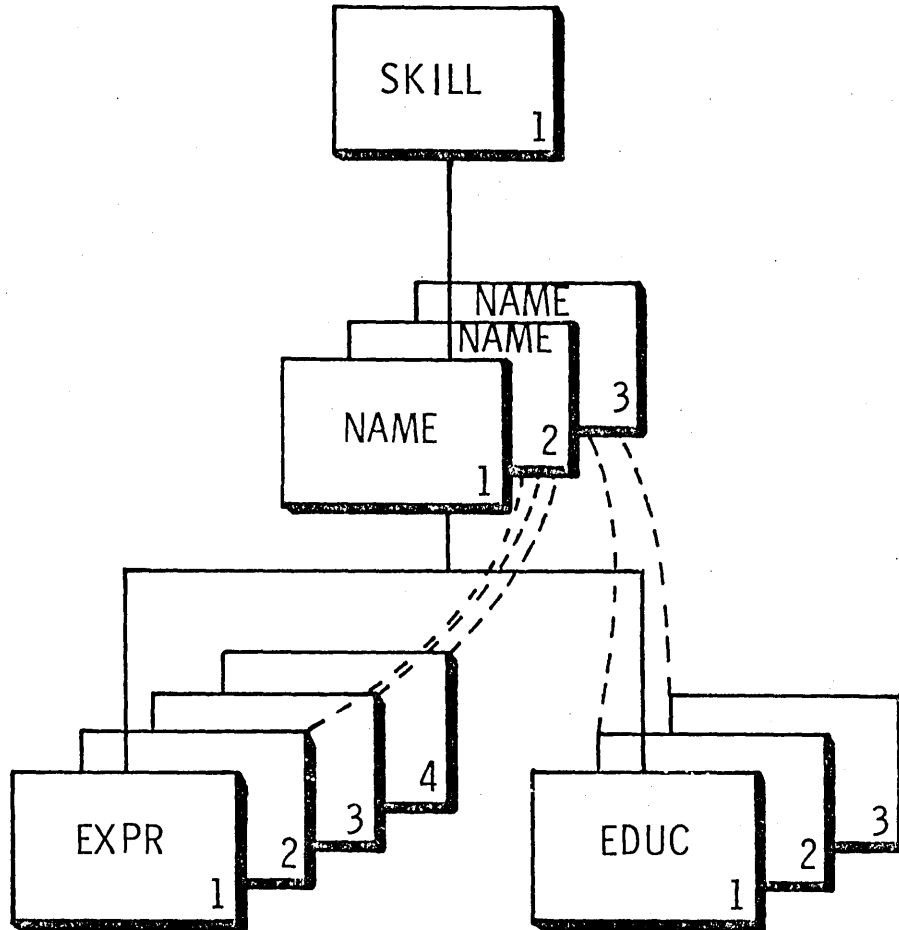


HIERARCHICAL SEQUENCE

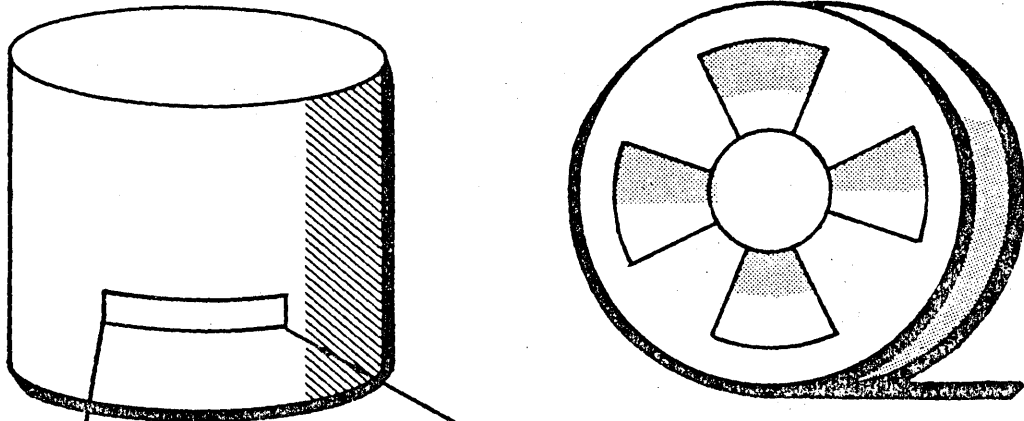
TOP-TO-BOTTOM

LEFT-TO-RIGHT

Data Base Record Structure



Sequential - Storage



SKILL A	NAME 1	EXPR 1	EDUC 1	NAME 2	
EXPR2	EXPR3	EXPR4	NAME 3	EDUC2	EDUC3
SKILL D	NAME 1	NAME 2	EDUC 1	EXPR 1	
SKILL E					

Sequential - Processing

WITHOUT INDEX

SK A	N1	EX1	ED1	N2	EX2	EX3
EX4	N3	ED2	ED3	SK C	N1	ED1
SK D	N1	EX1	N2	EX1	EX2	
SKE	SK F	N1				

WITH INDEX

INDEX	SK A:T1	SK B:T2	SK C:T2	SK D:T3
-------	---------	---------	---------	---------

SK A	N1	EX1	ED2	N2	EX2	EX3	
SK B	N1	N2	EX1	ED2	SKC	N1	ED1
SK D	N1	EX1	N2	EX1	EX2		
SKE							
SK F	N1						

Sequential Considerations

PRO FAST SEQUENTIAL PROCESSING

INDEX AVAILABLE FOR RANDOM PROCESSING

SMALL 'IMS OVERHEAD' ON STORAGE - SEGMENTS
RELATED BY ADJACENCY

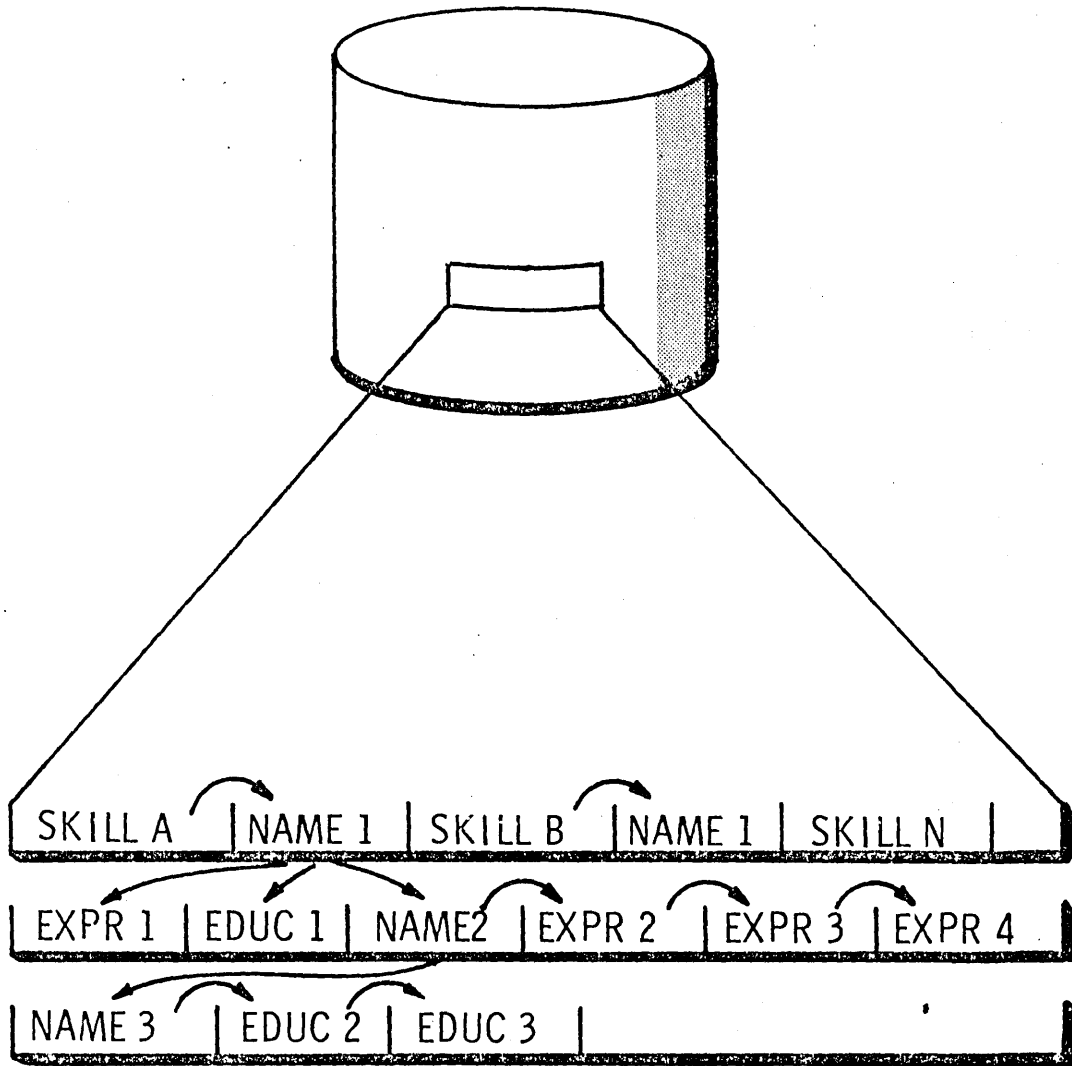
USEFUL AS CONVERSION AID

CON MAY NOT BE EFFICIENT SPACE UTILIZATION

SLOWER ACCESS TO RIGHT - MOST DEPENDENTS

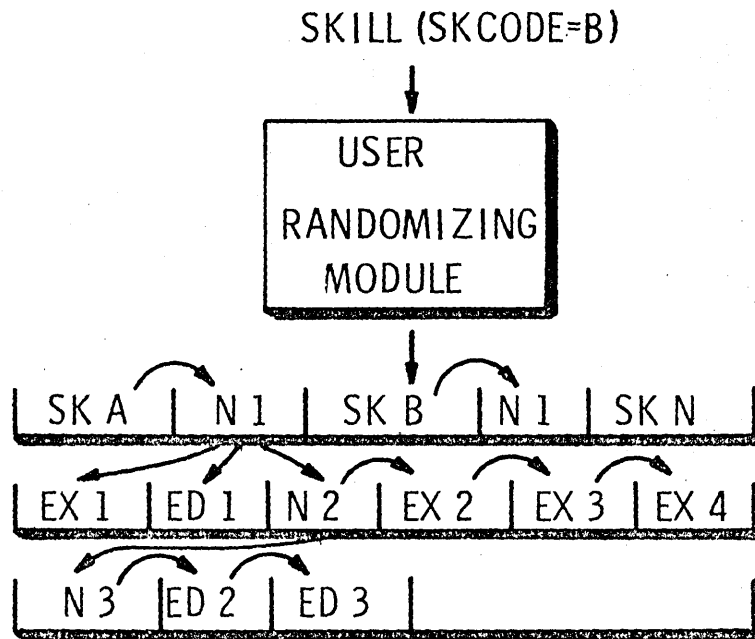
REQUIRES FREQUENT REORGANIZATION

Direct - Storage

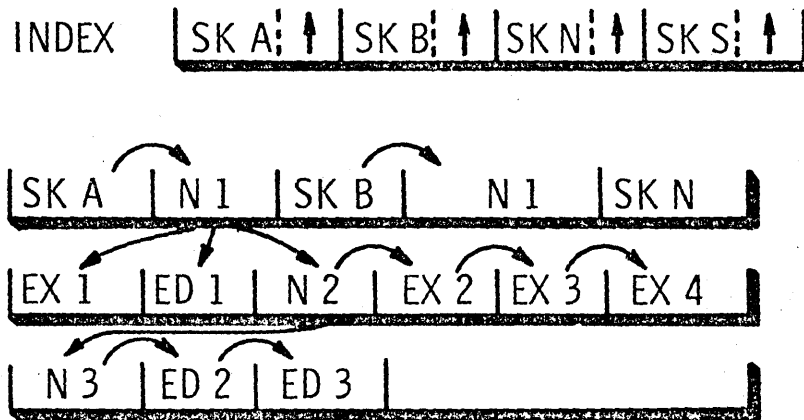


Direct - Processing

WITHOUT INDEX



WITH INDEX



Direct Considerations

PRO FAST RANDOM PROCESSING

INDEX AVAILABLE FOR SEQUENTIAL PROCESSING

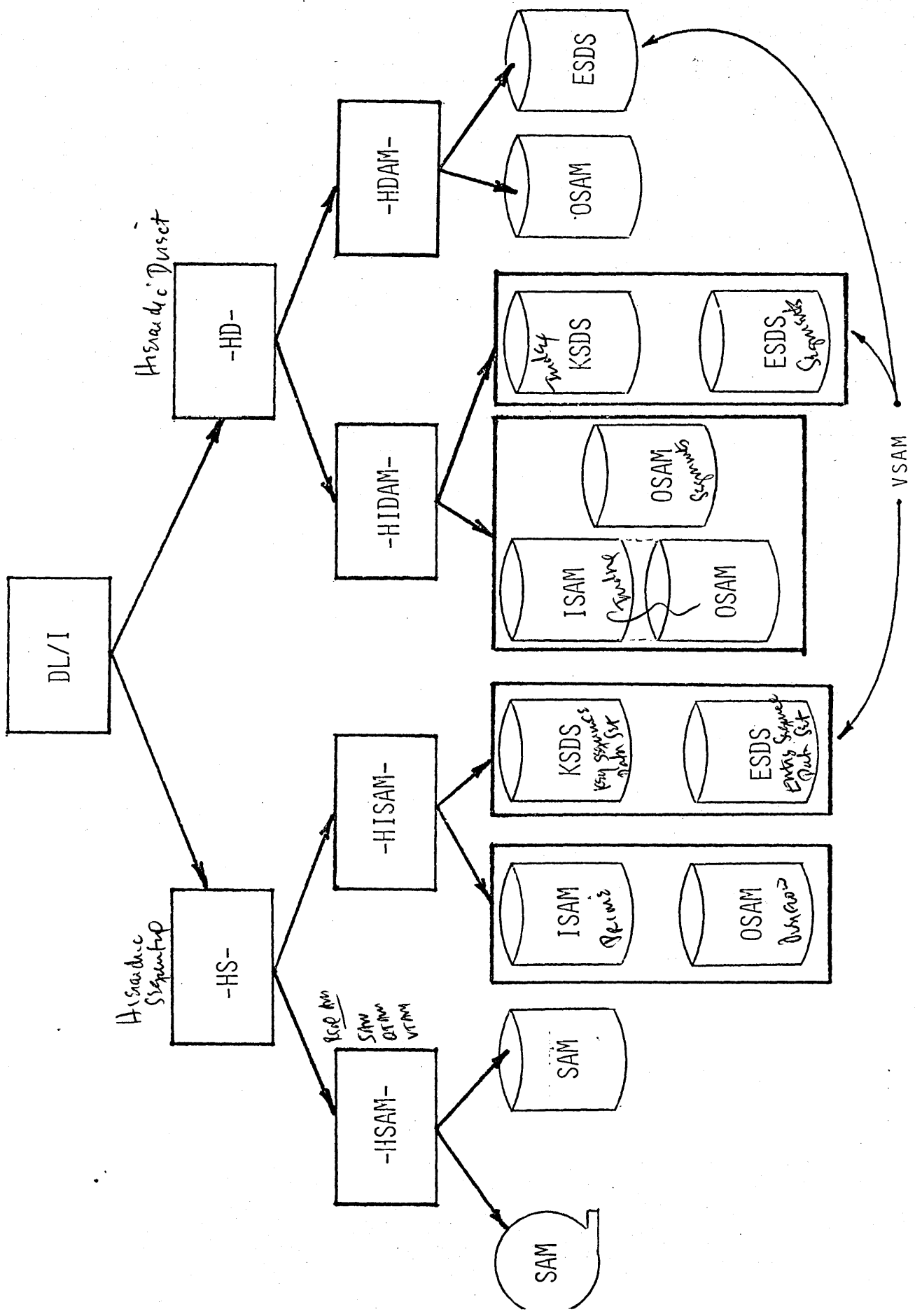
FAST ACCESS TO DEPENDENT SEGMENTS

EFFICIENT SPACE UTILIZATION

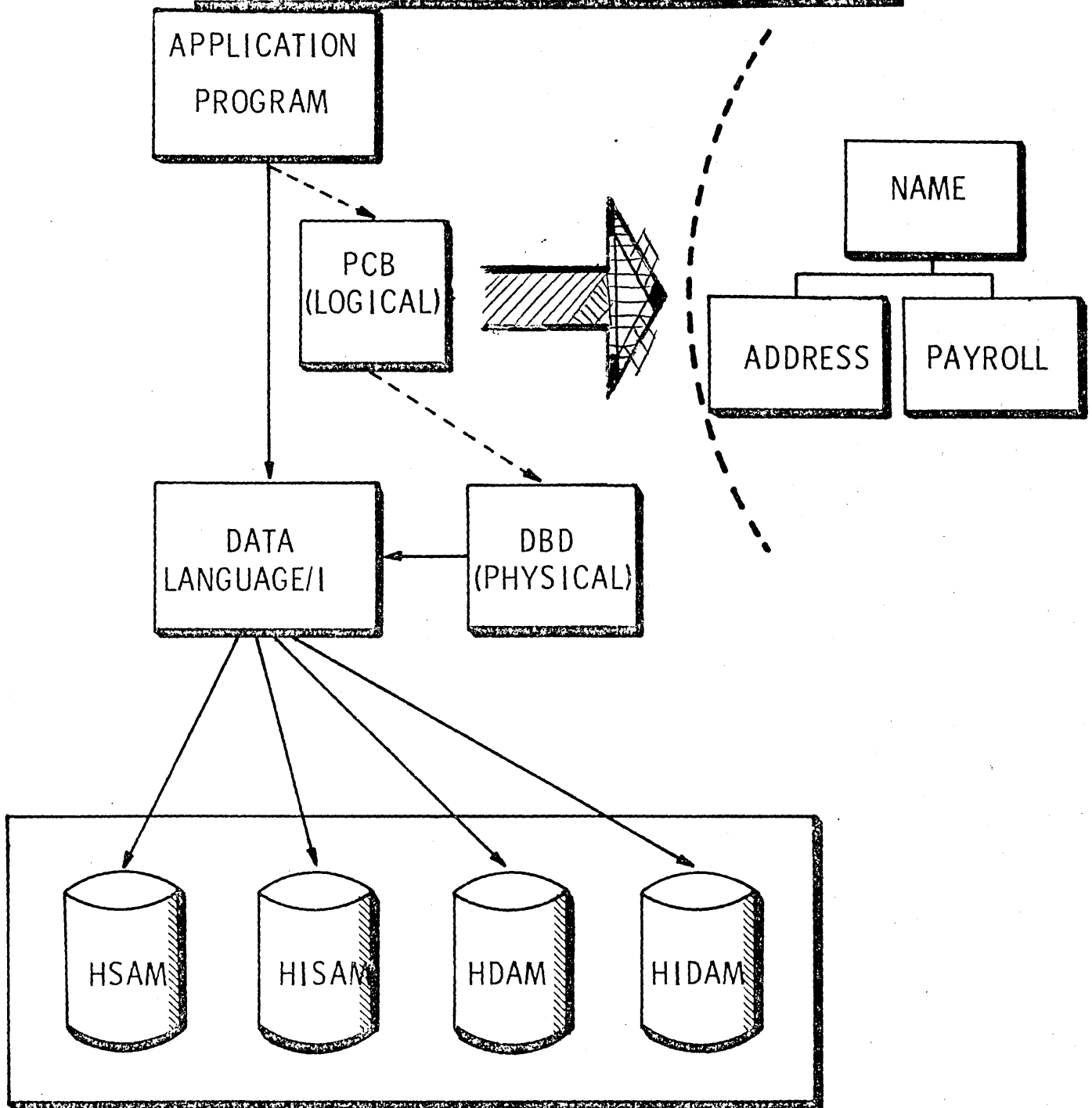
LESS FREQUENT REORGANIZATION

DIRECT ADDRESSES MAINTAINED BY SYSTEM

CON LARGER 'IMS OVERHEAD' DUE TO POINTERS, BUT...

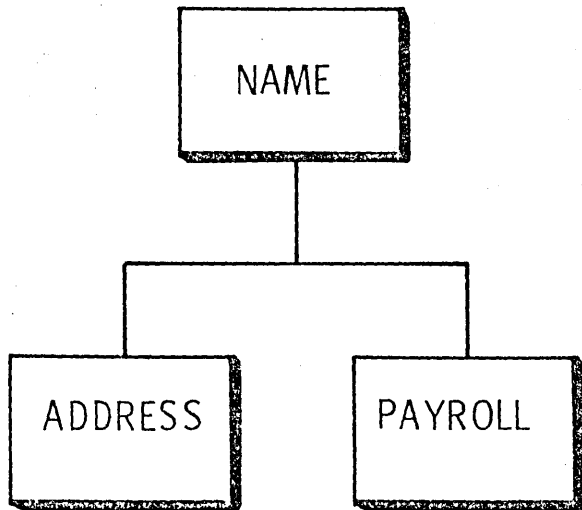


Application Program's View

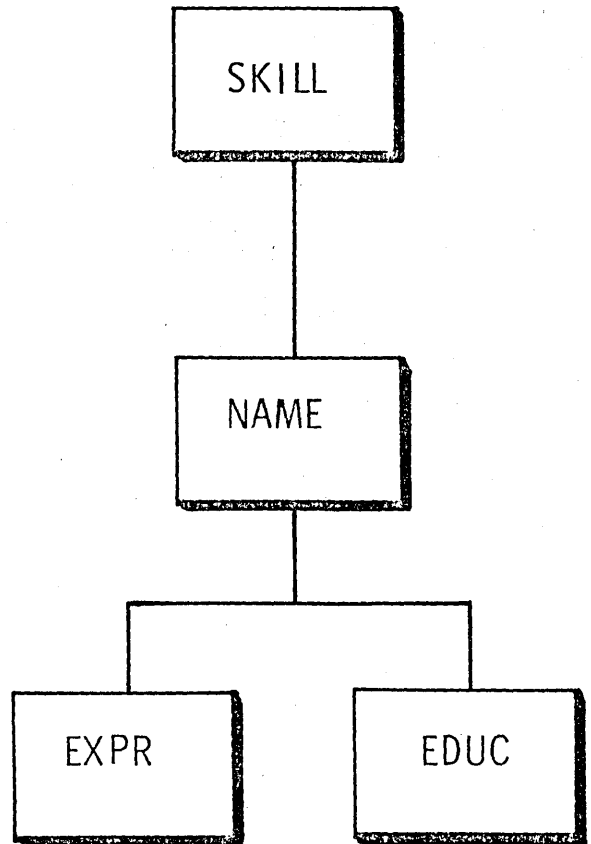


Two Physical Data Bases

PAYROLL DATA BASE



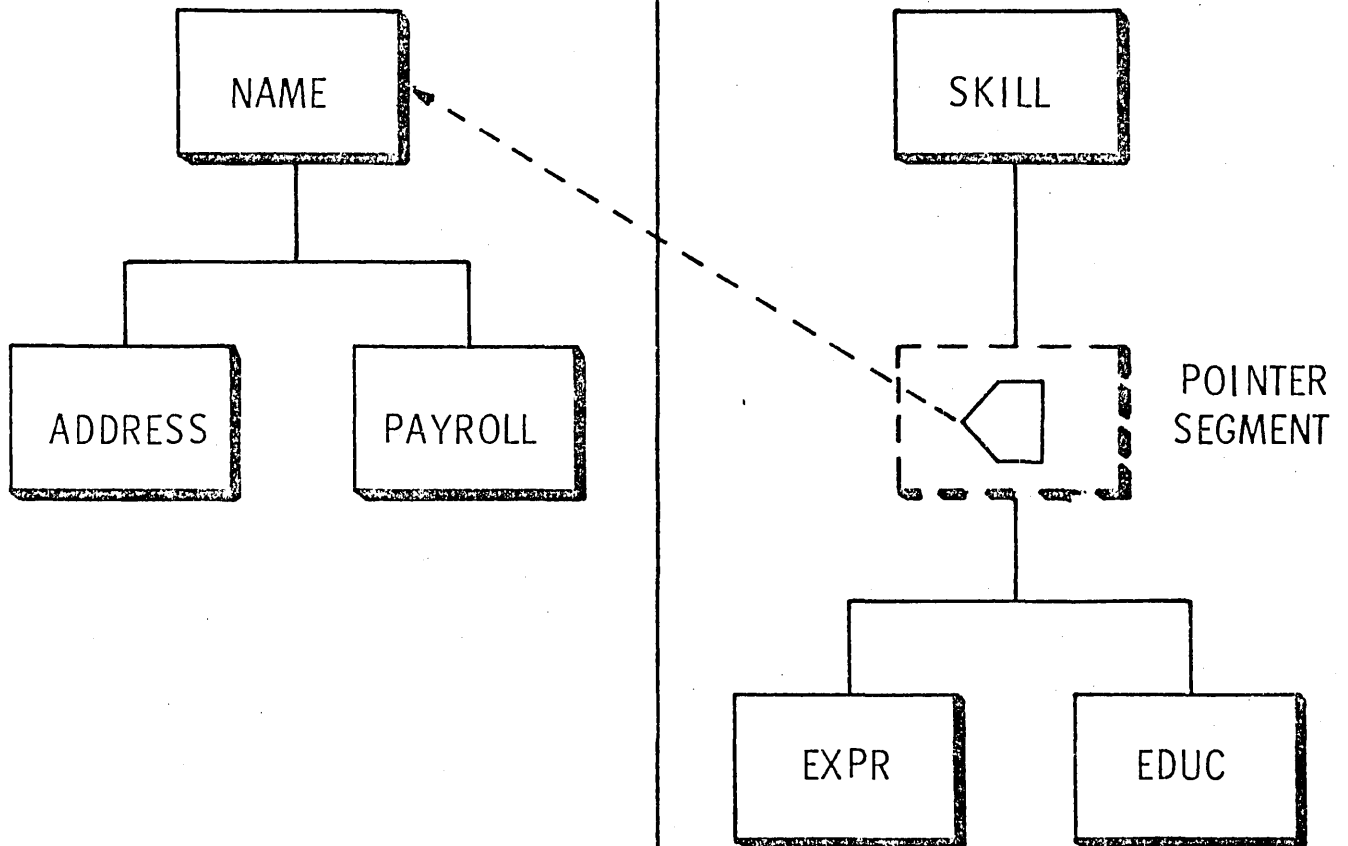
SKILLS INVENTORY DATA BASE



Interrelated Data Bases

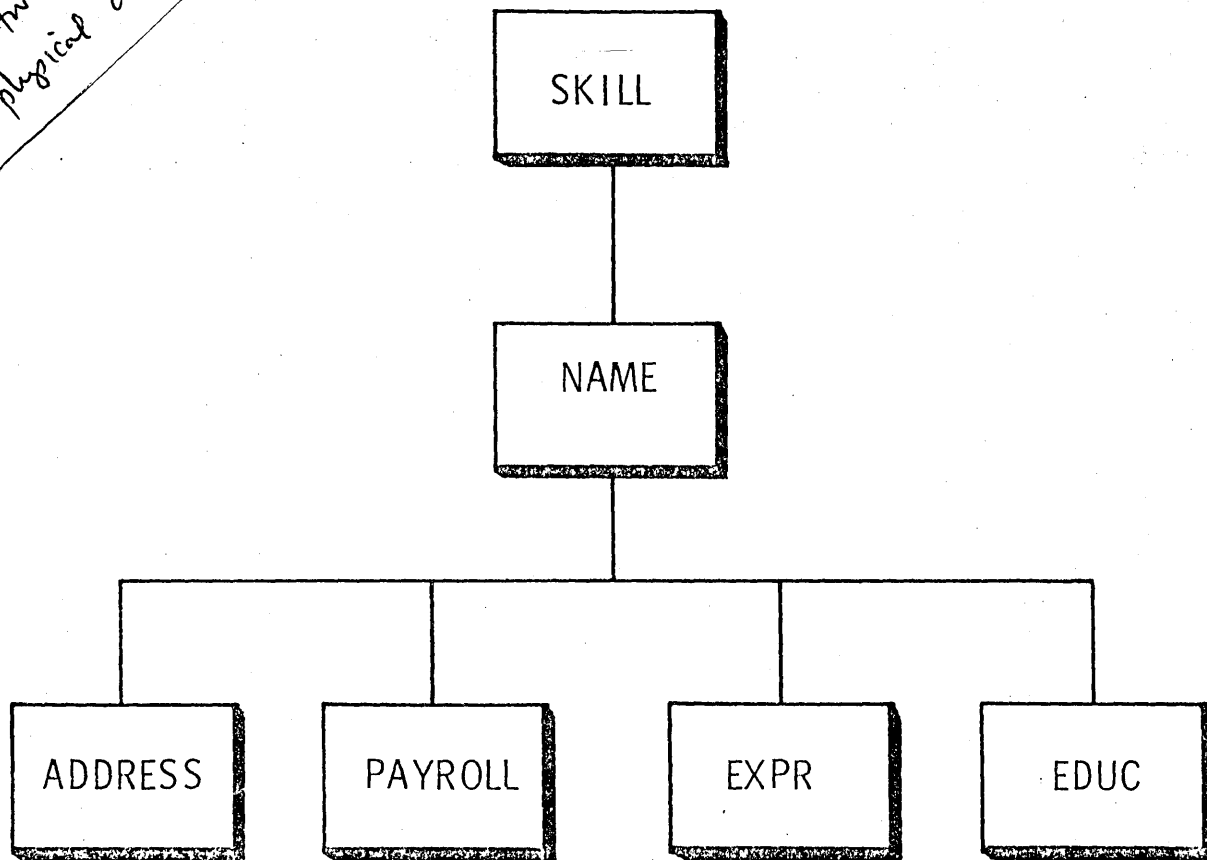
PAYROLL DATA BASE (PHYSICAL)

SKILLS INVENTORY DATA BASE (PHYSICAL)



*(Logical) Combination
of two previous
physical data bases*

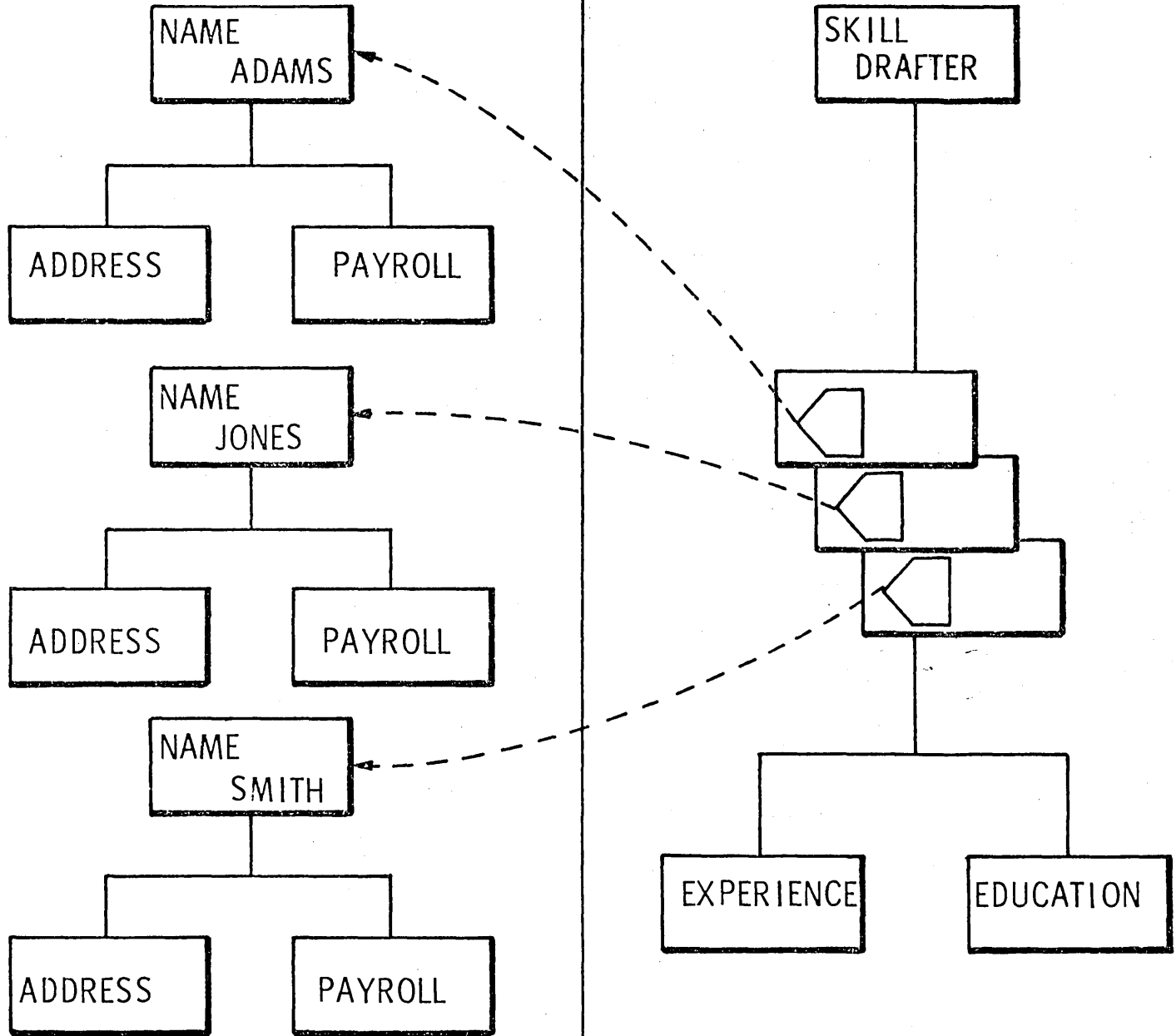
Logical Data Base



Logical Pointers

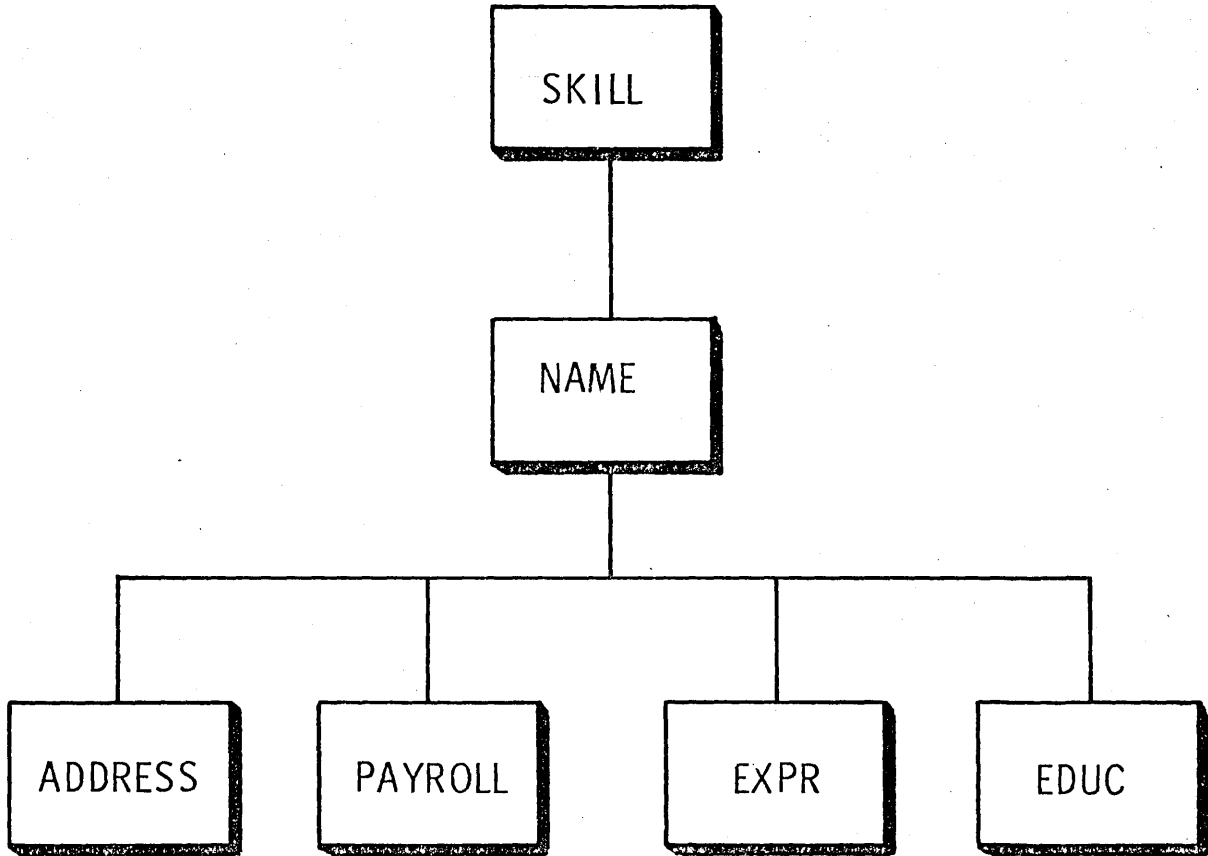
PAYROLL DATA BASE

SKILLS INVENTORY DATA BASE



Pointer Segment cannot point to pointer statement

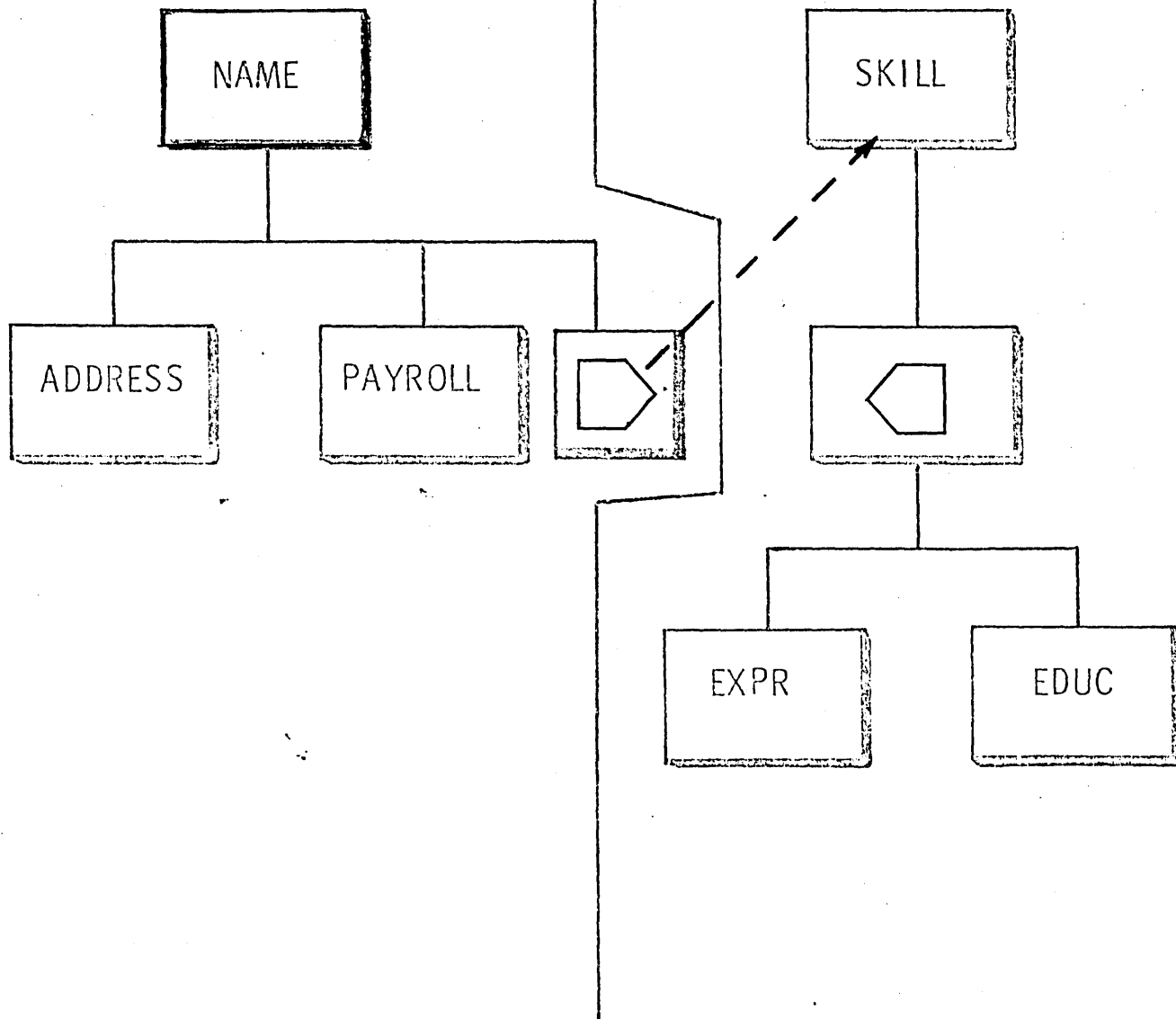
Logical Data Structure



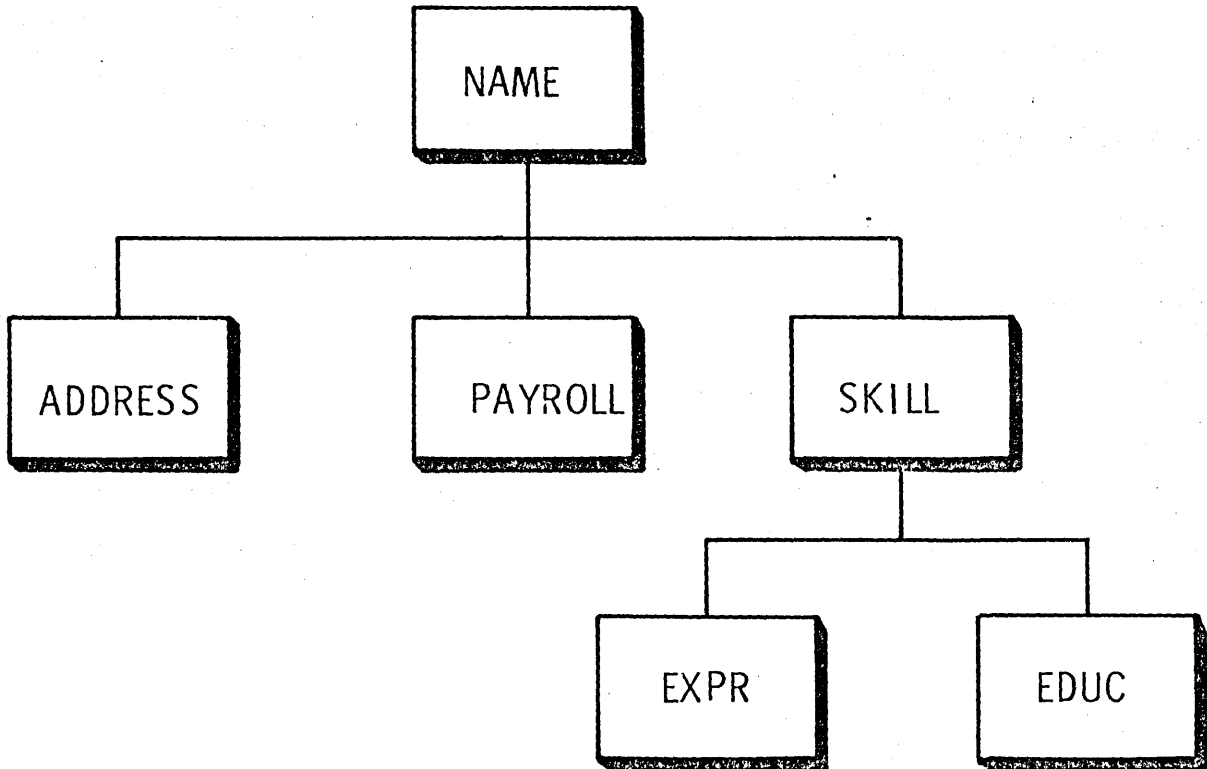
Two Physical Data Bases

PAYROLL DATA BASE

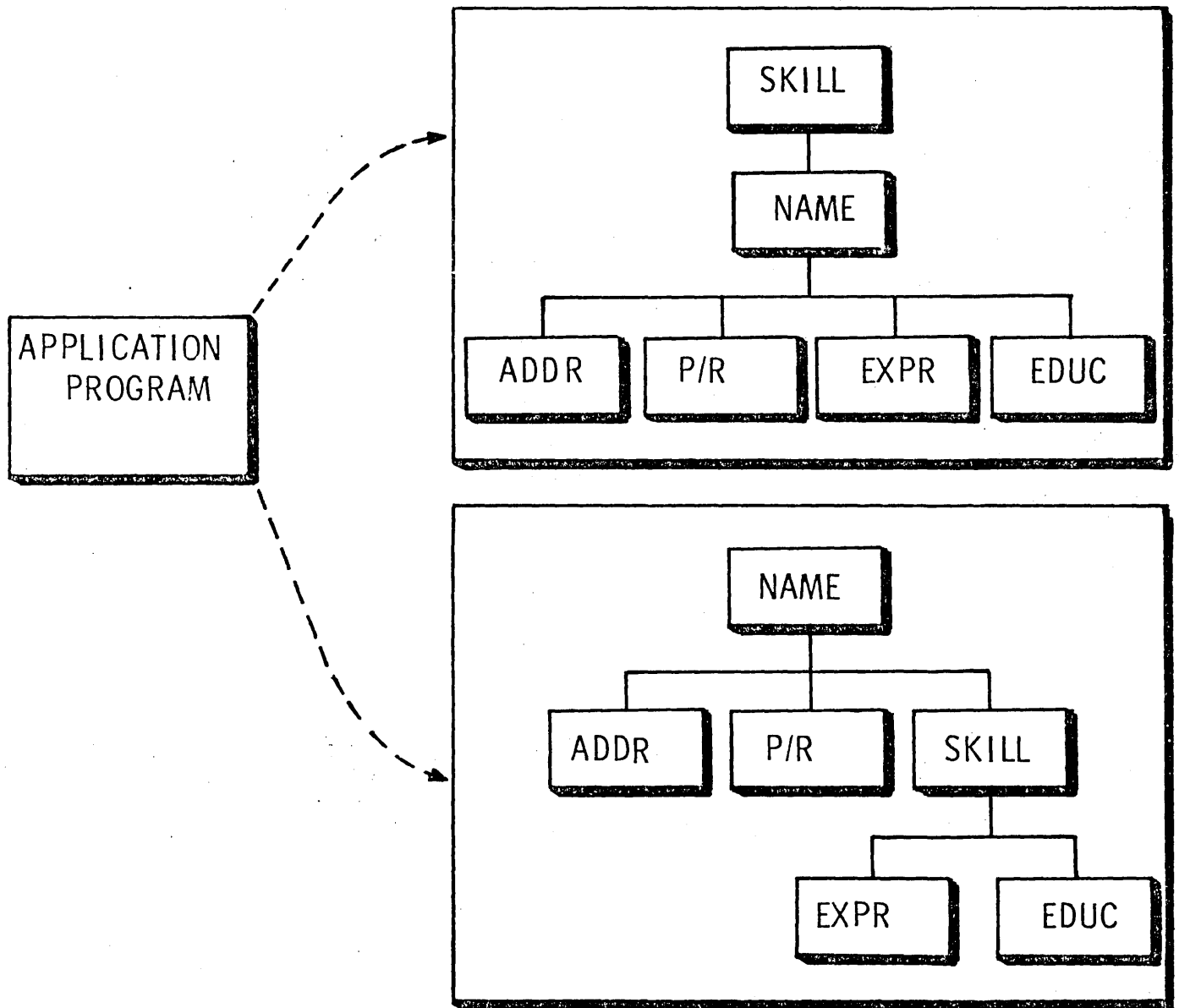
SKILLS INVENTORY DATA BASE



Logical Data Structure

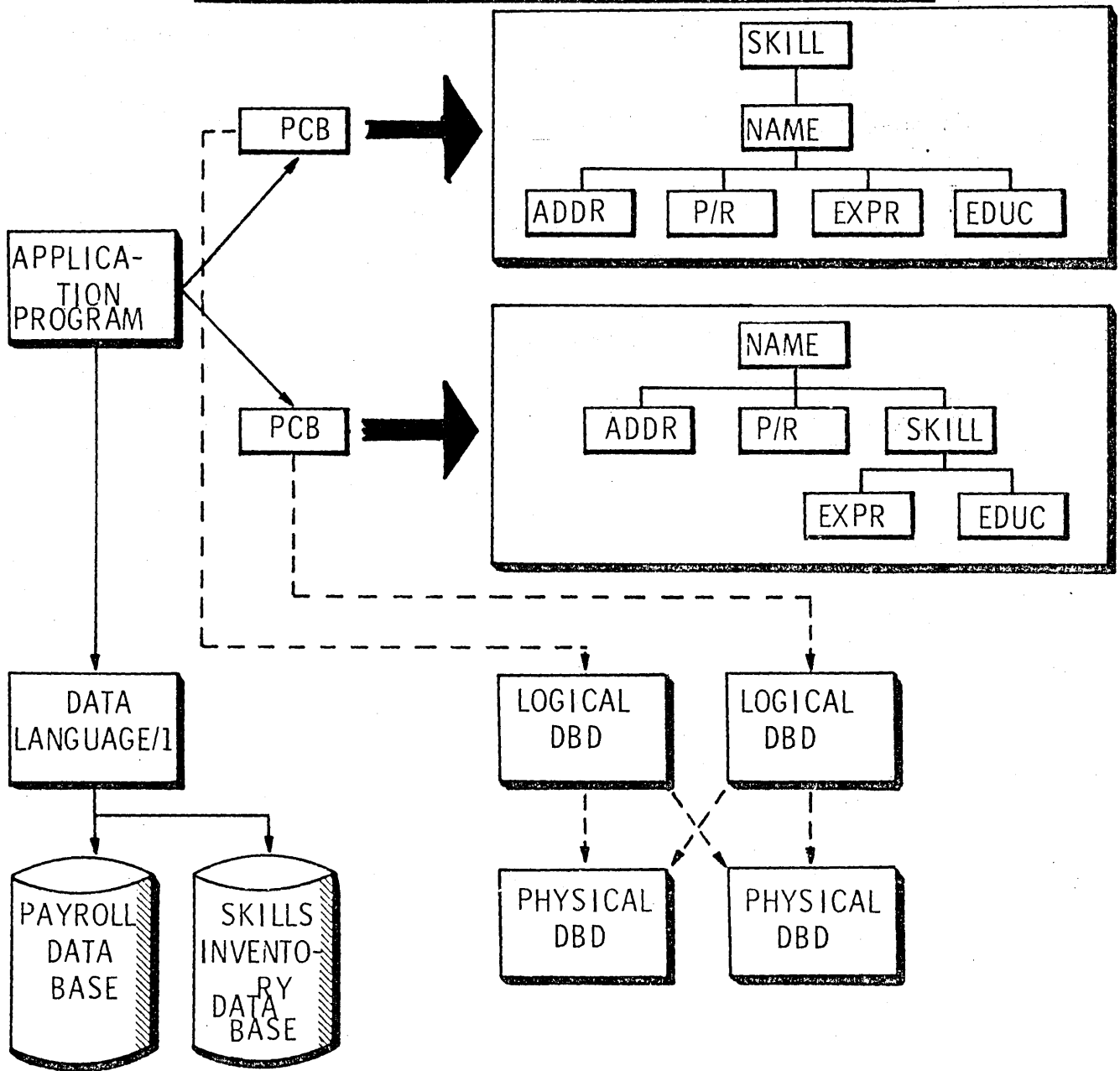


Two Views Of The Same Data



IBM-SP1-10

Two Views Of The Same Data



IMS/VS Secondary Indexing

PROVIDE ALTERNATE SEQUENCE CAPABILITY

PROVIDE ENTRY TO DATA BASE ON VALUES
OTHER THAN ROOT KEY

PROVIDE MULTIPLE ENTRY VALUES INTO A
COMMON DATA BASE

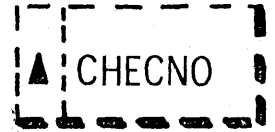
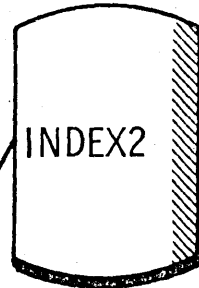
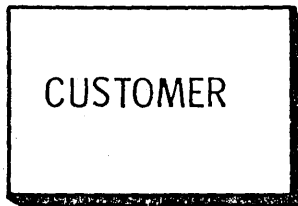
SECONDARY INDEXES ARE AUTOMATICALLY
MAINTAINED BY THE SYSTEM

IMS Secondary Indexing

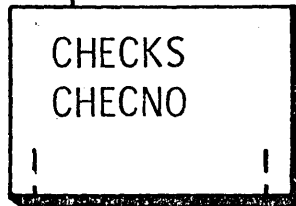
EXAMPLE

DATA BASE

INDEXED
SEGMENT



INDEX
SOURCE
SEGMENT



Index points to segment level to be retrieved, but source can be fixed in any segment which is descendant

CALL SEQUENCE

GU CUSTOMER (CHECKNO=56347)

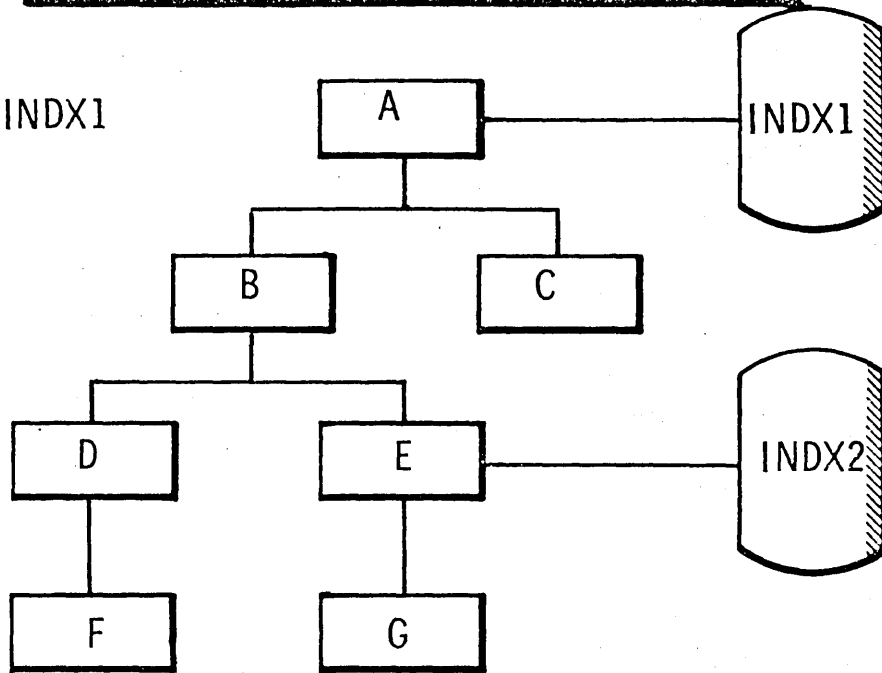
GNP

- INDEXED FIELD MAY BE USED IN SSA FOR INDEXED SEGMENT

IMS Secondary Indexing

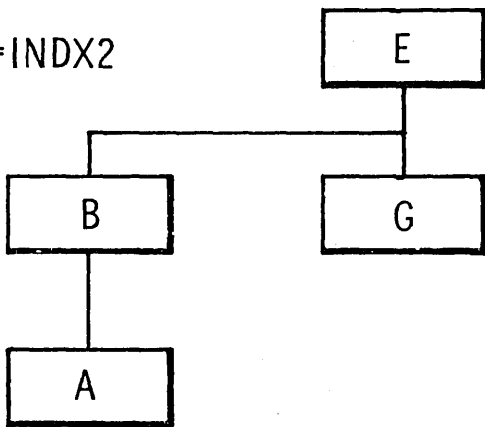
with PCB

PROCSEQ=INDX1



with PCB

PROCSEQ=INDX2



RESTRICTIONS

- E/B/A NOT LOGICAL CHILDREN
- DO NOT ISRT/DLET E/B/A
- DO NOT CHANGE KEY FIELDS

CHOOSE ROOT AS INDEXED SEGMENT IF YOU DO NOT WANT INVERSION

'PROCSEQ' DOES NOT RESTRICT USE OF OTHER SECONDARY INDEXES

Index base defined by Procseq value in PCB -

IMS/VS Secondary Indexing

CONSIDERATIONS

PRO:

SEQUENTIALLY PROCESS DATA IN A NON-KEY
FIELD SEQUENCE

RANDOM RETRIEVAL WHEN PRIMARY KEY NOT
KNOWN

SEQUENTIAL RETRIEVAL OF HDAM DATA BASE

EXCELLENT FOR PRODUCING REPORTS IN DIFFERENT
SEQUENCES

CHOICE OF EITHER PRIMARY OR SECONDARY
INDEXES AS MAIN PROCESSING SEQUENCE

CON:

EXTRA PROCESSING AND I/O ASSOCIATED WITH
THE MAINTENANCE OF SECONDARY INDEXES

DL/I TERMINOLOGY

HIERARCHY:

-DL/I data structure. A means of organizing data by relational (associative) attributes.

SEGMENT:

-An element of a hierarchy.
-The smallest unit of data which can be requested by an application program CALL.
-Consists of a prefix and a data portion.

SEGMENT TYPE/SEGMENT OCCURRENCE:

-A segment type is defined at DBDGEN (SEGM macro) time. It is an element of the structure.
-A segment occurrence is actual data loaded, stored and processed by IMS/VS application programs.
-There can be 0 to n segment occurrences of one segment type.
-There can be 1 to 255 different segment types defined for any data base.

ROOT SEGMENT:

-The first (top) segment of a data base hierarchy.
-There can be only 1 root segment type per data base.

PARENT/CHILD SEGMENTS:

-Parent/Child are terms used to define the hierarchical relationship between inter-dependent segments on two adjacent levels of a data base.
-A child is the dependent segment at the lower level. A child may have only one parent.
-A parent is the segment at the higher level. A parent may have 1 to n children.

SIBLING/TWIN SEGMENTS:

-Sibling/Twin are terms used to define the hierarchical relationship between segments which are children under the same parent segment type.
-All of the children of a particular parent segment occurrence have a twin relationship.
-Any segments having the same parent segment type but different parent segment occurrence can be called siblings.

POSITION:

- When a segment occurrence is retrieved or inserted, IMS/VS establishes a position on that occurrence and on all parent levels, if any, of that segment occurrence.
- IMS/VS maintains position (i.e. "knows where it is") in each data base an application program uses. If two A/Ps concurrently use the same DB, position is maintained for each.
- Users may optionally request IMS/VS to maintain position in each hierarchic path of a DB for an A/P.

DATA SET GROUP:

- HISAM, HDAM, and HIDAM data bases can be defined as consisting of from 1 to 10 data set groups, subject to rules regarding access methods, secondary indexing, and the use of the label field.

CONCATENATED KEY:

- For any segment, the concatenation of the key fields (including its own) of all of its successive parent segments up to and including the root.

SYMBOLIC POINTER:

- Concatenated Key.

SENSITIVITY:

- An application program is "sensitive" to those segments (of a data base) which it will access and process. Sensitivity is overtly declared via the "SENSEG" macro in PSBGEN.

PROCESSING INTENT:

- An application program declares, for each segment and/or data base the type of processing it intends (e.g. read-only, delete, replace, etc.)
- Processing intent is declared in both the PCB and SENSEG macros in PSBGEN.

INTENT PROPAGATION:

- Because of the associativity between segments of an IMS/VS data base, processing intent for one segment can propagate to related segments depending on the type of processing and the kind of relationship. IMS/VS takes care of intent propagation internally.

QUALIFIED/UNQUALIFIED CALLS:

- Application program calls to DL/I with or without segment search arguments (SSA's)

QUALIFIED/UNQUALIFIED SSAs:

- Segment search arguments with or without Boolean qualification statements.

FIELD:

- The data portion of any segment type may be defined as consisting of from 0 to 255 fields.
- Applications address a segment type by segment name; they address a specific segment occurrence by a field name and a search argument.
- Fields are defined at DBDGEN time via the FIELD macro.
- Up to 1000 fields.

KEY FIELD/DATA FIELD:

- In general, one field per segment type may be declared a key (or sequence) field. Key fields are used by IMS/VS to store segment occurrences in sequential ascending order. Non-unique occurrences of key fields are permitted for non-root segment occurrences of the same segment type having different parent segment occurrences.
- Key fields and the unique/non-unique attribute are defined in the DBDGEN FIELD macro.
- Non-key fields are called data fields.
- The successive vertical dependencies of a hierarchic structure are called "levels".
- A hierarchic data base structure may contain up to 15 levels.

LEVEL:

DATA BASE RECORD:

- A DBR consists of 1 Root Segment occurrence and all its dependents.
- A DBR can be equal to or greater than 1 Access Method Logical Record.

PATH:

- The sequence of segment occurrences, only one occurrence per level, leading from a Root Segment occurrence to a desired segment occurrence.
- There exists only one path to any segment.

HIERARCHIC SEQUENCE:

- By convention, IMS/VS traverses a hierarchic structure in top-to-bottom, front-to-back, left-to-right sequence. At every position, it seeks a lower level; if none exists, it seeks the next occurrence of the same segment type; if none exists, it seeks the first occurrence of the next-right segment type on the same level; if none exists it seeks, in the level immediately above, the element which is behind (or next-right to) the last element it had reached earlier at that level.

IMS / VS UTILITIES

- GENERATION

DBDGEN PSBGEN ACBGEN ←

Generates and saves control blocks built by combination of DBDGEN & PSBGEN

- DATA BASE

REORGANIZATION
RECOVERY

UTILITY CONTROL FACILITY ←

Generates JCL for utility jobs and then executes.

- LOGGING

LOG RECOVERY
LOG TERMINATOR
STATISTICS

- OTHERS

DB & DC MONITORS
PI TRACE
SECURITY MAINTENANCE
MESSAGE FORMAT LANGUAGE

PROGRAM ISOLATION (PI) →

~~DATA BASE UTILITIES~~



REORGANIZATION/LOAD PROCESSING:

PHYSICAL -

HISAM REORGANIZATION UNLOAD (DFSURULØ)

HISAM REORGANIZATION RELOAD (DFSURRLØ)

HD REORGANIZATION UNLOAD (DFSURGUØ)

HD REORGANIZATION RELOAD (DFSURGLØ)

LOGICAL -

DATA BASE PREREORGANIZATION (DFSURPRØ)

DATA BASE SCAN (DFSURGSØ)

DATA BASE PREFIX RESOLUTION (DFSURG1Ø)

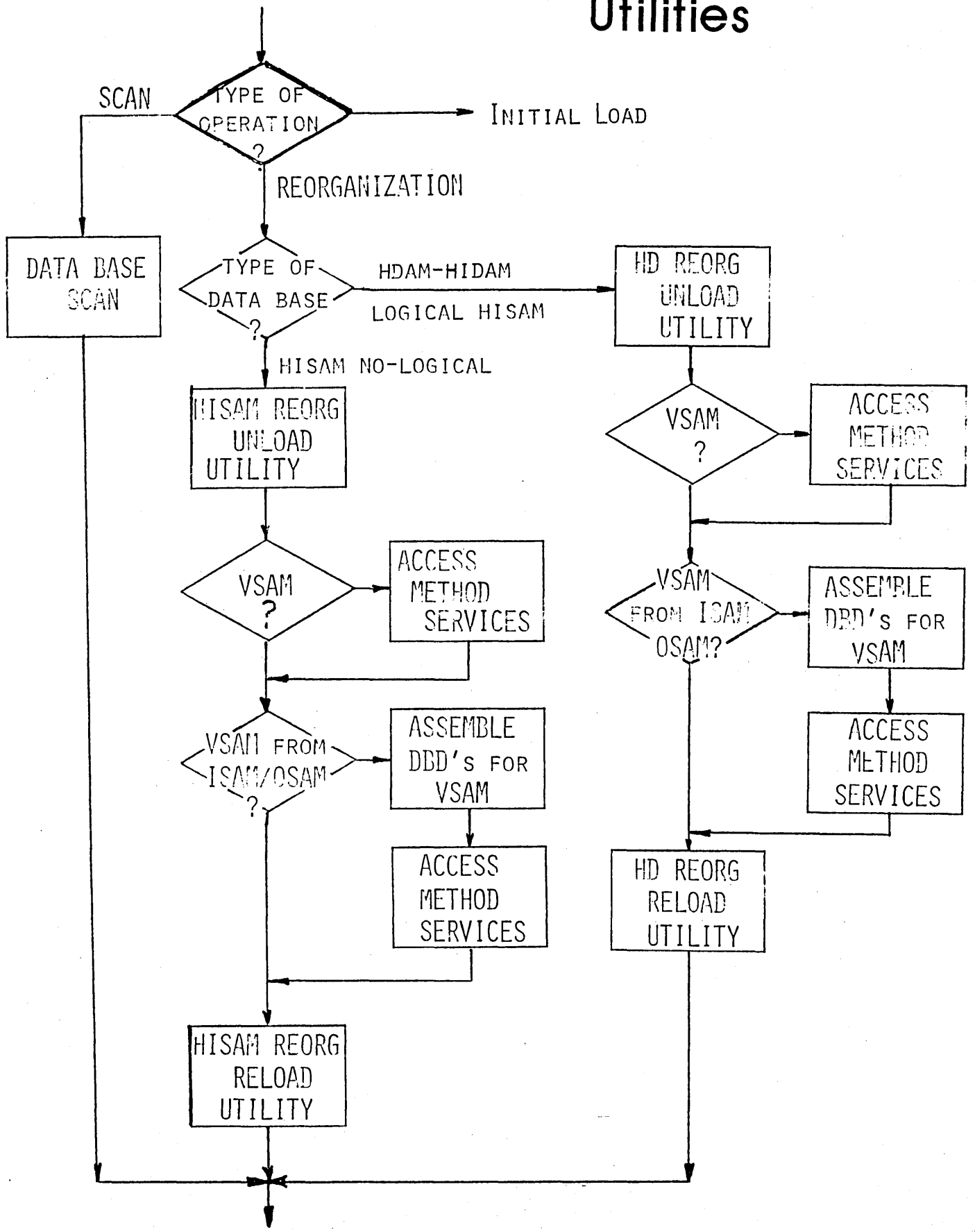
DATA BASE PREFIX UPDATE (DFSURGPØ)

(REF: U.R.M. CHAPTER 4)

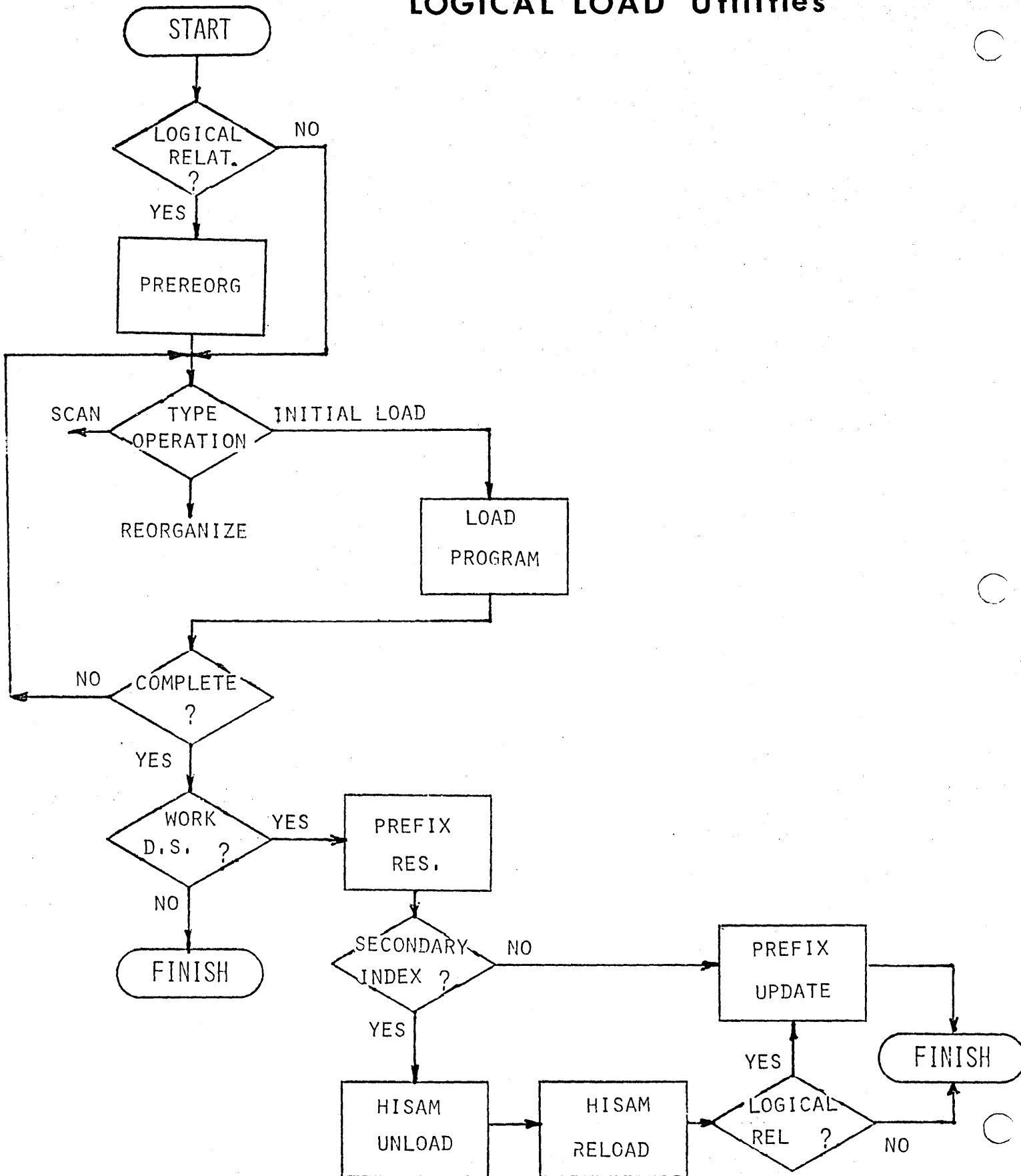
WHY REORGANIZE?

- RECOVER SPACE
- CHANGE BLKSIZE
- CHANGE LRECL
- CHANGE ACCESS TECHNIQUES
- ADD LOGICAL RELATIONSHIPS
- STRUCTURE CHANGE

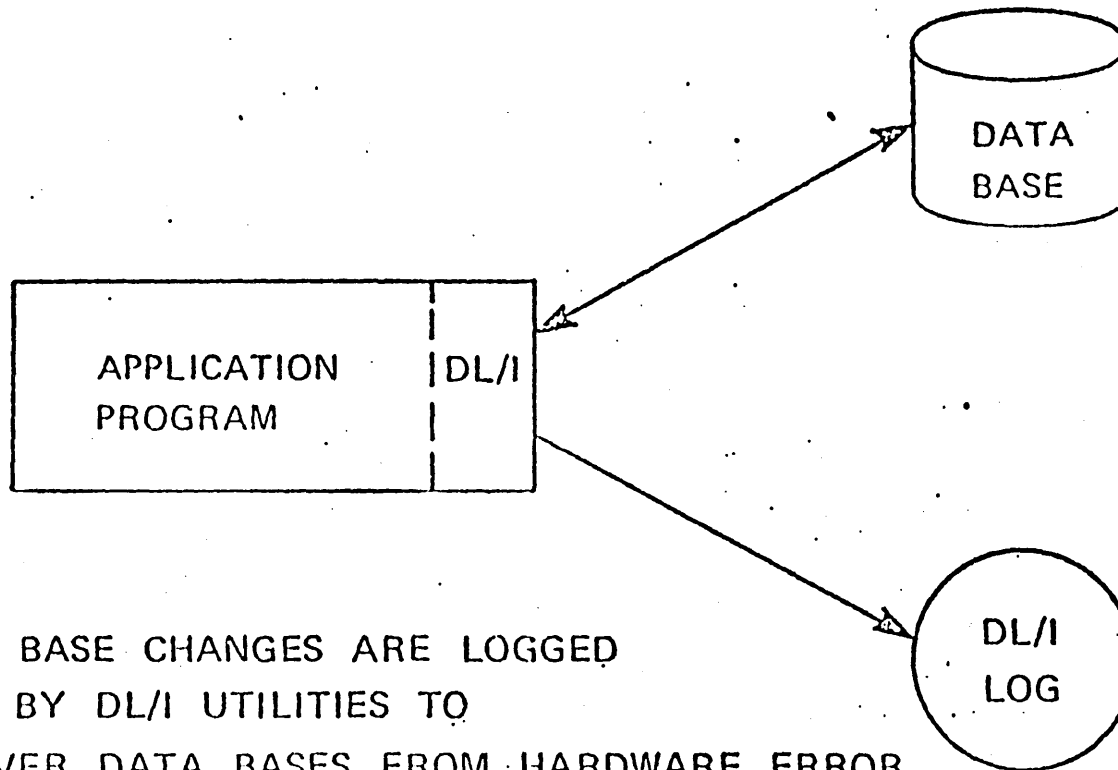
REORGANIZATION Utilities



LOGICAL LOAD Utilities

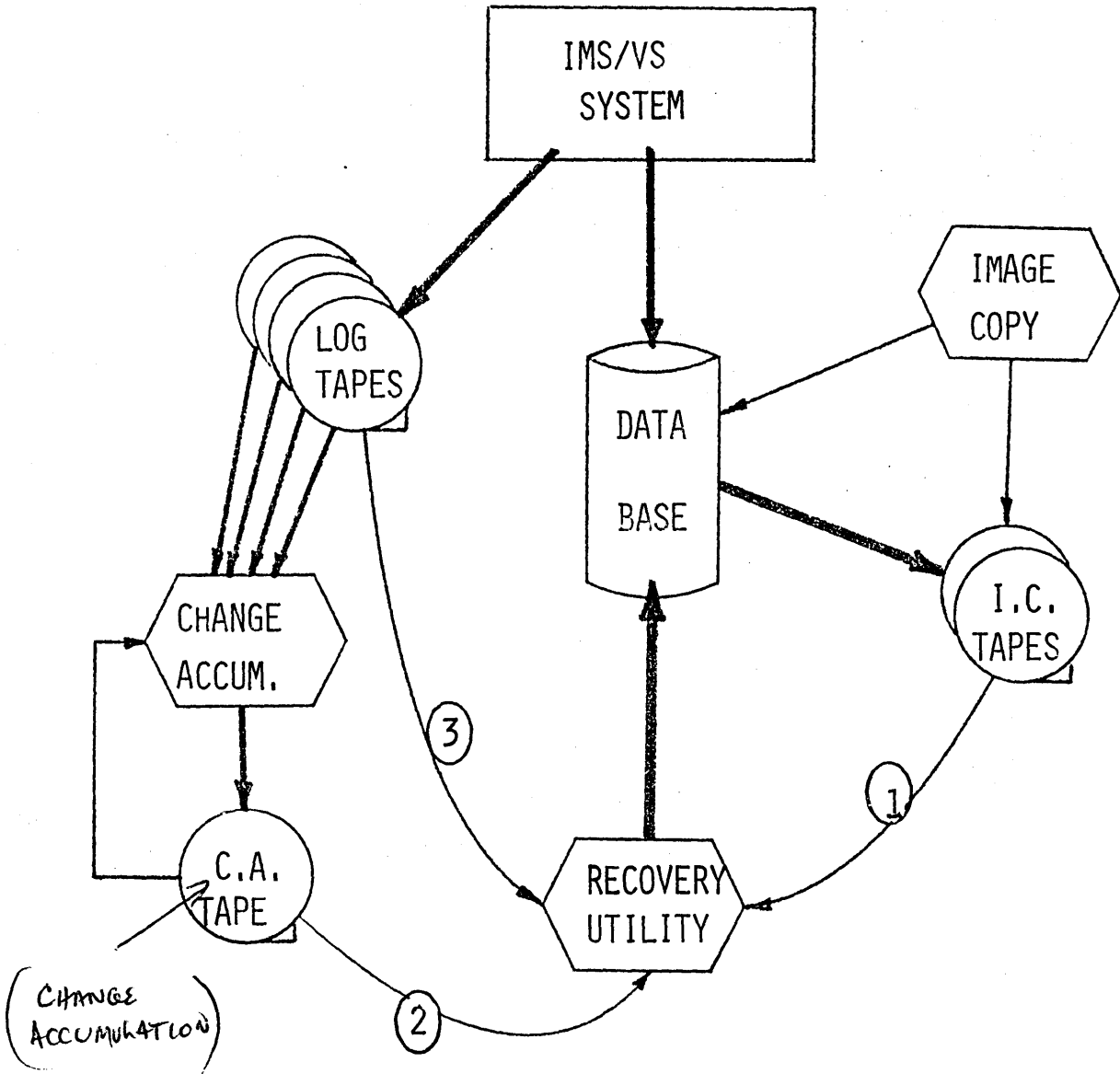


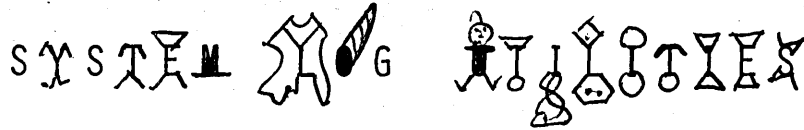
DATA RECOVERY PROBLEM



- ALL DATA BASE CHANGES ARE LOGGED
- LOG USED BY DL/I UTILITIES TO
 1. RECOVER DATA BASES FROM HARDWARE ERROR
 2. BACK-OUT UPDATES MADE BY PROGRAM ERROR

DATA BASE RECOVERY





I. LOG MAINTENANCE

- LOG RECOVERY UTILITY (DFSULTR0)
- LOG TERMINATOR UTILITY (DFSFL0T0)

II. LOG DATA FORMATTING

- STATISTICAL ANALYSIS UTILITY (DFSISTxx) *
- FILE SELECT & FORMATTING PRINT UTILITY
(DFSERA10) (DFSERA30)
- LOG TRANSACTION ANALYSIS UTILITY
(DFSILTA0)

(REF: U.R.M. CHAPTERS 7 & 8)

* FOUR MODULES - xx = S0, 20, 30, 40

OTHER UTILITIES

I. PERFORMANCE (U.R.M. CHAPTER 9)

- DB MONITOR REPORT PRINT PROGRAM
- DC MONITOR REPORT PRINT PROGRAM
- P.I. TRACE REPORT UTILITY PROGRAM

II. SECURITY (INST. GUIDE STEP 12)

- SECURITY MAINTENANCE PROGRAM

III. MESSAGE FORMAT SERVICE (MFS USERS GUIDE)

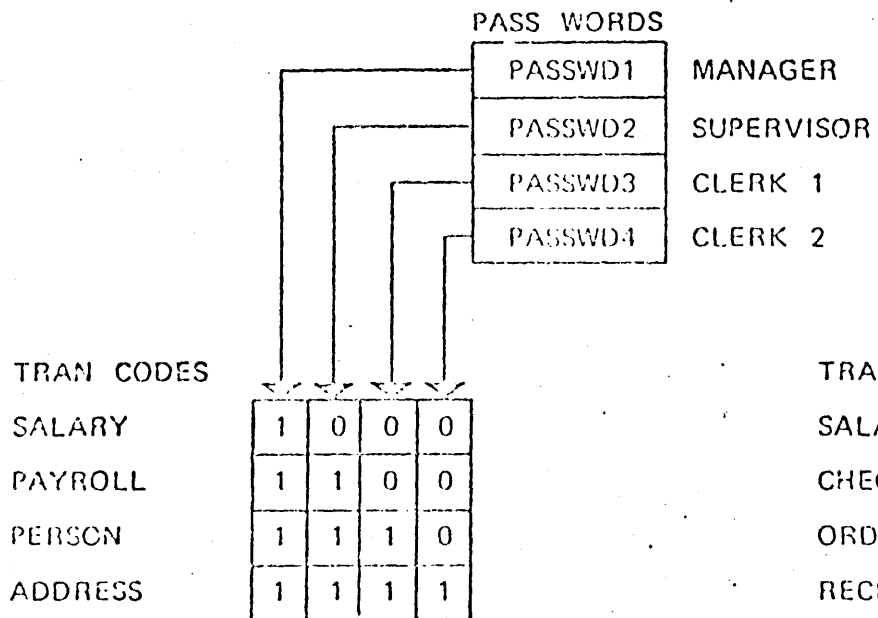
- MESSAGE/FORMAT LANGUAGE UTILITY
- MESSAGE/FORMAT SERVICE UTILITY

SECURITY MAINTENANCE PROGRAM

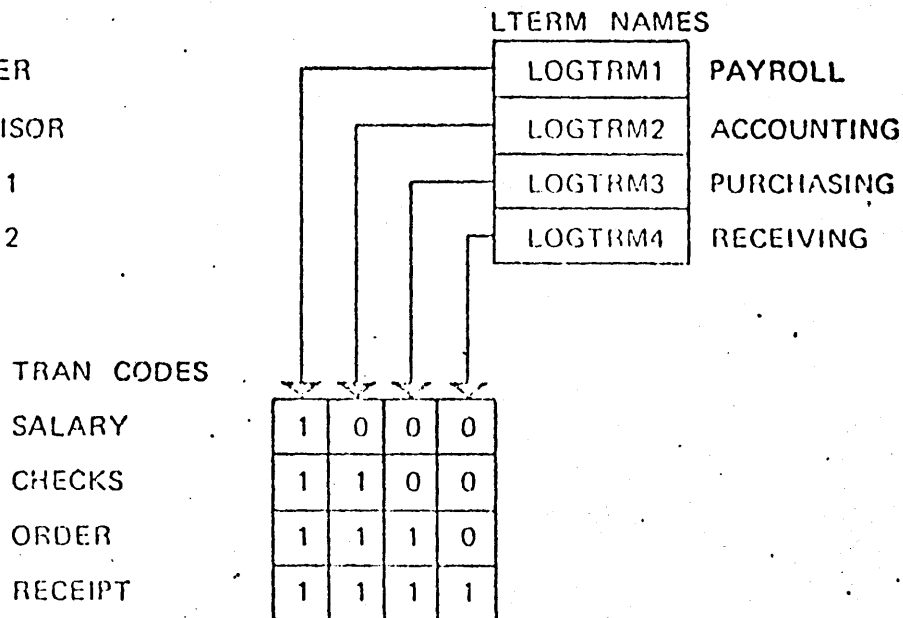
PASSWORD	LTERM	MODIFYING STATUS (/LOCK,/UNLOCK,/IAM)
PASSWORD	TRANSACT	} REQUIRES PASSWORD W/EACH MSG. ENTERED
PASSWORD	COMMAND	
PASSWORD	DATA BASE	} MODIFYING STATUS (/LOCK,/UNLOCK,/IAM
PASSWORD	PROGRAM	
PASSWRD	PTERM	
TRANSACT	TERMINAL	} RESTRICT USE OF MSG TO SPECIFIC LTERM
COMMAND	TERMINAL	

SECURITY

PASSWORD SECURITY



TERMINAL SECURITY



UTILITY CONTROL FACILITY

- CONTROL CARD DRIVEN PROGRAM
- UTILITY PROGRAMS EXECUTE UNDER U.C.F.
- RESTART CAPABILITY
- USER EXITS SUPPLIED
- FUNCTION IS TO PROTECT THE USER
- DETAILS IN U.R.M., CHAPTER 6

C

C

C

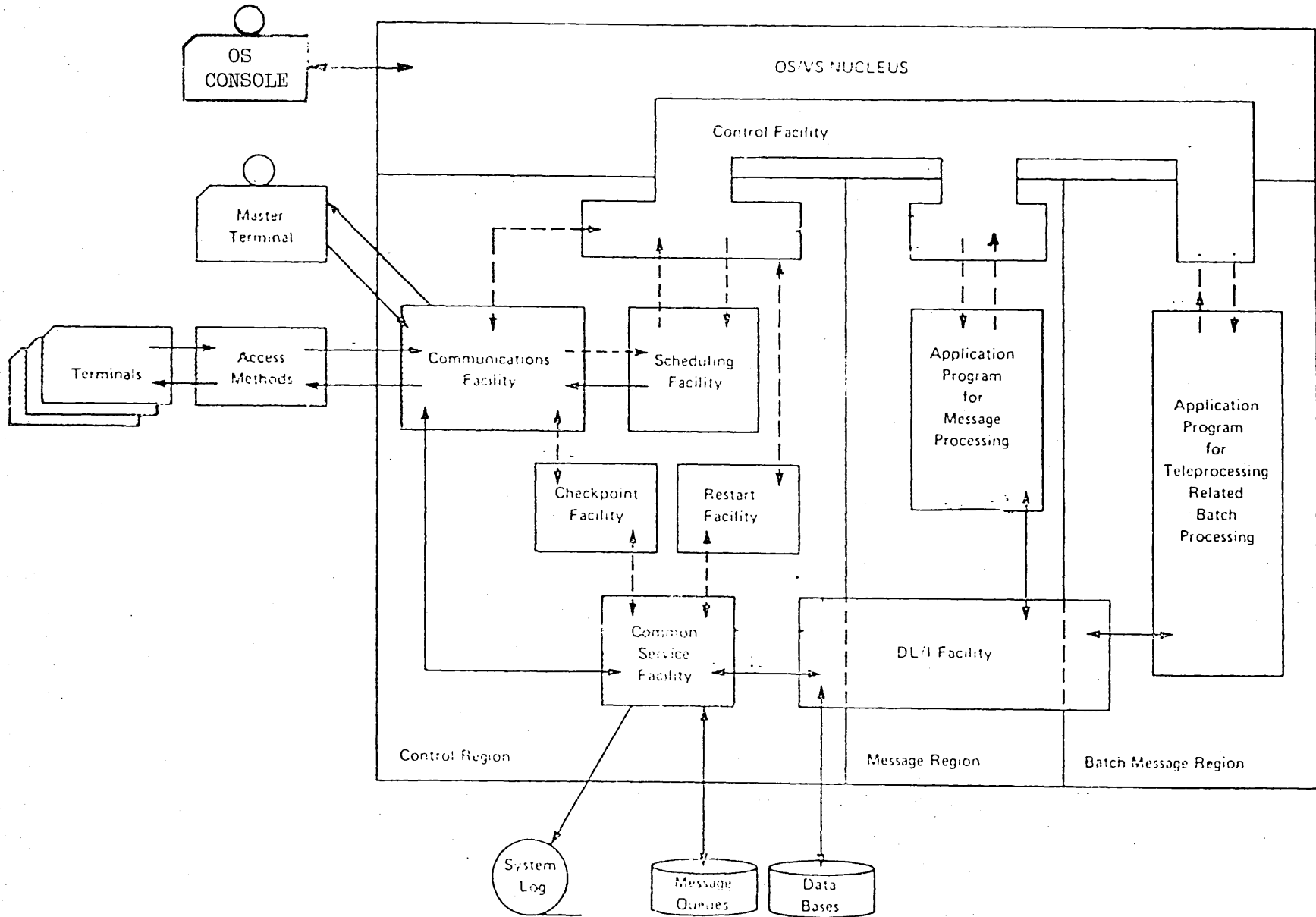
DC facilities

C

C

C

IMS/VS D.C. SYSTEM OVERVIEW



TERMINALS SUPPORTED

1050

2260-1&2

2265

2740-1&2

2741

2770

2780

2980

3270

3600

3770

3790

7770-3

MODEL 33/35 TELETYPEWRITER

SYSTEM/3 & SYSTEM/7

LOCAL CARD READER

LOCAL PRINT - PRINTER

PUNCH

DISK

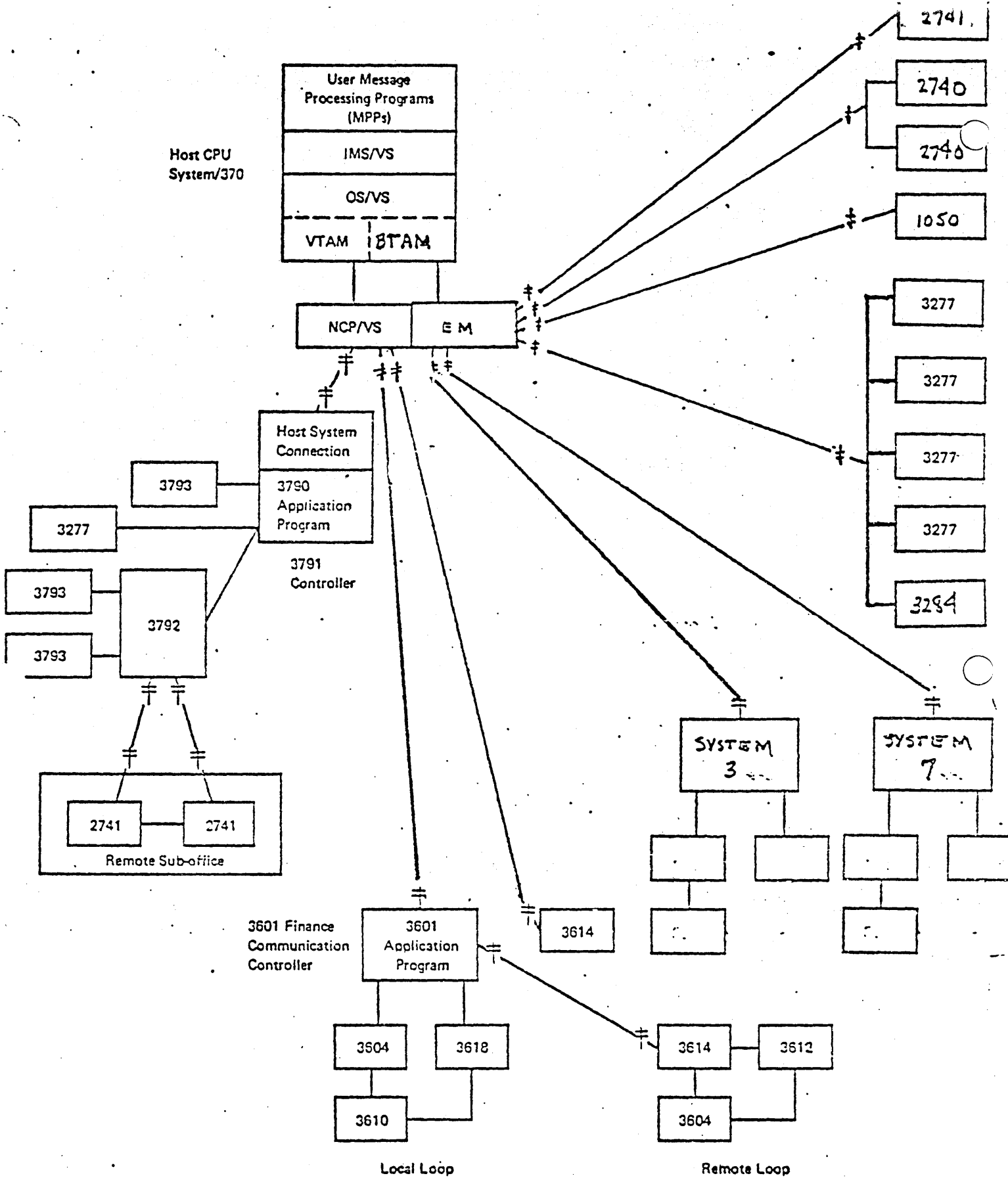
TAPE

SPOOL (SYSOUT)

IMS/VS

System/3 and System/7 Support

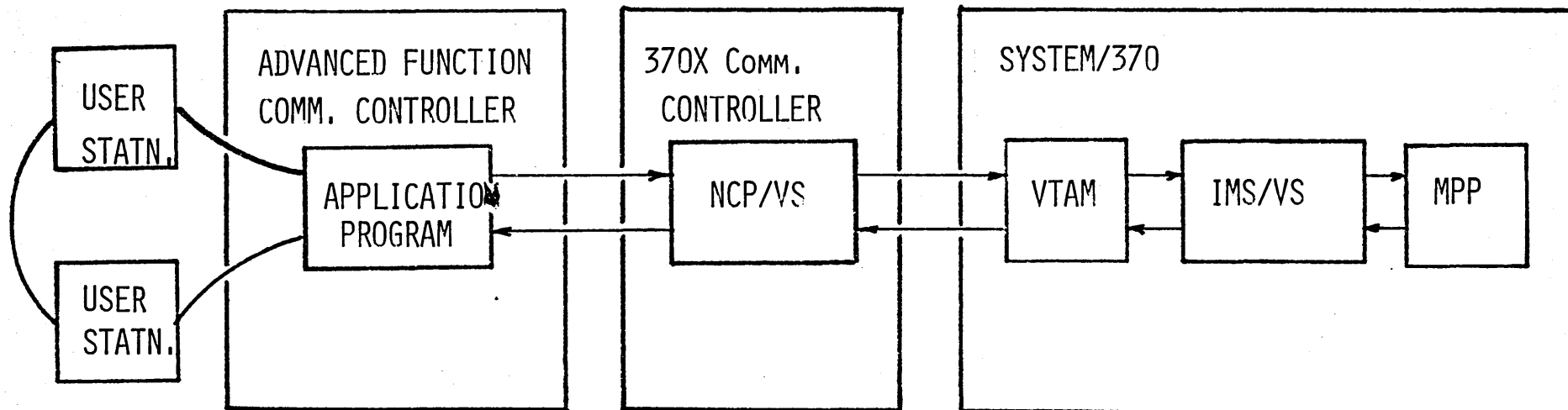
- THESE SYSTEMS BELONG TO A CLASS OF DEVICES CALLED INTELLIGENT TERMINALS.
- AS INTELLIGENT TERMINALS, THEY CAN ACT AS EITHER TERMINALS OR REMOTE CONCENTRATORS.
- A STANDARD TRANSMISSION AND CONTROL FORMAT HAS BEEN DEFINED SUCH THAT THEIR DATA HANDLING CHARACTERISTICS ARE SIMILAR, BUT THE LINE CONTROL IS DIFFERENT.
- ALL ATTACHED TERMINALS WILL BE KNOWN TO IMS.
- TERMINAL ID WILL BE VERIFIED BY IMS.



Local Loop

Remote Loop

ADVANCED FUNCTION COMMUNICATIONS SYSTEMS



(REF: IMS/VS AFC MANUAL)

ADVANCED FUNCTION COMMUNICATIONS SYSTEMS (CONT.)

APPLICATION PROGRAM

- READ DATA FROM TERMINAL
- WRITE DATA TO CPU
- READ DATA FROM CPU
- WRITE DATA TO TERMINAL

NCP/VS

- ASSEMBLE & DISS. CHARACTERS
- RECOGNIZE CTL. CHARACTERS
- CHECK & RECORD ERRORS
- CONTROL POLLING & ADDRESS'N.

IMS/VS

- SECURITY CHECK XACTIONS
- SCHEDULE PROPER MPP
- DIRECT DATA TO PROPER STATN.
- PROVIDE CHECKPOINT AND RECOVERY CAPABILITIES

COMMUNICATIONS CONTROLLER

- ADD LINE CTL. CHARACTERS
- RESPOND TO POLL
- TRANSMIT DATA
- RECEIVE DATA
- REMOVE LINE CTL. CHARACTERS
- CHECK FOR XMIT. ERRORS
- CONTROL BUFFERING

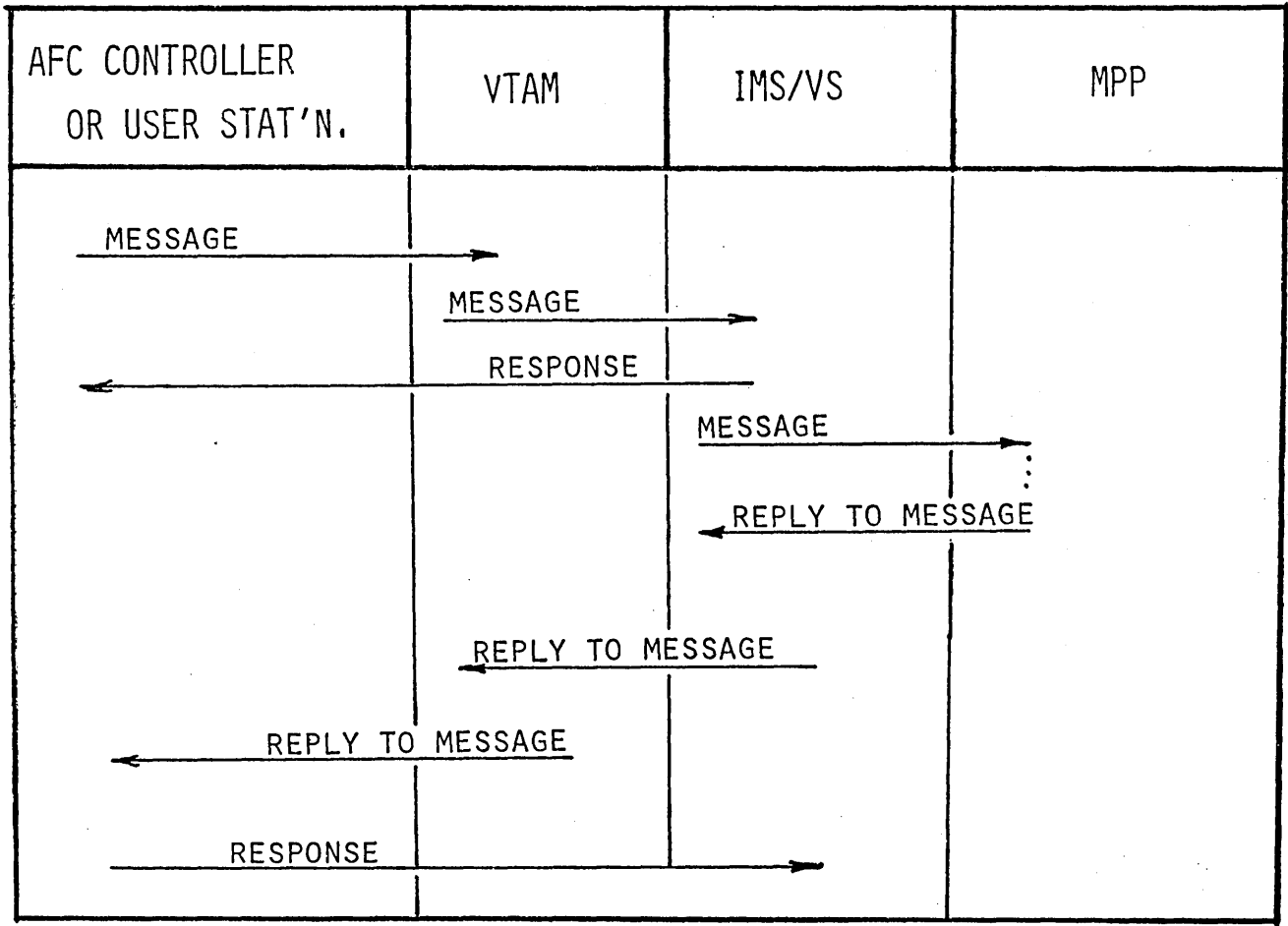
VTAM

- CONNECT & DISCONNECT STATIONS FROM NETWORK
- DIRECT DATA FROM STATION TO IMS/VS
- PERMIT MONITORING AND ALTERING NETWORK
- DIRECT DATA FROM IMS/VS TO PROPER STATION

MPP

- PROCESS INPUT XACTIONS
- PERFORM DB PROCESSING
- PRODUCE REPLY

AFC MESSAGE FLOW

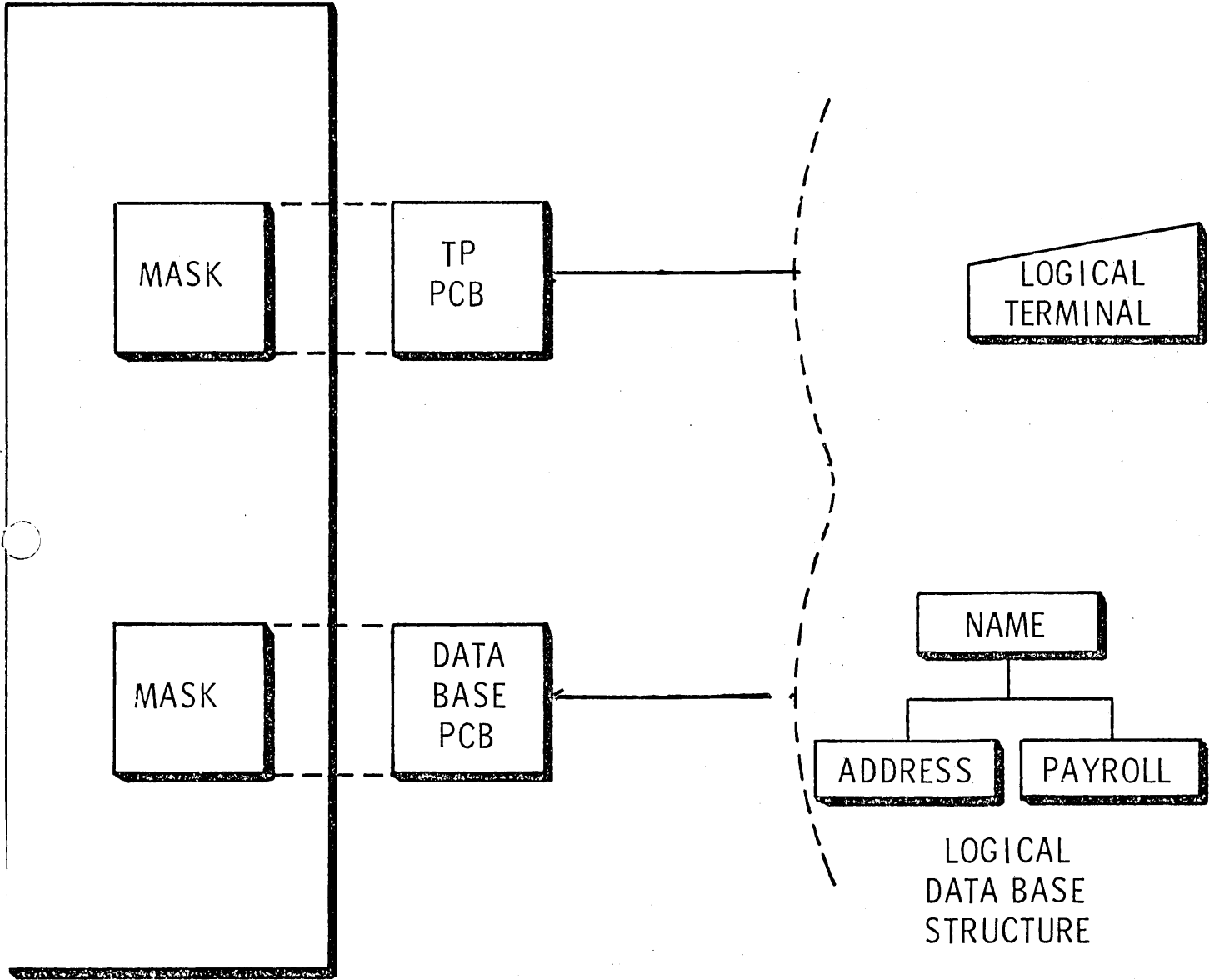


Data Communications Facilities

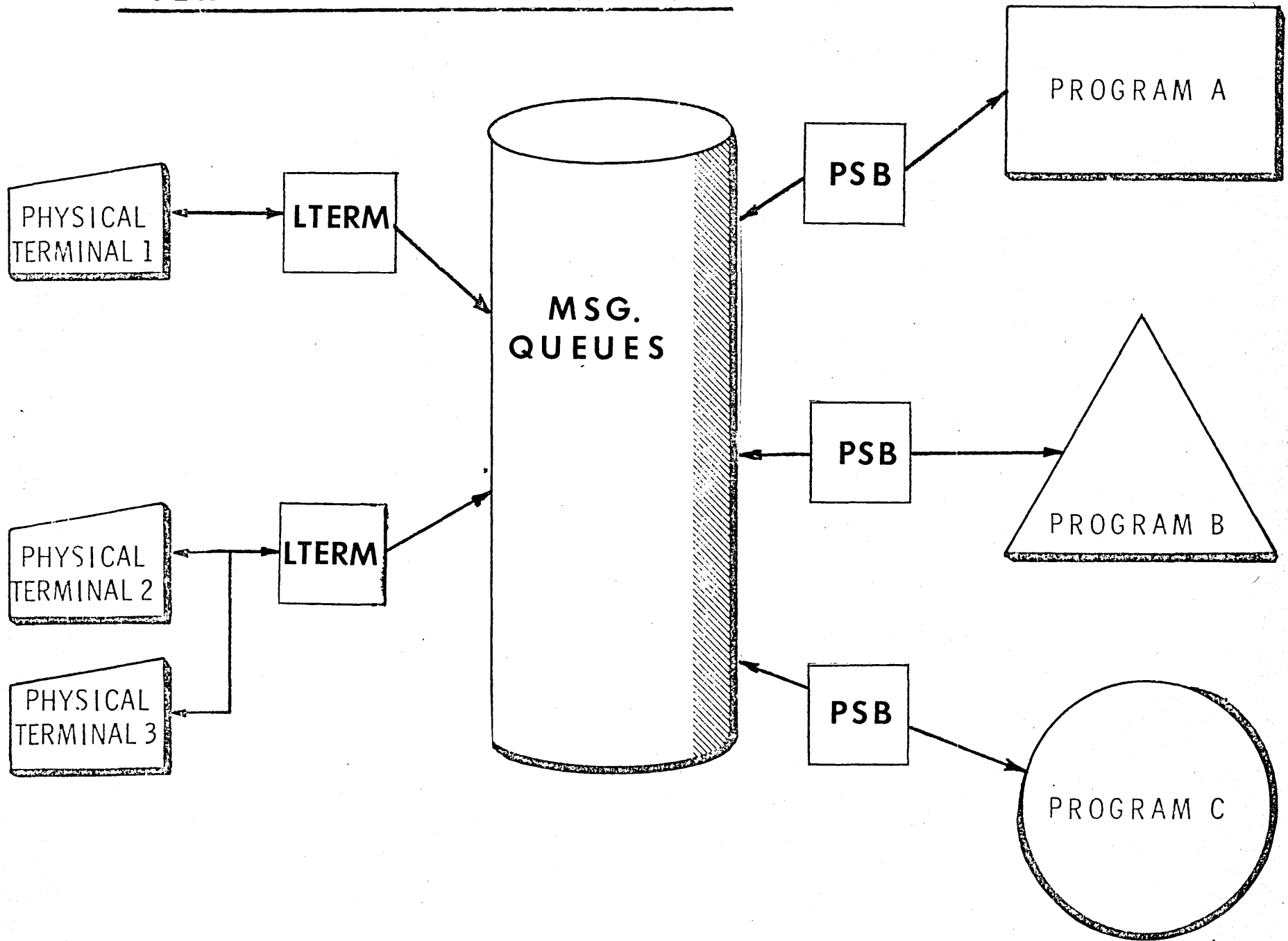
1. PHYSICAL TERMINAL SUPPORT
2. CENTRALIZED SYSTEM CONTROL
 - MASTER TERMINAL
3. LOGICAL TERMINAL OPERATION
 - TERMINAL INDEPENDENCE
 - SERVICEABILITY
4. MESSAGE PROCESSING
 - HIGH LEVEL LANGUAGES
 - PROGRAM SCHEDULING
 - CONVERSATIONAL PROCESSING
5. SECURITY AND PRIVACY
 - FOR MESSAGE ENTRY
6. SYSTEM COMMAND SET
 - CONTROL/INFORMATION
7. RELIABILITY/SERVICEABILITY
 - SYSTEM RECOVERY
 - DATA RECOVERY

Logical Structures and Logical Terminals

APPLICATION PROGRAM

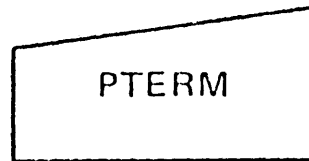


TERMINAL INDEPENDENCE

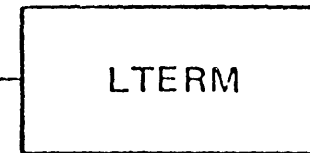


PHYSICAL/LOGICAL TERMINAL CONCEPT

PHYSICAL TERMINAL



LOGICAL TERMINAL



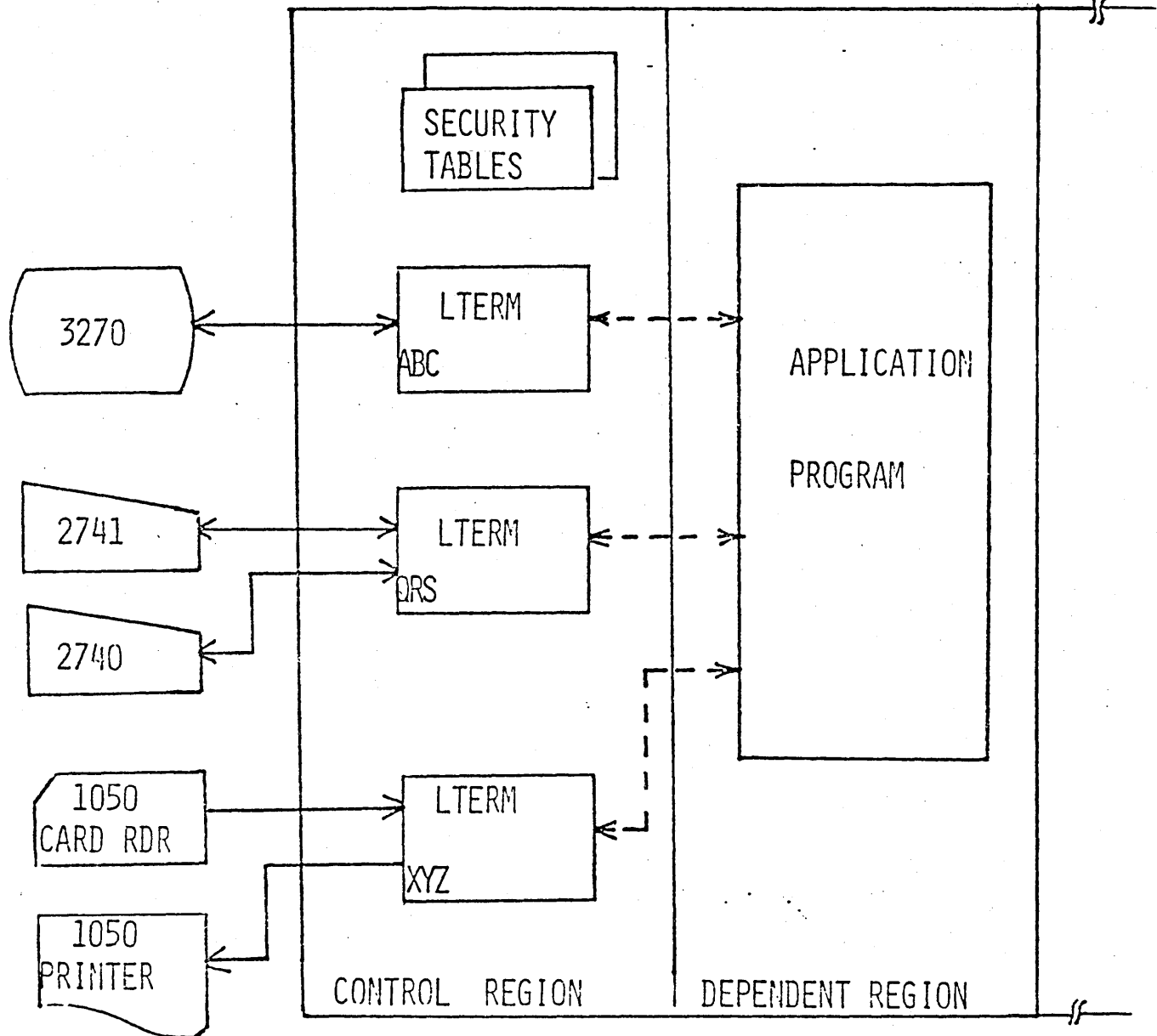
- REPRESENTS A PHYSICAL TERMINAL
- NUMBERED BY LINE
- HAS HARDWARE ADDRESS
- REPRESENTS A TERMINAL BY A SYMBOLIC NAME
- IMS INTERNAL DESTINATION
- ONE OR MORE NAMES PER PTERM
- ONE TO EIGHT CHARACTER NAMES
- USED BY APPLICATION PROGRAMS

PTERM/LTERM RELATIONSHIP PROVIDES:

- DEVICE INDEPENDENCE
- INPUT/OUTPUT SEPARATION IF DESIRED
- SECURITY

LOGICAL TERMINAL CONCEPT

IMS/VS



- ALL PHYSICAL TERMINALS ARE ASSIGNED A LOGICAL TERMINAL NAME
- APPLICATION PROGRAMS ACCESS TERMINALS BY LTERM NAME INDEPENDENT OF:
 - TERMINAL ADDRESS
 - LINE DISCIPLINE
 - DEVICE CHARACTERISTICS
- INTEGRAL WITH TERMINAL SECURITY

LTERM

PTERM

	1	2	3	4	5	6	...
A	X						
B		X					
C		X					
D			X				
E				X			
F				X			
G				X			
H					X		
I						X	
:							

IMS/VS SYSTEM DEFINITION MACROS

• TERMINAL

• NAME

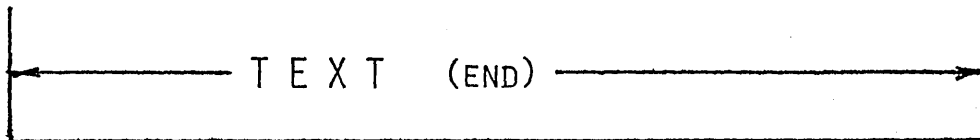
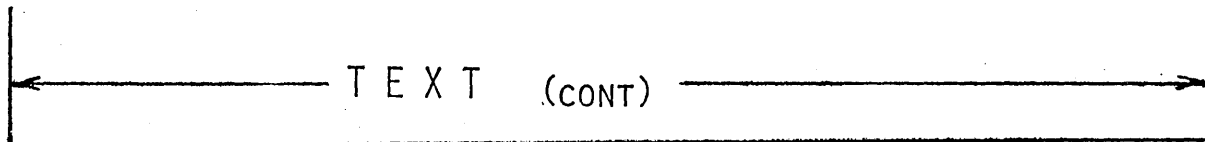
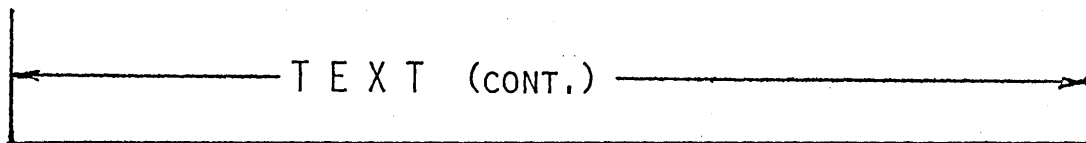
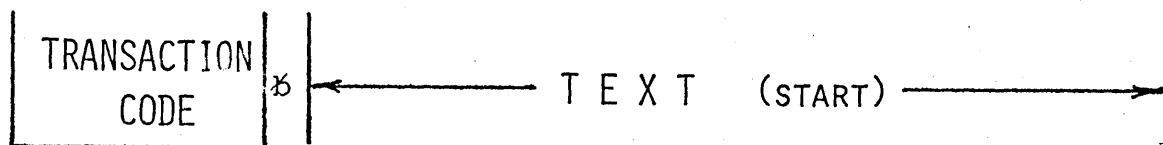
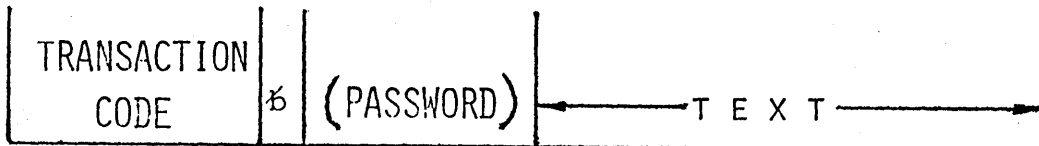
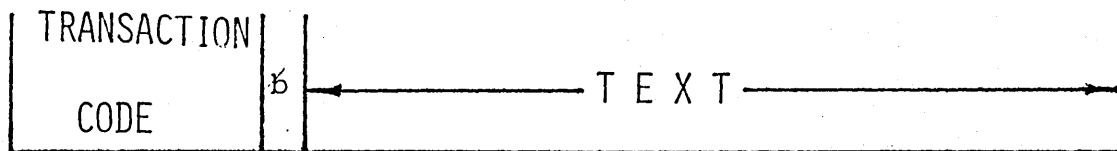
Input Message Format



A TRANSACTION IS:

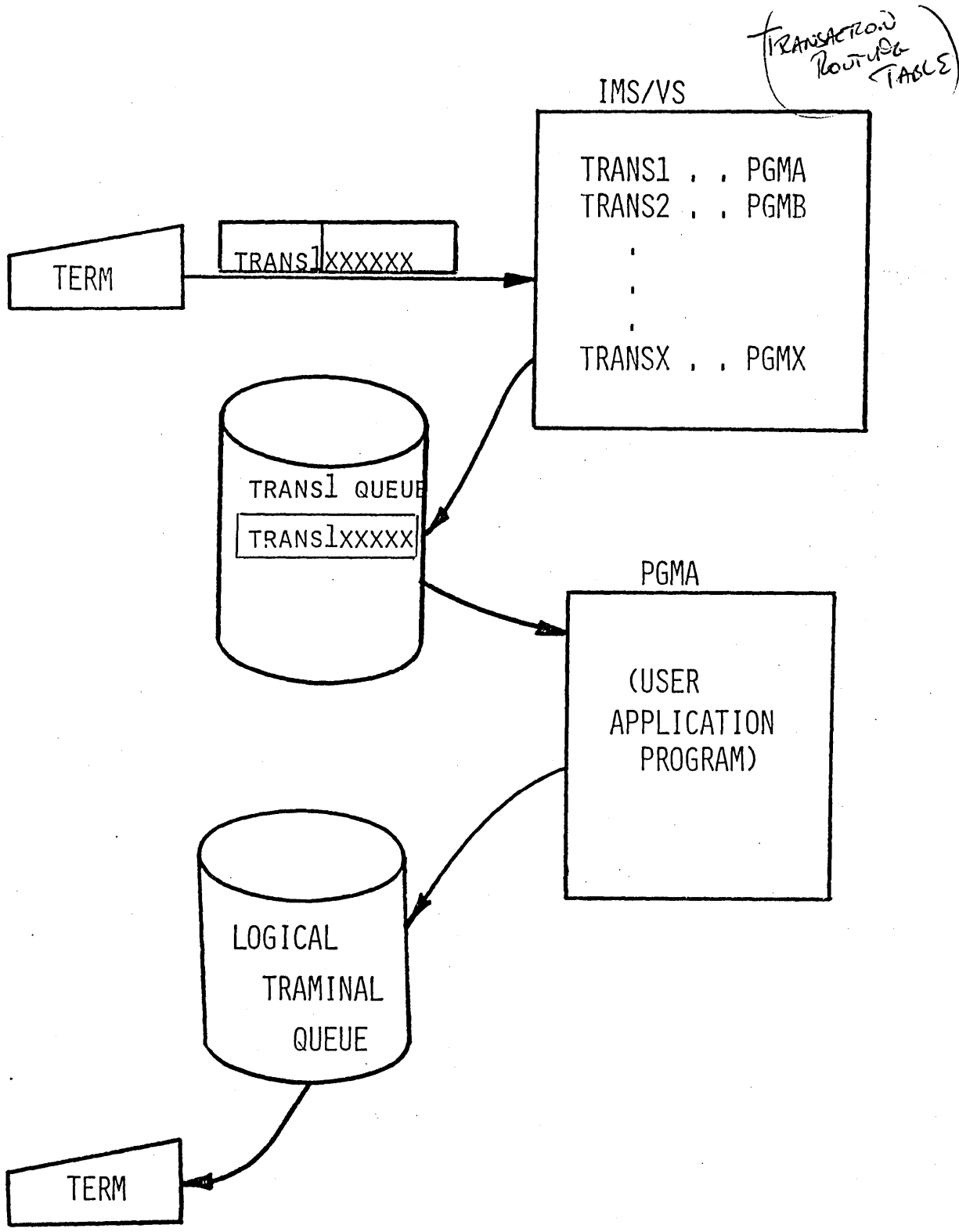
- A MESSAGE DESTINED FOR
AN APPLICATION PROGRAM
- A UNIT OF WORK TO BE
SCHEDULED FOR PROCESSING
- A RECOVERABLE UNIT
OF WORK

INPUT MESSAGE - TRANSACTION

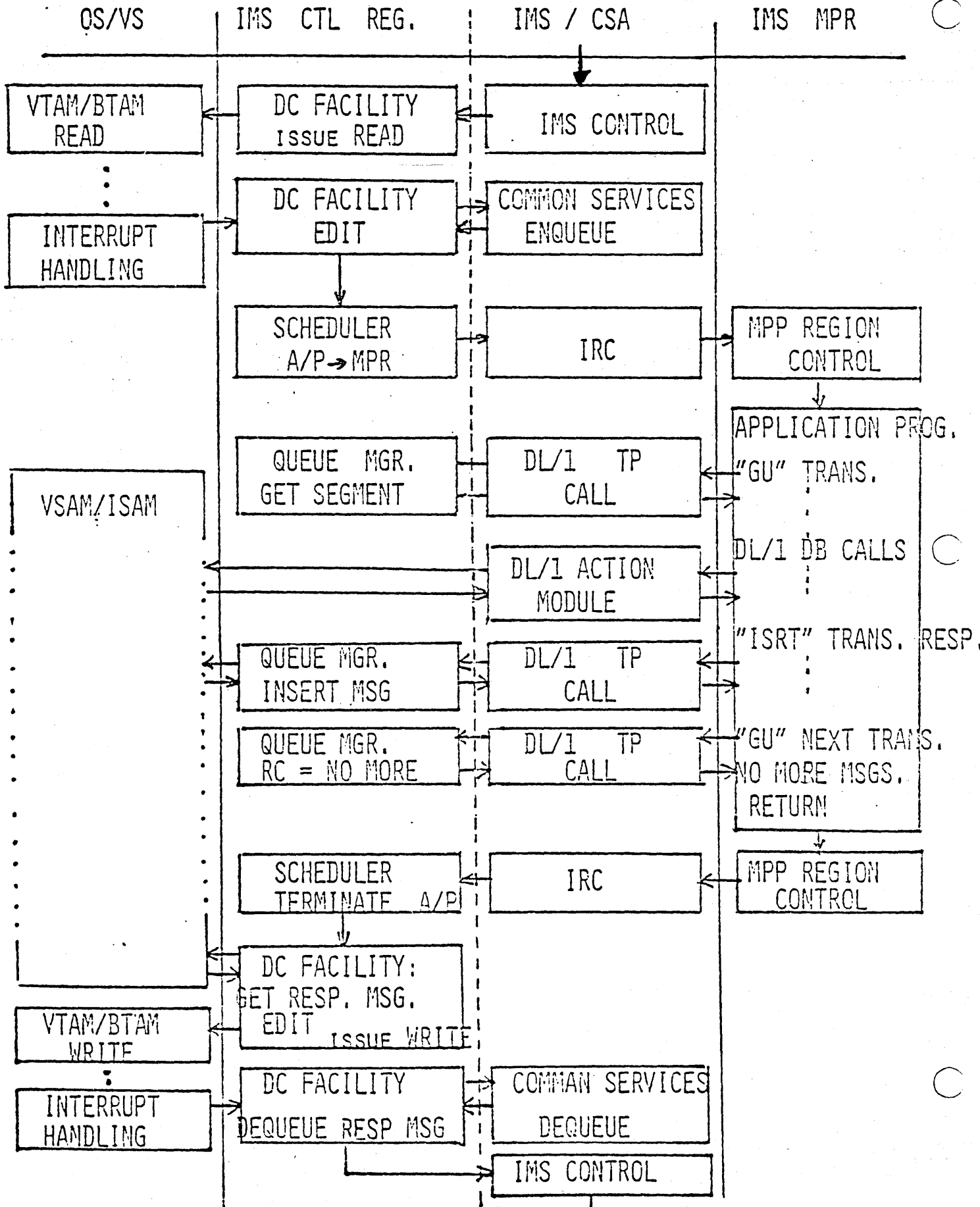


MULTI-SEGMENT MESSAGE

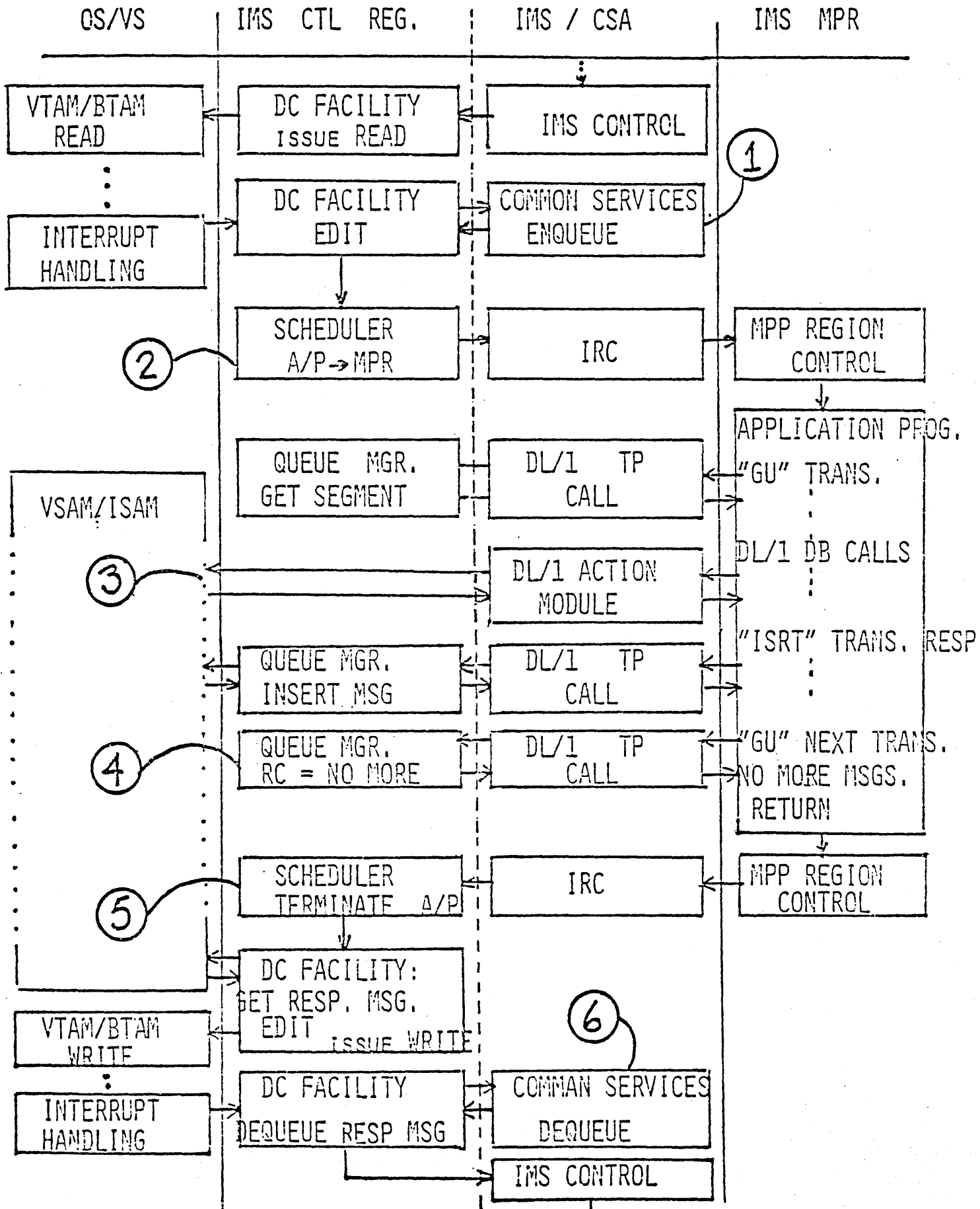
TRANSACTION FLOW



TRANSACTION - CONTROL FLOW



TRANSACTION - UNIT OF RECOVERY



Input Message Format



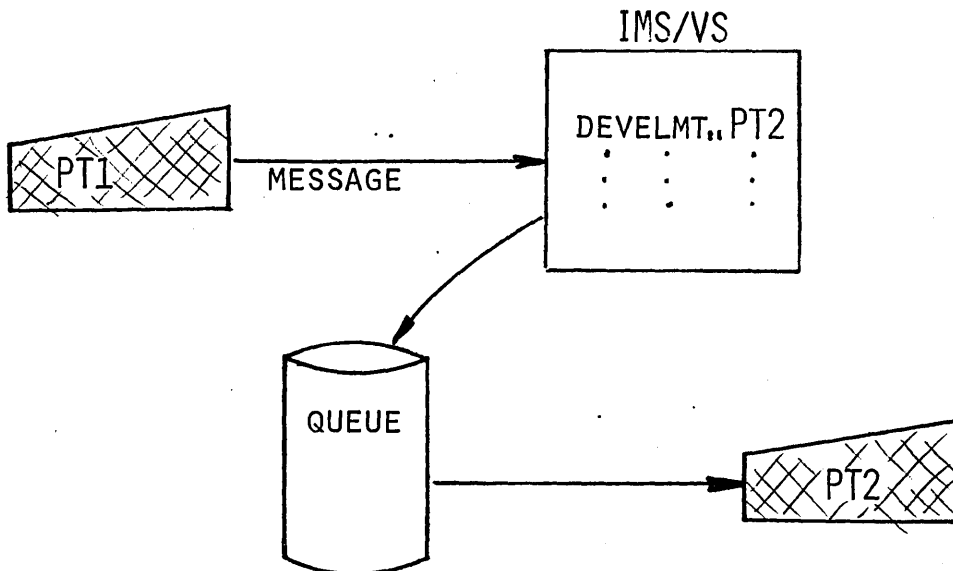
MESSAGE SWITCHING

MESSAGE FORMAT :



EXAMPLE :

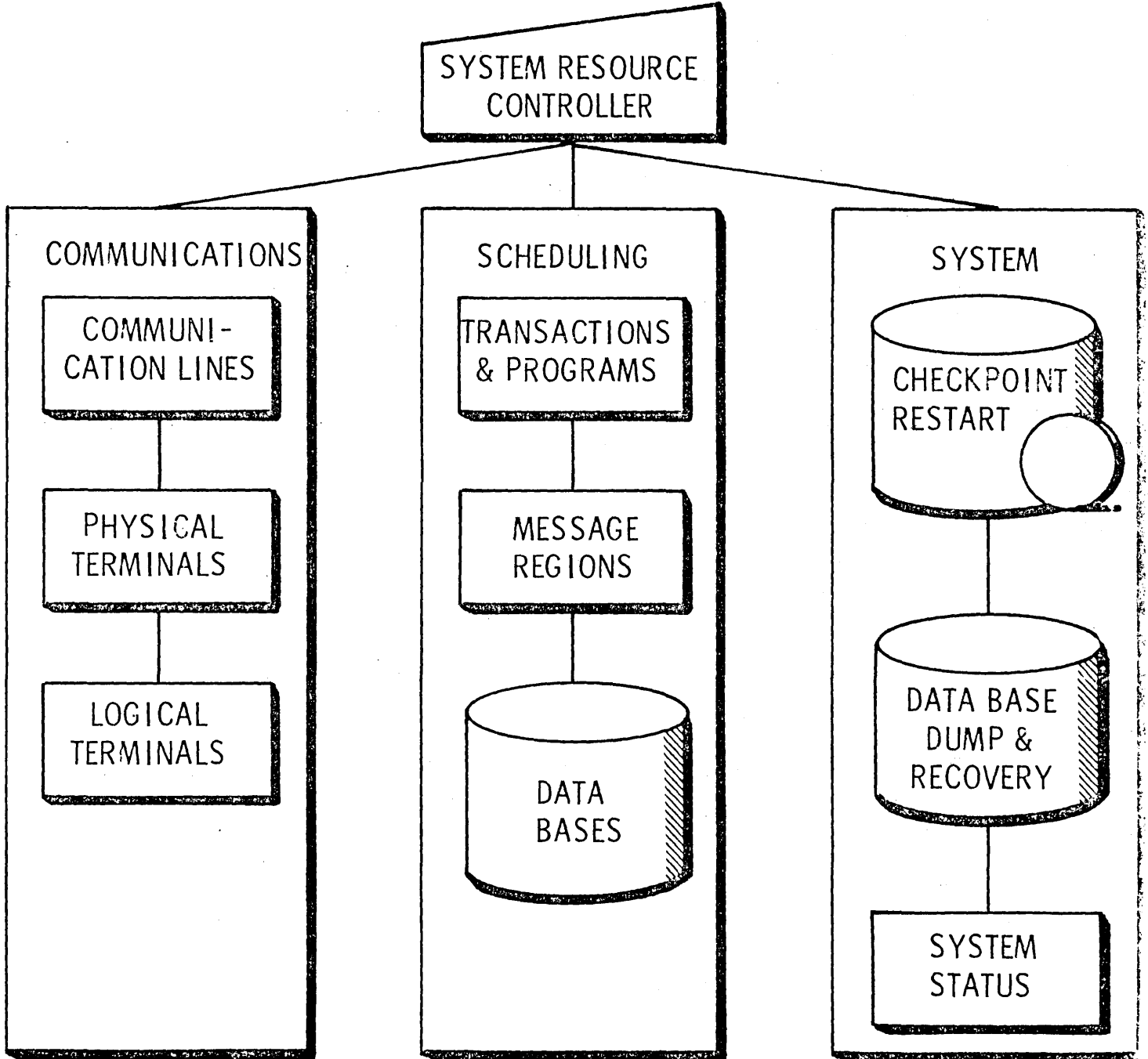
DEVELMT(HINGMT) THERE WILL BE A MEETING OF ALL
DEPARTMENTS IN THE CAFETERIA AT 2 PM



Input Message Format



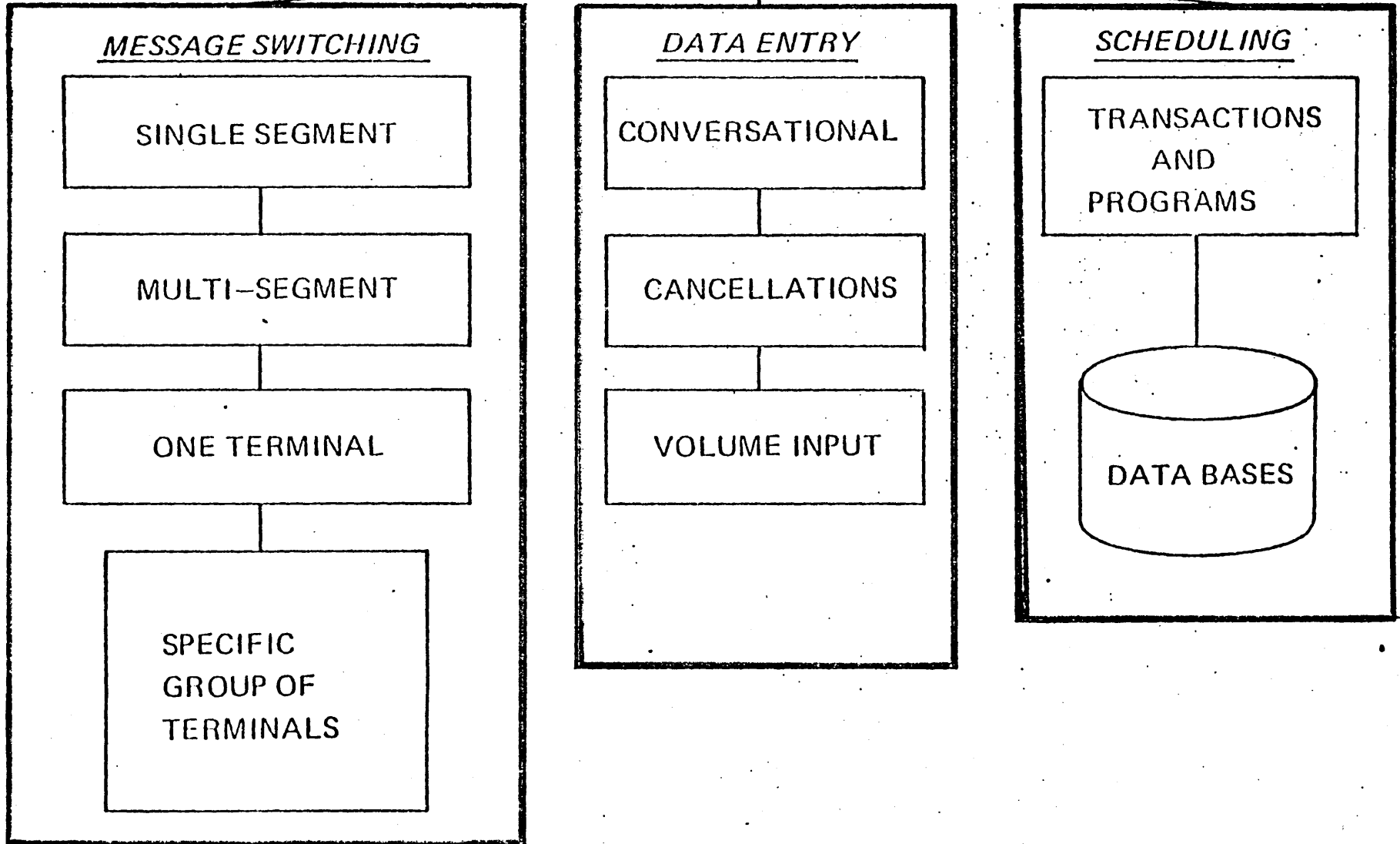
Master Terminal



MASTER TERMINAL COMMANDS

COMMAND	FUNCTION
• /ASSIGN	RELATE LTERM TO PTERM
• /DISPLAY	STATUS, QUEUES, TRAN CODE, ETC.
• /DBRECOVERY • /DBDUMP	} RECOVERY OF ONLINE DB W/O IMS/VIS TERMINATION
• /CHECK POINT • /IDLE	} TERMINATE IMS/VIS
• /NRESTART • /ERESTART	} INITIATE IMS/VIS (IPL)
• /CHANGE • /DELETE	} ALTER TERMINAL AND PASSWORD SECURITY
• /START • /STOP • /PSTOP • /PURGE • /RSTART	} AVAILABILITY OF SYSTEM RESOURCES: LINES, PTERM, LTERM, PGM, DB, TRAN, ETC.

REMOTE TERMINAL COMMANDS



SOME REMOTE TERMINAL COMMANDS

COMMAND	FUNCTION
• /BROADCAST	– TERMINAL-TO-TERMINAL MESSAGE SWITCH – TO MULTIPLE TERMINALS – MULTIPLE SEGMENTS
• /HOLD	– SAVE A CONVERSATION
• /RELEASE	– RESUME A CONVERSATION
• /EXIT	– TERMINATE A CONVERSATION
• /CANCEL	– CANCEL A MULTISEGMENT MESSAGE PRIOR TO END OF MESSAGE
• /SET	– PRESET TRAN CODE
• /RESET	– FOR VOLUME ENTRY OF DATA – SAVES KEY STROKES
• /LOCK	– PREVENT SENDING AND RECEIVING MESSAGES TO TERMINAL ENTERING /LOCK COMMAND
• /UNLOCK	– PREVENT SCHEDULING SPECIFIC – TRANSACTION CODES } UNDER – PROGRAMS } PASSWORD – DATA BASES } CONTROL

Suggested Command Utilization

MASTER TERMINAL ONLY

/DISPLAY
/ASSIGN
/CHANGE
/DELETE
/DBDUMP
/DBRECOVERY
/CHECKPOINT
/IDLE
/NRESTART
/ERESTART
/PSTOP
/PURGE
/STOP
/START
/RSTART

REMOTE AND MASTER TERMINAL

/RDISPLAY
/BROADCAST
/EXCLUSIVE
/END
/HOLD
/RELEASE
/EXIT
/SET
/RESET
/LOCK
/UNLOCK
/CANCEL
/TEST
/LOOPTEST
/LOG
/IAM (REMOTE ONLY)

STANDARD EDIT FUNCTIONS

INPUT

✂ REMOVE LEADING BLANKS AND CONTROL CHARACTERS ON THE FIRST SEGMENT

✂ ELIMINATE BACKSPACES ON ALL SEGMENTS

✂ REMOVE THE PASSWORD

✂ INSERT A TRANSACTION CODE IF THE /SET COMMAND WAS USED

• REMOVE CARRIAGE AND LINE CONTROL CHARACTERS ON ALL SEGMENTS -

• TRANSLATE ALL SEGMENTS TO EBCDIC

PHYSICAL TERMINAL INPUT EDIT

- ALLOWS USER ACCESS TO INPUT DATA BEFORE IMS/VIS BASIC EDIT AND DESTINATION DETERMINATION
- CANNOT BE USED WITH MFS
- SPECIFIED ON TYPE OR LINEGRP MACRO
- GETS CONTROL FOR EACH INPUT SEGMENT

OPTIONS: ACCEPT SEGMENT
 MODIFY SEGMENT
 CANCEL SEGMENT
 CANCEL MESSAGE,
 SEND MESSAGE TO M.T.O.
 CANCEL MESSAGE,
 SEND USER MESSAGE TO M.T.O.

OUTPUT EDITS

BASIC IMS EDIT

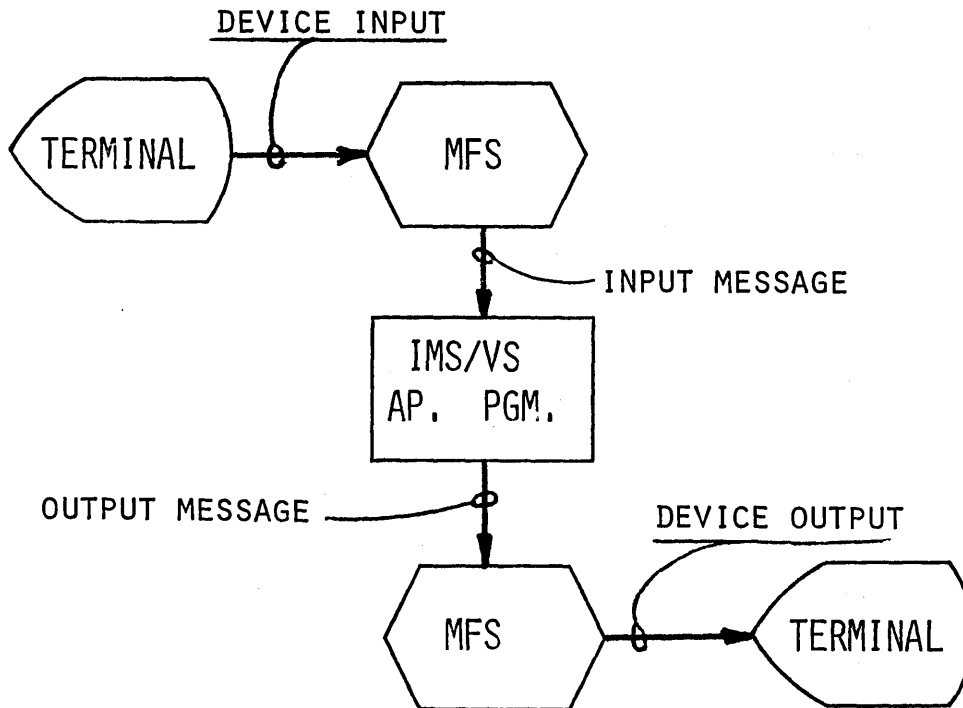
- INSERT IDLE CHARACTERS AFTER NEW LINE, LINE FEED AND TAB CHARACTERS
- ADD LINE CONTROL CHARACTERS FOR COMMUNICATION LINE
- MOVES SEGMENT FROM QPOOL BUFFER TO OUTPUT BUFFER.

OPTIONAL USER EDIT

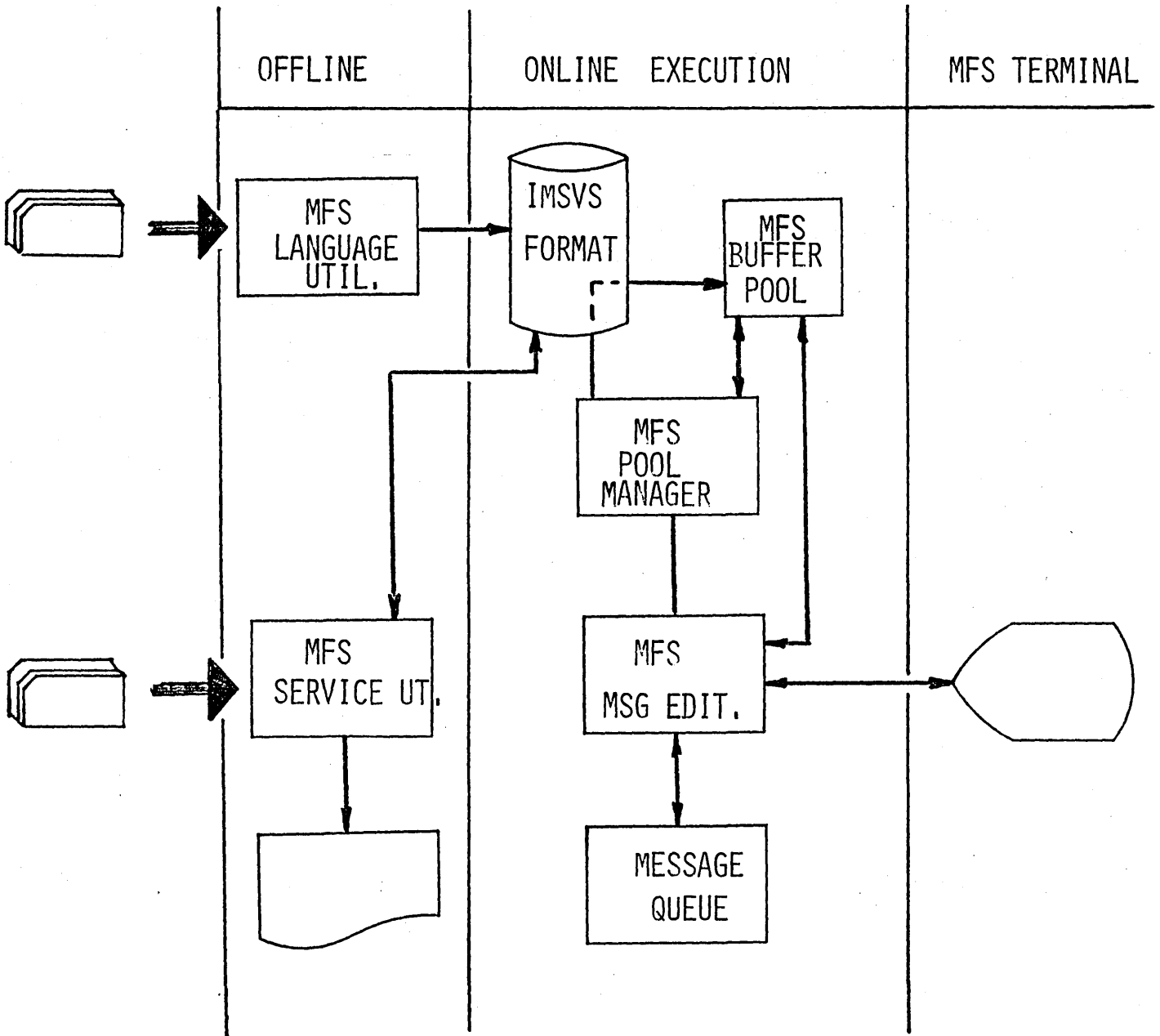
- 1/LINE GROUP
 - SPECIFY WHICH PHYSICAL TERMINALS WILL USE EDIT
- GAINS CONTROL ONCE PER SEGMENT BEFORE BASIC EDIT & ONCE MORE AT END OF MESSAGE.
- MESSAGE MAY BE EDITED IN PLACE IN WHICH CASE LENGTH MAY NOT BE INCREASED BY MORE THAN 10 BYTES.

MESSAGE FORMAT SERVICES

- PRIMARY SUPPORT FOR THE 3270 DISPLAY
- ALSO SUPPORTS THE 2741, 2740 AND 3600 SYSTEM
- SIMPLIFIES APPLICATION PROGRAMS
- OPTIMIZES ONLINE PERFORMANCE
- PROVIDES DEVICE INDEPENDENCE



MESSAGE FORMAT SERVICE OVERVIEW



MESSAGE FORMAT SERVICES

<APPLICATION PROGRAM FUNCTIONS:>

JUSTIFICATION

PADDING

EXITS FOR VALIDITY CHECKS

TIME/DATE STAMP

PAGE/MSG NUMBERING

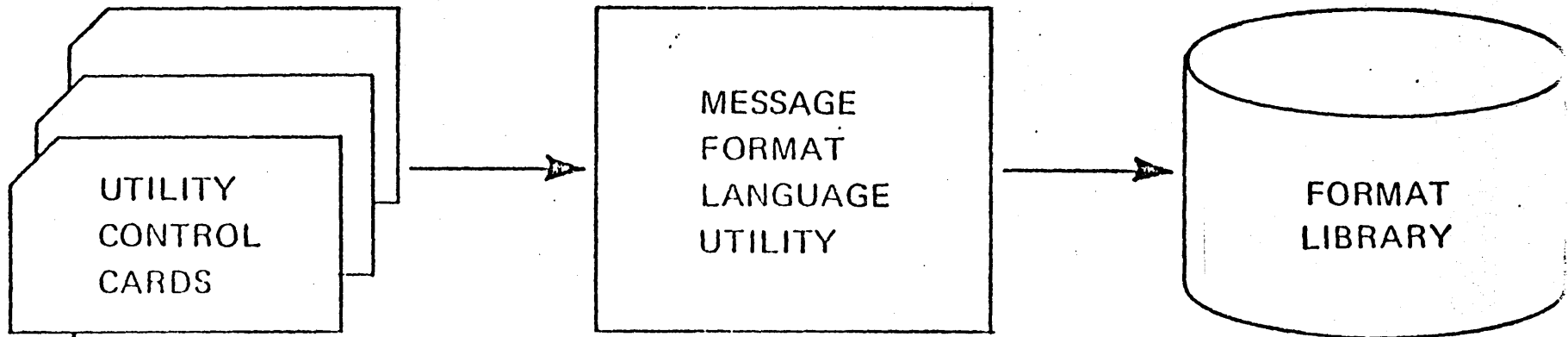
DATA SEQUENCING/SEGMENTING

PERFORMANCE FUNCTIONS:

PRE-FETCH

LOOK-ASIDE BUFFERING

OFFLINE



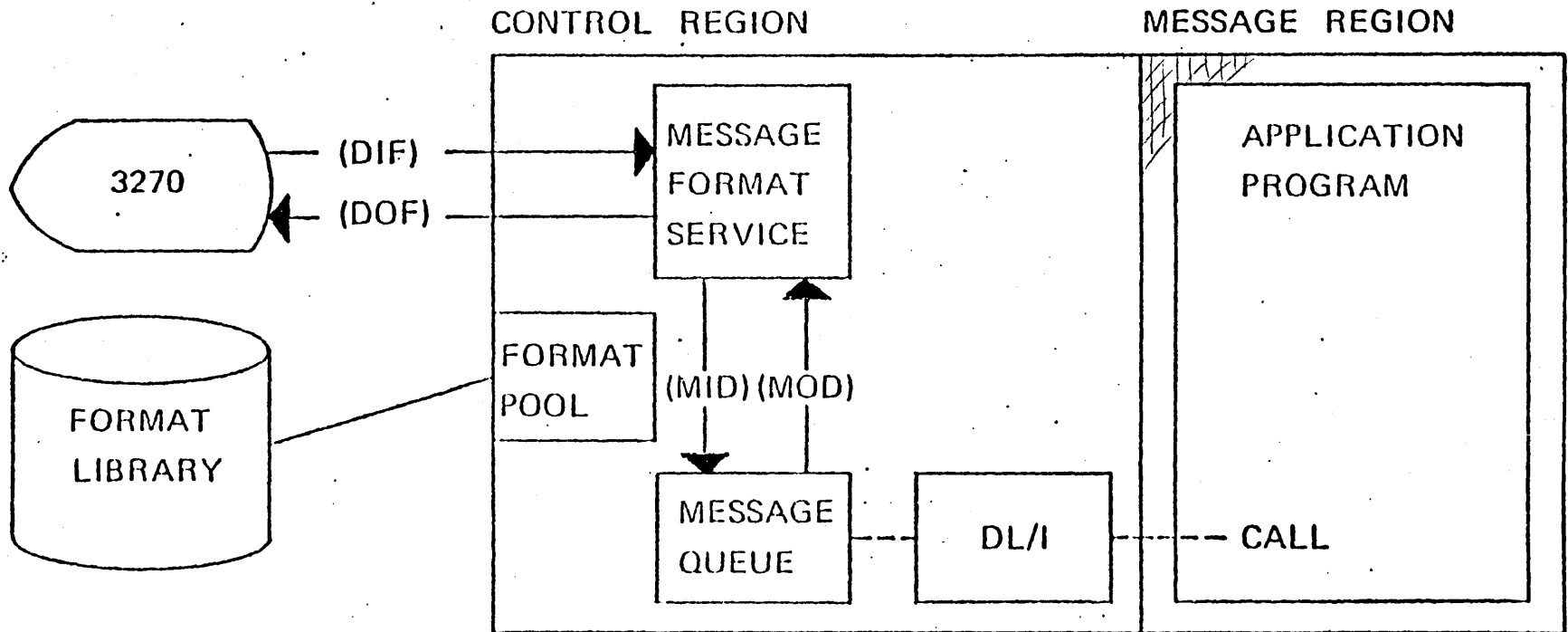
APPLICATION PROGRAM FORMAT

- INPUT AND OUTPUT SEGMENT FIELD LAYOUT
- FIELD JUSTIFICATION/PADDING/TRUNCATION
- LITERALS

3270 DISPLAY SYSTEM FORMAT

- DEVICE FORMAT BY FIELD
- LITERALS

MESSAGE FORMAT SERVICE



DIF - DEVICE INPUT FORMAT
MID - MESSAGE INPUT DESCRIPTOR

{ DOF } - DEVICE OUTPUT FORMAT
MOD - MESSAGE OUTPUT DESCRIPTOR

DISPLAY DESIGN

1 INPUT

USER REQUEST
FORMAT 'LE'

/FORMAT LE

2 OUTPUT

FORMAT
PROVIDED FOR USER

SKILL _
SKILL LEVEL
YEARS/FIRM
YEARS/SKILL
LOCEMP

3 INPUT

USER KEYS IN
VARIABLE DATA, ENTERS

SKILL INDENG
SKILL LEVEL 3
YEARS/FIRM 4
YEARS/SKILL 1 _
LOCEMP

4 OUTPUT

RESPONSE
SENT TO USER

SKILL INDENG		
NAME	NUMBER	DEPT
JONES, CE	3294	B91
BAKER, KT	4105	C24
ALLAN, T	4469	C36

MESSAGES

DEVICE INPUT

SKILL INDENG
SKILL LEVEL 3
YEARS/FIRM 4
YEARS/SKILL 1
LOCEMP

MESSAGE INPUT

LL ZZ LOCEMP INDENG030401

DEVICE

APPLICATION PROGRAM

DEVICE OUTPUT

SKILL INDENG		
NAME	NUMBER	DEPT
JONES, CE	3294	B91
BAKER, KT	4105	C24
ALLAN, T	4469	C36

MESSAGE OUTPUT

LL ZZ INDENG JONES, CE 3294B91BAKER, KT

4105C24ALLAN, T 4469C36

INPUT MESSAGE EDITING

- INPUT TO PROGRAM DEFINED TO BE

MESSAGE

SEGMENT(S)

FIELD(S)

- EACH FIELD IS DEFINED AS TO ITS

SOURCE (WHERE IS IT ON THE SCREEN)

LENGTH

JUSTIFICATION

FILL CHARACTER

- 3 OPTIONS ARE AVAILABLE FOR HOW TO PRESENT THE FIELDS IN SEGMENTS TO THE APPLICATION PROGRAM

⟨DEVICE DEPENDENT INPUT OPTIONS⟩

- INPUT CURSOR LOCATION

LL	CC
----	----

2 2

LL = LINE NUMBER

CC = COLUMN

- DEFERED DETECTABLE FIELDS

?DATA

IF NO STRIP

DATA

IF STRIP

- AN IMMEDIATELY DETECTABLE FIELD

?

IF THERE ARE OTHER DATA FIELDS

DATA

IF IT IS THE ONLY FIELD

OUTPUT MESSAGE EDITING

- OUTPUT FROM PROGRAM DEFINED TO BE

MESSAGE

LOGICAL PAGE(S)

SEGMENT(S)

FIELD(S)

- 3 OPTIONS ARE AVAILABLE AS WITH INPUT

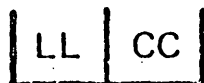
DEVICE DEPENDENT OUTPUT OPTIONS

- SYSTEM CONTROL AREA (SCA)

A FOUR BYTE FIELD IN THE FIRST SEGMENT OF A MESSAGE THAT CHANGES CONTROL INFORMATION SUCH AS ERASING, WRITE CONTROL, SOUNDING THE ALARM, AND COPY

- CURSOR LOCATION

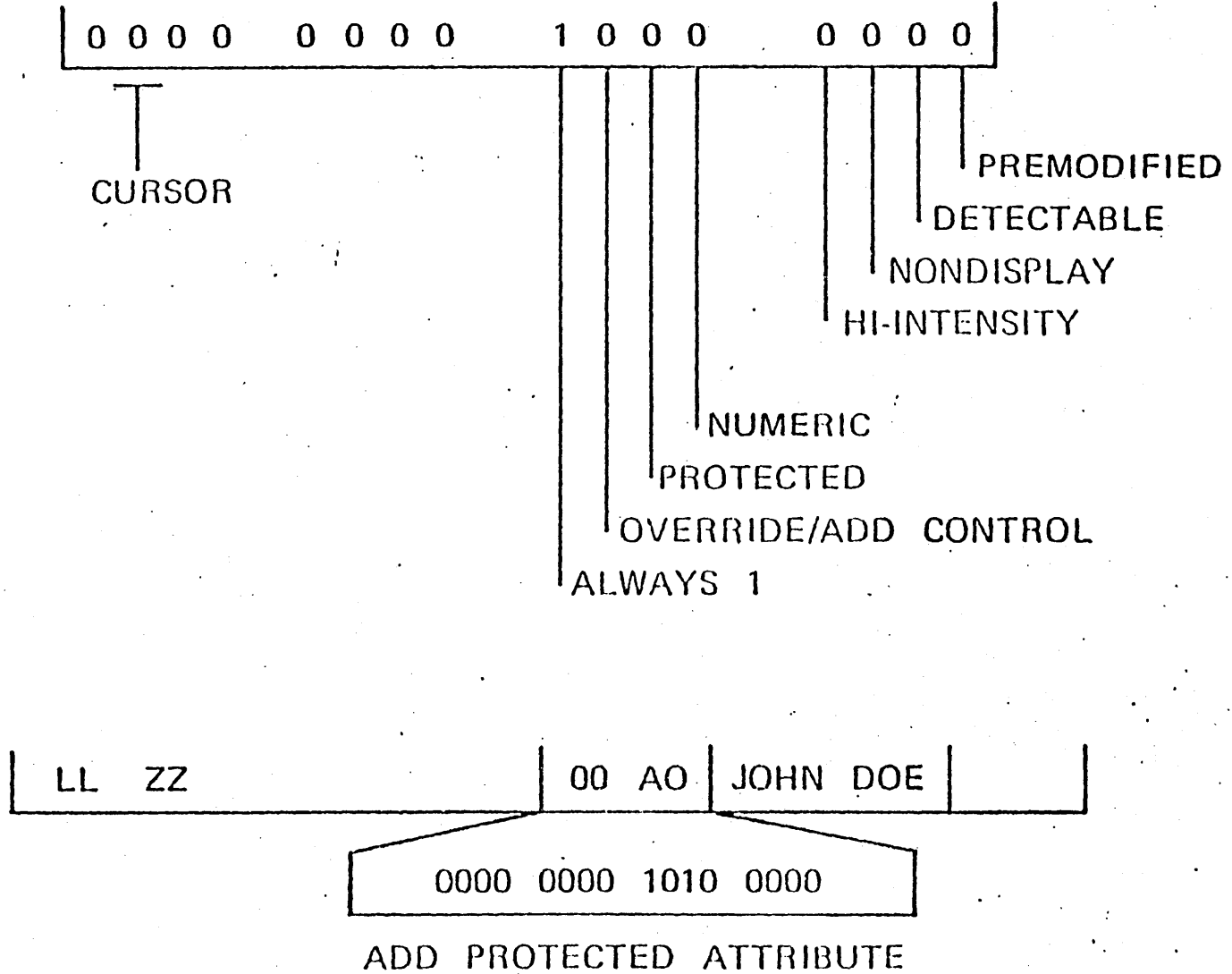
A FOUR BYTE FIELD WITH THE LINE AND COLUMN POSITION FOR THE CURSOR



- DYNAMIC ATTRIBUTE BYTE MODIFICATION

- MUST BE SPECIFIED IN THE FIELD DEFINITIONS TO MFS
- TWO BYTES AT THE BEGINNING OF THE FIELD CONTAIN THE ATTRIBUTE BYTE

DYNAMIC ATTRIBUTE MODIFICATION



FORMAT DOCUMENTATION

INPUT

SEGMENT FORMATS

OPTION

FILL

SPECIAL DEVICE OPTIONS

OUTPUT

SEGMENT FORMATS

OPTION

LOGICAL PAGING

MODNAME REQUIREMENTS

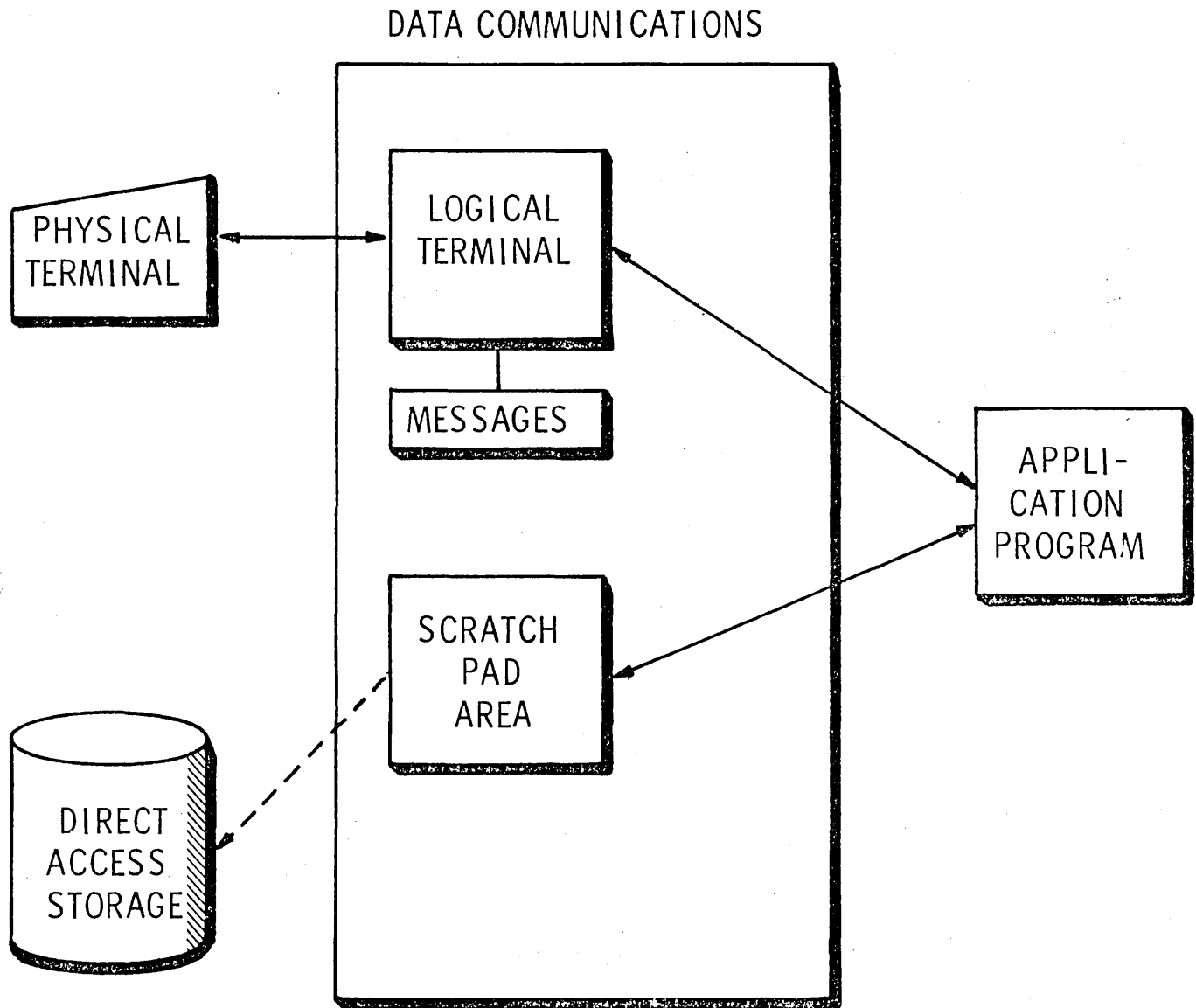
SPECIAL DEVICE OPTIONS

USE OF ATTRIBUTES

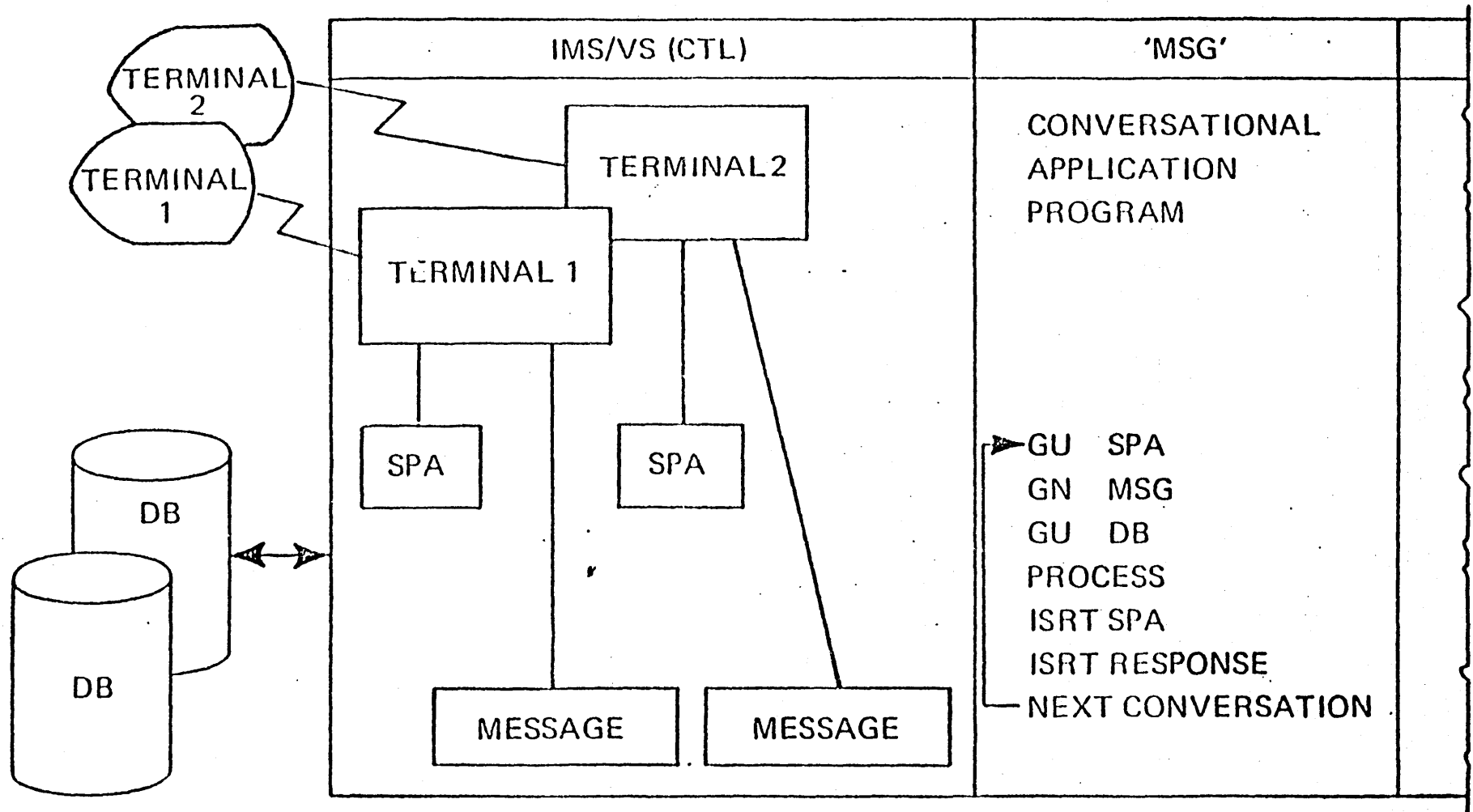
CONVERSATIONAL PROCESSING

- ALLOWS TERMINAL OPERATOR TO INTERACT MORE THAN ONCE WITH AN APPLICATION PROGRAM (OR PROGRAM SET)
- INTERMEDIATE DATA IS STORED IN A SCRATCH PAD AREA (SPA) THAT THE PROGRAM HAS ACCESS TO THE NEXT TIME IT GETS SCHEDULED (CAN "REMEMBER" LAST RESPONSE)
- PROGRAM DOES NOT REMAIN RESIDENT IN MSG BETWEEN INTERCHANGES
- "SPA" CAN BE IN MAIN OR DISK STORAGE
- CAN SEND INFO TO NEW PGM VIA "SPA" AND CONTINUE CONVERSATION

Conversational Processing

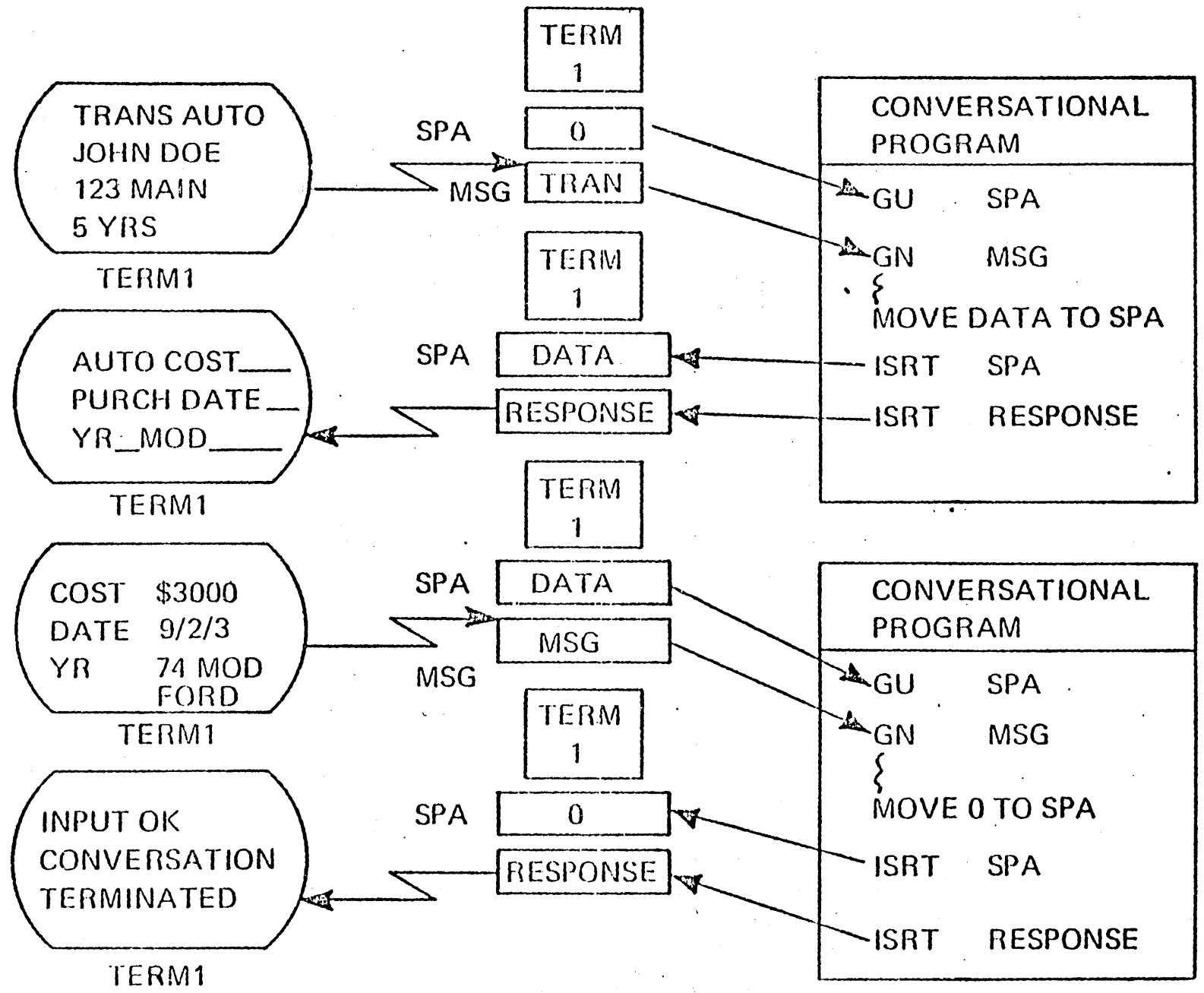


CONVERSATIONAL PROCESSING

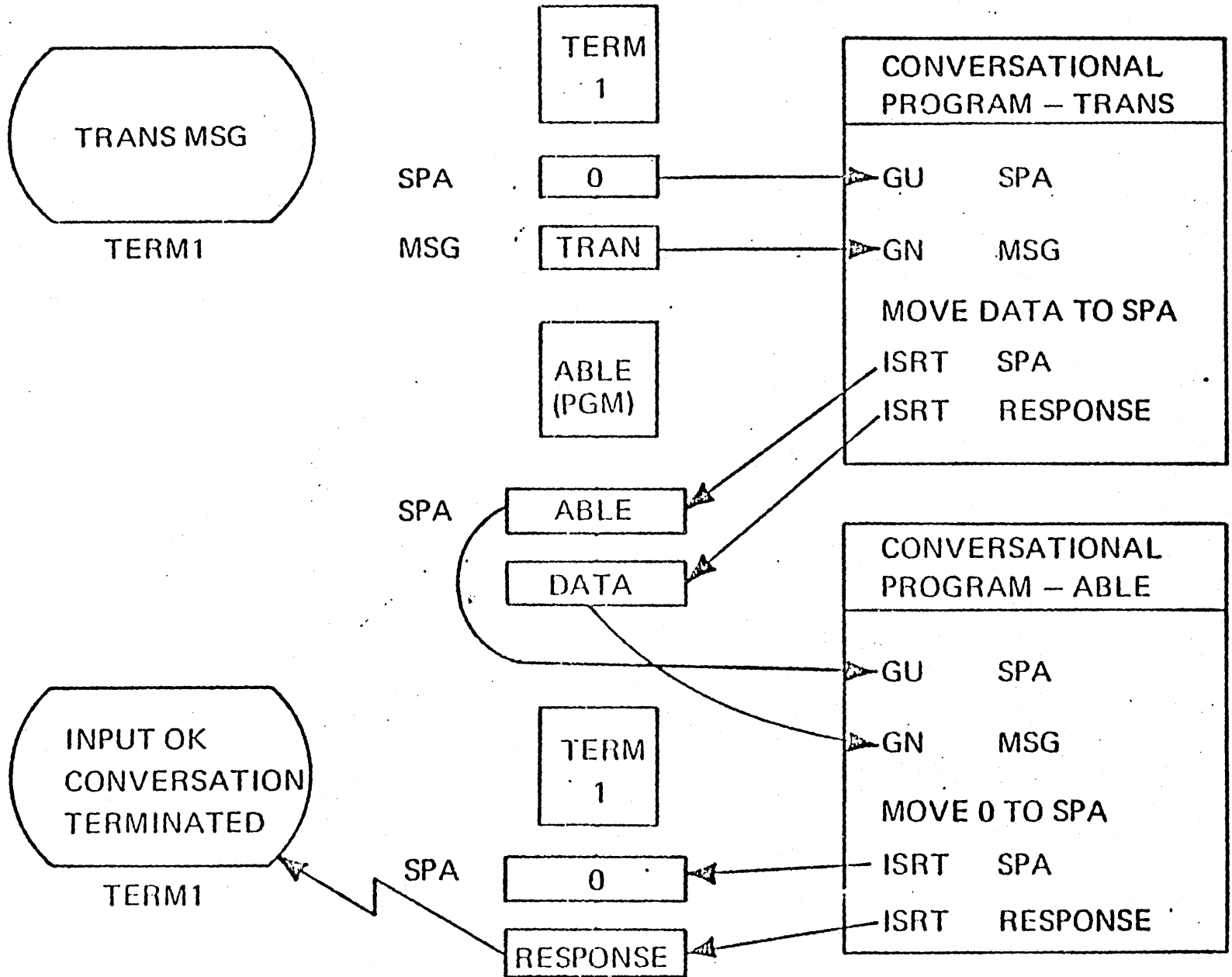


A SEPARATE SPA IS ASSOCIATED WITH EACH TERMINAL INVOLVED IN A CONVERSATION, SO PROGRAM CAN CONCURRENTLY CONVERSE WITH ANY NUMBER OF TERMINALS.

CONVERSATIONAL EXAMPLE - 1



CONVERSATIONAL EXAMPLE - 2



OPERATOR COMMANDS

● COMMAND: /HOLD

REPLY: . DFS999 *HELD CONVERSATION ID IS xxx

● COMMAND: /RELEASE CONV xxx

REPLY: DFS058 RELEASE COMMAND COMPLETE
LAST OUTPUT MESSAGE SENT AGAIN TO OPERATOR

● COMMAND: /EXIT

REPLY: DFS058 EXIT COMMAND COMPLETE
(CONVERSATION IS TERMINATED)

IMS INTEGRITY FUNCTIONS

- AUTOMATIC LOGGING
- DATA BASE RECOVERY UTILITIES
- AUTOMATIC CHECKPOINTS
- PROGRAM ISOLATION
- NORMAL RESTART
- EMERGENCY RESTART
 - TRANSACTION RESTART
 - DATA BASE BACKOUT
 - MESSAGE QUEUE RECONSTRUCTION
 - MESSAGE TRANSMISSION REPOSITIONING
- COMMUNICATIONS HARDWARE ERROR HANDLING

SYSTEM INTEGRITY

POINTS OF FAILURE

RESOURCE AFFECTED

RESPONSIBILITY

TERMINAL HARDWARE	FAILING DEVICE	COMMUNICATIONS ROUTINES
DATA BASE	DATA BASE	RECOVERY UTILITIES
IMS CONTROL PROGRAM	SYSTEM	EMERGENCY RESTART
OPERATING SYSTEM	SYSTEM	EMERGENCY RESTART
APPLICATION PROGRAM	APPLICATION PROGRAM	PROGRAM ISOLATION

I M S / V S I M M E D I A T E C H E C K P O I N T

ACTIONS:

-
1. WRITE ALTERED MSG QUEUE BUFFERS TO DISK
 2. WRITE CHECKPOINT RECORDS TO LOG TAPE
 3. WRITE SUMMARY OF ACTIVE PST'S TO LOG TAPE
 4. WRITE CHECKPOINT ID TO MASTER TERMINAL
 5. WRITE CHECKPOINT ID TO DISK (QBLKS)
 6. CONTINUE NORMAL PROCESSING

PARTITION
SPECIFICATION
TABLE

SHUTDOWN CHECKPOINTS

/CHECKPOINT FREEZE

[QUIESCE]

[ABDUMP]

/CHECKPOINT DUMPQ

/CHECKPOINT PURGE

SHUTDOWN CHECKPOINTS

Action is top to bottom

ACTION ORDER:

	FREEZE	DUMPQ	PURGE
STOP TERMINAL INPUT	X	X	X
STOP TERMINAL OUTPUT	X	X	
PROCESS ALL SCHEDULABLE TRANSACTIONS			X
FREE MESSAGE REGIONS	X	X	X
TERMINATE MESSAGE REGIONS	X	X	X
SEND ALL OUTPUT			X
STOP TERMINAL OUTPUT			X
PURGE ALTERED MSG QUEUE BUFFERS TO DISK	X	X	X
CLOSE ALL DATA BASES	X	X	X
LOG IMS/VS SYSTEM CONTROL BLOCKS	X	X	X
DUMP QUEUES AND SPA'S TO LOG TAPE		X	X
CLOSE IMS/VS SYSTEM LOG	X	X	X
WRITE CHECKPOINT ID TO MASTER TERMINAL	X	X	X
WRITE CHECKPOINT ID TO SYSTEM CONSOLE	X	X	X
UPDATE CHECKPOINT-ID TABLE	X	X	X
CLOSE IMS/VS SYSTEM DATA SETS	X	X	X
TERMINATE	X	X	X

RESTART FACILITY

- COLD START - NO PREVIOUS SYSTEM
STATUS RETAINED

- WARM START - OBTAIN PREVIOUS SYSTEM
STATUS FROM CHECKPOINT
WRITTEN AT SYSTEM
SHUTDOWN

- EMERGENCY RESTART - WHEN SYSTEM FAILURE
CAUSED IMS/VIS TERMINATION
WITHOUT ORDERLY SHUTDOWN

BATCH CHECKPOINT

OBJECTIVE

IMPROVE SYSTEM INTEGRITY BY PROVIDING:

- IMS CHECKPOINT FOR BATCH DATA BASES
- SYNCHRONIZATION OF NON-IMS RESOURCES
- USER CONTROL OF SYNC POINTS
- FASTER BATCH RECOVERY TIME

SYSTEM ACTION

- FLUSH DATA BASE BUFFERS
- LOG CHECKPOINT ID AND WTO
- OPTIONAL OS/VS CHECKPOINT

PROGRAM ISOLATION

- ENABLES THE EFFECTS (DATA & MESSAGES) OF AN ABENDING PROGRAM TO BE REMOVED DYNAMICALLY WITHOUT CONTAMINATING OTHER PROCESSING.
- * RESOLVES DEADLOCKS BETWEEN PROGRAMS WISHING TO UPDATE THE SAME SEGMENT OCCURRENCE.
- PERMITS 2 OR MORE PROGRAMS WITH UPDATE INTENT AGAINST THE SAME SEGMENT TYPE TO BE SCHEDULED CONCURRENTLY, & LETS DL/I MANAGE THEIR REAL INTENT ON A SEGMENT OCCURRENCE LEVEL.

P. I. CONCEPT :

ALL ACTIVITY OF AN APPLICATION

- DB CHANGES
- MSGs GENERATED

ISOLATED FROM ANY OTHER APPL. ACTIVITY

UNTIL

THE APPLICATION COMMITS VIA A SYNCH PT.

THAT ITS ACTIVITY TO THIS

POINT IS VALID.

SYNCH POINTS

TAKEN WHEN:

- APPLICATION PROGRAM TERMINATES
- DL/I CHECKPOINT CALL
- GU MSG Q (SINGLE MODE XACTION)

INS/VIS ACTION:

- FLUSH DB BUFFERS
- DEQ INPUT MSG SEGMENTS
- ENQ OUTPUT MSG SEGMENTS
(I.E. MOVE FROM TEMP Qs TO LTERM Qs)
- TERMINATE DYNAMIC LOGGING FOR THAT REGION
- EXCLUSIVE CTL ENG/DEQ RELEASES OWNERSHIP
OF DB SEGMENTS

SECURITY & PRIVACY

EXPLICIT RESTRICTIONS IMS ALLOWS

VIA

TRANSACTIONS / APPLICATIONS PSBGEN

DATA BASES / APPLICATIONS PSBGEN

SEGMENTS / APPLICATIONS PSBGEN

PROCESSING / SEGMENTS / APPLICATIONS PSBGEN

TRANSACTIONS / CLASSES PSBGEN

CLASSES / REGIONS MASTER TERMINAL

START / STOP / MODIFY / RESOURCES MASTER TERMINAL

LINES

DATA BASES

TERMINALS

TRANSACTIONS

⋮

UNAUTHORIZED ATTEMPTS REPORTED TO MASTER TERMINAL

DATA ENCRYPTION (VIA EDIT/COMPRESS EXIT)

SECURITY MAINTENANCE PROGRAM

Types Of Statistical Reports

- MESSAGES QUEUED BUT NOT SENT--BY TERMINAL
- MESSAGES QUEUED BUT NOT SENT--BY TRANSACTION
- TRANSACTION REPORT
- TRANSACTION RESPONSE REPORT
- APPLICATION ACCOUNTING REPORT
- IMS/VIS ACCOUNTING REPORT

Transaction Response Report

DATE 09/30/7- PAGE 1

TYPE TRANS.	TOTAL RESPONSES	LONGEST RESPONSE	95% RESPONSE	75% RESPONSE	50% RESPONSE	25% RESPONSE	SHORTEST RESPONSE
(NOSORC)	15	03M19.2S	03M19.2S	07.4S	05.9S	00.6S	00.4S
/CHE	3	55.4S	55.4S	19.6S	10.8S	08.\$s	06.2S
PAYINQ	25	30.0S	28.0S	12.0S	05.0S	03.0S	02.0S
SKILLINQ	5	20.0S	15.0S	08.0S	06.0S	04.0S	03.0S

Application Accounting Report

DATE 09/30/7-

PAGE 1

DATA BASE COUNTS

PROGRAM NAME	TRANSACTION NAME	MESSAGE PRI	QUANTITY	GU	GN	ISRT	GU	GN	BLKS MOVED	BAD CC	TOT MESS CPU TIME	AVR TIME
PSB00001	PAYINQ	01	71	142	14	71	81	42	1	1	10.65S	0.15S
		02	81	162	16	81	91	31	1	0	12.15S	0.15S
		**	152	304	30	152	172	73	2	1	22.80S	0.15S
SYSTEM TOTAL			152	304	30	152	172	73	2	1		

System Definition

