HEWLETT hp PACKARD

IMAGE DATA BASE DESIGN
AND
PERFORMANCE MEASUREMENT

Presented To:

HP GENERAL SYSTEMS USERS GROUP
7th International Meeting

October 30th - November 3rd 1978

Denver Hilton
Denver, Colorado

BY

Orland J. Larson
IMAGE/3000 Product Manager

General Systems Division
HEWLETT-PACKARD COMPANY

OUTLINE

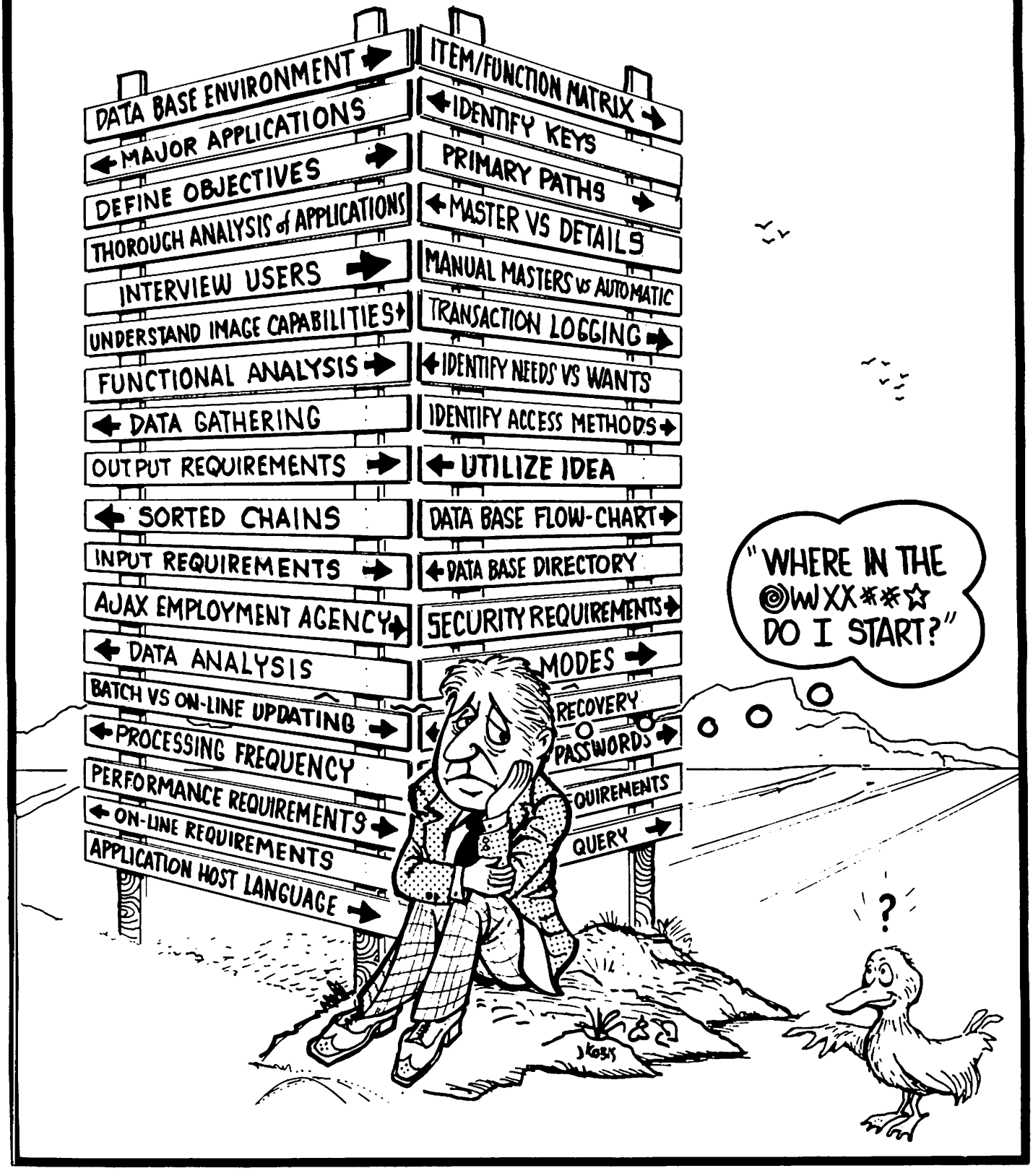
I. INTRODUCTION
   A. OVERVIEW
   B. CONCERNS OF THE DATA BASE DESIGNER
   C. LOGICAL DATA BASE DESIGN
   D. PHYSICAL DATA BASE DESIGN

II. DATA BASE DESIGN METHODOLOGY
   A. OVERVIEW OF DESIGN AIDS
   B. DATA BASE DESIGN AIDS
      1. DATA BASE ENVIRONMENT
      2. USER FUNCTION FLOW DIAGRAM
      3. USER FUNCTIONS AND REQUIREMENTS
      4. ITEM/FUNCTION MATRIX
      5. DATA BASE MODEL
      6. DATA BASE DIRECTORY
      7. DETAILED TRANSACTION ANALYSIS

III. IDEA – A PERFORMANCE MEASUREMENT TOOL
   A. OVERVIEW OF IDEA
   B. REQUIREMENTS
   C. OPERATION
   D. SAMPLE SCRIPT CREATION
   E. SAMPLE OUTPUT OF IDEA RUN
   F. GRAPHS GENERATED FROM IDEA RESULTS
   G. DISADVANTAGES OF PERFORMANCE MEASUREMENT
   H. ADVANTAGES OF PERFORMANCE MEASUREMENT

IV. SUMMARY

DATA BASE DESIGN



CONCERNS OF THE DATA BASE DESIGNER

- **MUST FULFILL TWO MAIN OBJECTIVES**
  - PROVIDE A STABLE DESIGN TO SATISFY NEEDS OF CURRENT AND FUTURE USERS
  - PROVIDE COST/PERFORMANCE TRADEOFF INFORMATION TO MANAGEMENT
- **CHANGEABILITY OF USER REQUIREMENTS**
  - OUTPUT FORMATS (RAPIDLY)
  - POLICIES (MODERATELY)
  - MANUAL PROCEDURES AND INPUT REQUIREMENTS (SLOWLY)
- **COMPLEXITY OF INTERACTIONS**
  - DATA PEOPLE MACHINES PROCEDURES
- **DATA INTEGRITY**
  - MAKING "ALL DATA" AVAILABLE TO "ALL APPROPRIATE USERS"
- **FLEXIBILITY**
  - CHANGING STRUCTURE WITH MINIMUM IMPACT ON EXISTING USERS
  - DATA INDEPENDENCE
- **LACK OF A DATA BASE DESIGN METHODOLOGY**
- **LACK OF PERFORMANCE MEASUREMENT TOOLS**
  - THROUGHPUT ESTIMATES
  - RESPONSE TIME ESTIMATES

LOGICAL DATA BASE DESIGN

INVOLVES THE BUSINESS ENVIRONMENT

- CURRENT AND FUTURE INFORMATION REQUIREMENTS
- OVERALL OPERATIONAL PLANS AND POLICIES
- ANALYSIS OF FUNCTIONS AND INTERACTIONS (INTERVIEWS)
  - TOP MANAGEMENT VIEW
  - FUNCTIONAL MANAGEMENT VIEW
  - OPERATIONAL PERSONNEL VIEW

IDENTIFICATION OF DATA ELEMENTS

- DATA ELEMENTS USED BY EACH FUNCTION
  - DEFINITION
  - SIZE AND DATA TYPE
  - VALUES
  - SECURITY SPECIFICATIONS
- DATA ELEMENTS SHARED BETWEEN FUNCTIONS

IDENTIFICATION OF LOGICAL DESIGN COMPONENTS

- KEYS
- ATTRIBUTES
- RELATIONSHIPS

PHYSICAL DATA BASE DESIGN

- **INVOLVES THE TECHNICAL ENVIRONMENT**

- **PROVIDES STRUCTURES AND ACCESS METHODS NECCESSARY TO IMPLEMENT THE LOGICAL DATA BASE DESIGN**

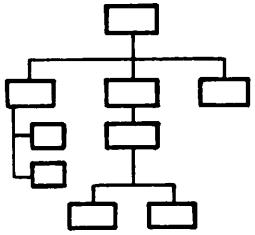
- **CONCERNED WITH:**
  - CAPABILITIES AND LIMITATIONS OF A SPECIFIC DBMS
  - COMPUTER HARDWARE UTILIZED
  - USER REQUIREMENTS OF EACH APPLICATION
  - PERFORMANCE REQUIREMENTS OF EACH APPLICATION
  - PERFORMANCE MEASUREMENT
  - BACKUP AND RECOVERY PROCEDURES
  - DATA BASE IMPLEMENTATION
  - DATA BASE MONITORING
  - SECURITY

- **DATA BASE DESIGN DECISIONS**

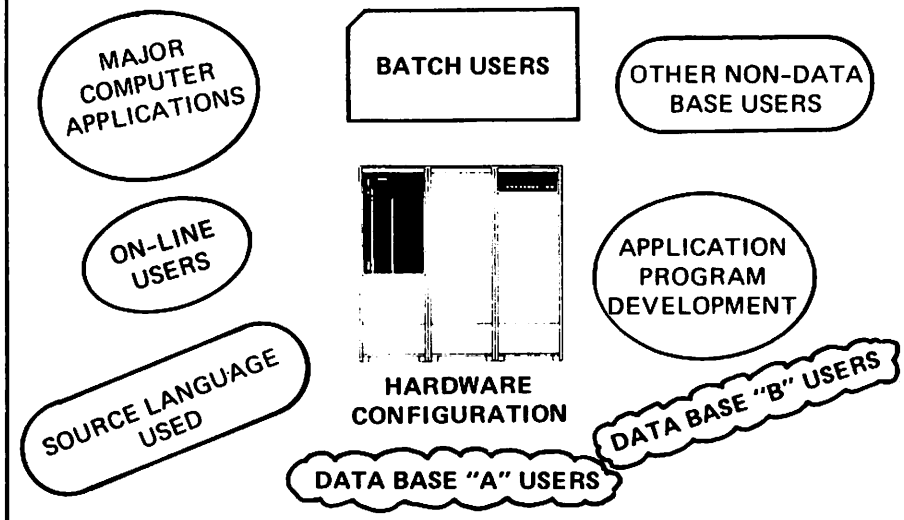
HEWLETT hp PACKARD

DATA BASE DESIGN AIDS OVERVIEW

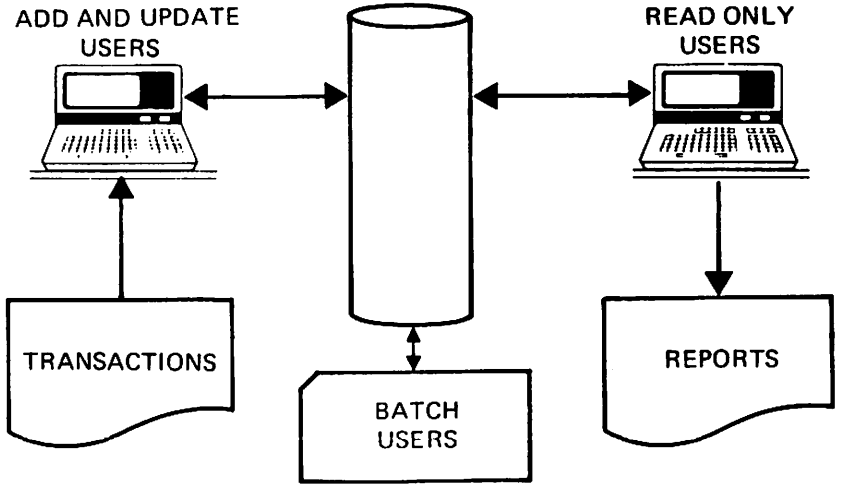
PRELIMINARY INVESTIGATIONS
STRUCTURE - FUNCTIONS - OBJECTIVES OF THE COMPANY

ORGANIZATIONAL STRUCTURE	BUSINESS FUNCTIONS	OBJECTIVES
	Services Provided Products Manufactured	Reduce Costs Increase Business Make A Profit Improve Services New Products

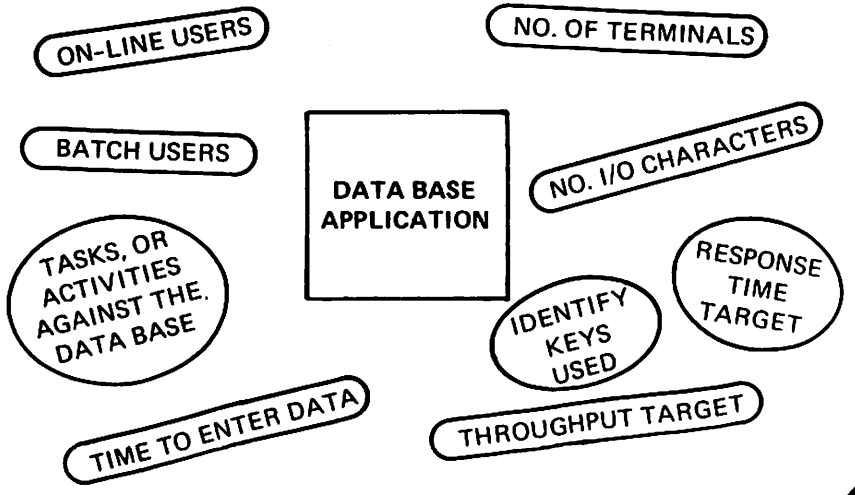
DESIGN AID #1
COMPUTER/DATA BASE ENVIRONMENT



DESIGN AID #2
FUNCTION/ACTIVITY/FLOW DIAGRAM



DESIGN AID #3
USER FUNCTIONS AND REQUIREMENTS



HEWLETT hp PACKARD

DATA BASE DESIGN AIDS OVERVIEW

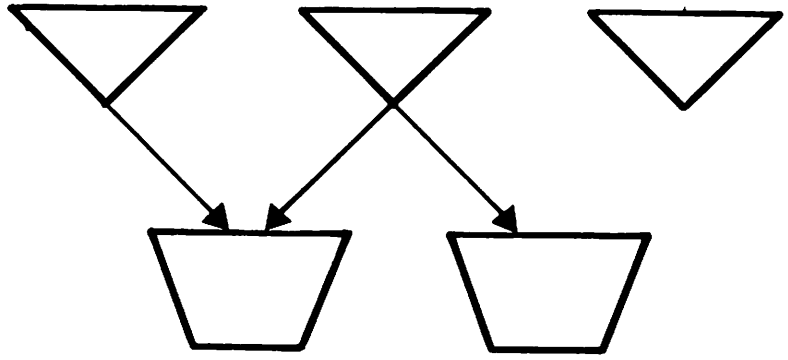
DESIGN AID #4
ITEM/FUNCTION MATRIX

FUNCTIONS

ITEMS	TYPE/ SIZE	SEARCH ITEM	SORT ITEM	UPDATE FREQ.						

(ADDITIONAL DATA USAGE STATISTICS MAY BE APPROPRIATE)

DESIGN AID #5
DATA BASE MODEL



DESIGN AID #6
DATA BASE DIRECTORY

SETS	ITEMS	TYPE/ SIZE	SECURITY	KEY	SORT	P/C	CAP	DESC.

DESIGN AID #7
DETAILED ACTIVITY ANALYSIS AGAINST THE DATA BASE

USER	TRANS ID	NO. TERMS	TIME TO ENTER	NO. I/O CHAR	DB NAME	OPEN MODE	SET NAME	CALL	KEY	ARG

DESIGN AID #1
DATA BASE ENVIRONMENT

COMPANY NAME: _____

TYPE OF BUSINESS: _____

ADDRESS: _____

DATA BASE ADMINISTRATOR: _____

TELEPHONE NUMBER: _____

HARDWARE ENVIRONMENT

COMPUTER _____ MODEL _____

MEMORY SIZE _____

DISCS _____ CAPACITY _____

TAPES _____ B.P.I. _____

PRINTERS _____ L.P.M. _____

NO. TERMINALS _____ MAKE, MODEL & BAUD RATE _____

MAJOR ACTIVITIES OR APPLICATIONS	NO. BATCH USERS	NO. ONLINE USERS	NO. READ ONLY USERS	NO. ADD & UPDATE USERS	SUBSYS. OR LANG. USED	PRIORITY

DESIGN AID #1

DATA BASE ENVIRONMENT

COMPANY NAME: ACME WIDGETS

TYPE OF BUSINESS: WIDGET DISTRIBUTOR

ADDRESS: CUPERTINO, CALIF.

DATA BASE ADMINISTRATOR: O J. LARSON

TELEPHONE NUMBER: (408) 253-1234

HARDWARE ENVIRONMENT

COMPUTER HP 3000 SERIES II MODEL 8

MEMORY SIZE 512 KB

DISCS (2) 7920 CAPACITY 50 MB EA

TAPES (2) 7970 B.P.I. 1600

PRINTERS 30133A L.P.M. 600

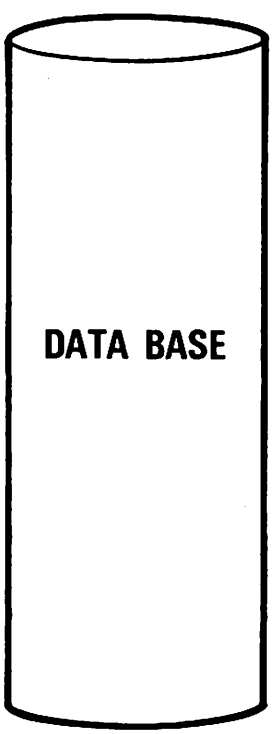
NO. TERMINALS 46 MAKE, MODEL & BAUD RATE HP 2645 9600 Baud

MAJOR ACTIVITIES OR APPLICATIONS	NO. BATCH USERS	NO. ONLINE USERS	NO. READ ONLY USERS	NO. ADD & UPDATE USERS	SUBSYS. OR LANG. USED	PRIORITY
ORDER PROCESSING	1	13	2	11	COBOL/ QUERY	
INUENTORY CONTROL	1	5	3	2	COBOL	
ACCOUNTS RECEIVABLE	1	3	1	2	COBOL	
ACCOUNTS PAYABLE	1	3	2	1	COBOL	
PAYROLL	2	6	4	2	COBOL	
PERSONNEL/ADMIN.	2	6	4	2	COBOL	
COBOL PROGRAM DEV.	2	10			COBOL	

DESIGN AID #2

USER FUNCTION FLOW DIAGRAM

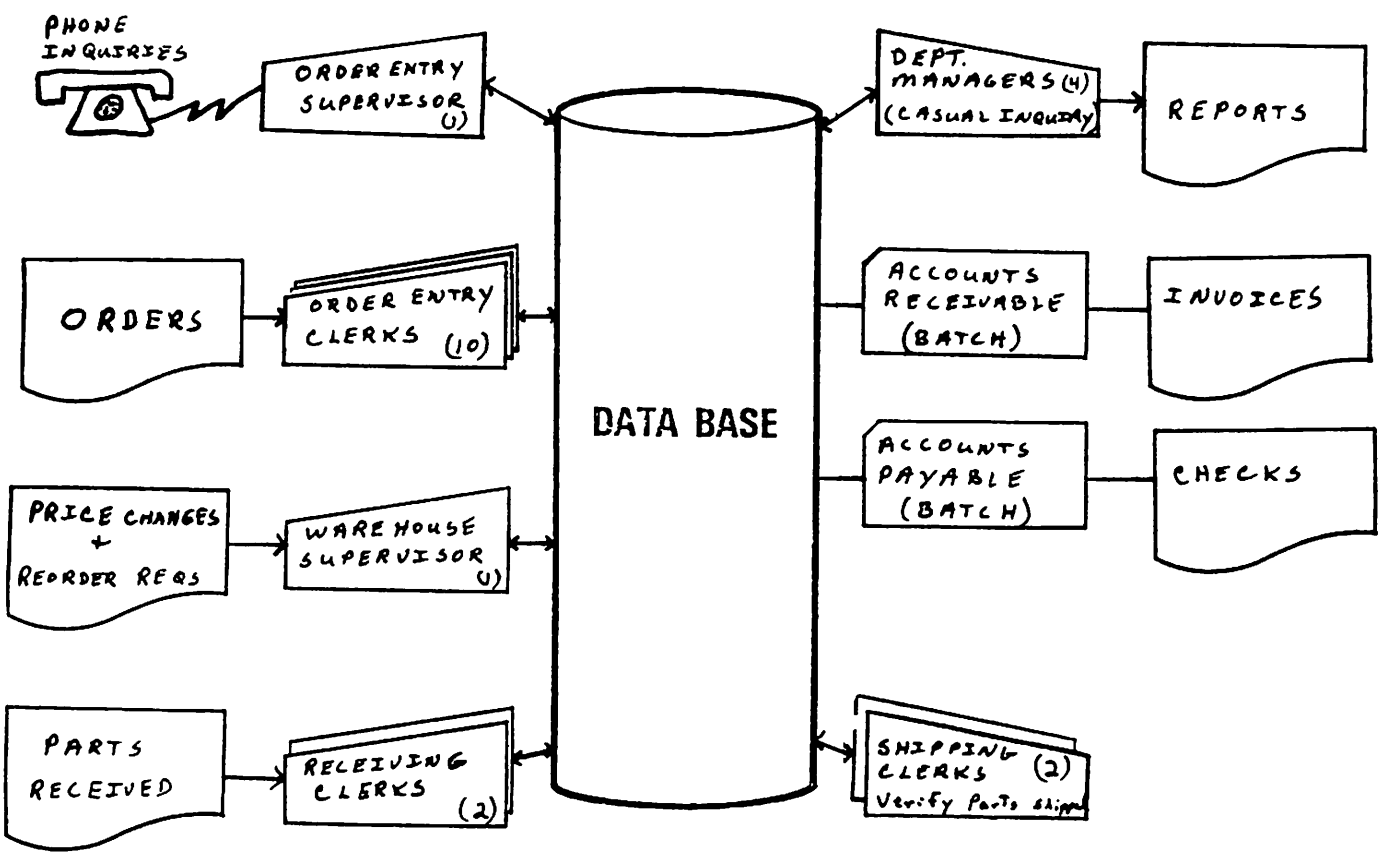
INDICATE, WITH SYMBOLS, THE DATA FLOW AND ACTIVITY AGAINST THE DATA BASE.



DESIGN AID #2

USER FUNCTION FLOW DIAGRAM

INDICATE, WITH SYMBOLS, THE DATA FLOW AND ACTIVITY AGAINST THE DATA BASE.



DESIGN AID #3

USER FUNCTIONS AND REQUIREMENTS

DATA BASE APPLICATION: _____

OBJECTIVES: _____ _____

_____ _____

_____ _____

USER NAME OR BATCH PROGRAM	DATA BASE FUNCTION OR ACTIVITY	NO. ONLINE TERMINALS	NO. OF I/O CHAR.	KEY INFORMATION IDENTIFIER	ESTIMATED TIME TO ENTER DATA	HOURLY THRUPUT TARGET	RESPONSE TIME TARGET

DESIGN AID #3

USER FUNCTIONS AND REQUIREMENTS

DATA BASE APPLICATION: ORDER PROCESSING/INV. CONTROL/ACCTS PAY + REC

OBJECTIVES: ENTER ORDERS; KEEP INVENTORY UPDATED; HANDLE PHONE INQUIRIES O/L; BILLING (ACCTS RECEIVABLE); KEEP PRICES & INVENTORY CURRENT; ACCTS PAYABLE

USER NAME OR BATCH PROGRAM	DATA BASE FUNCTION OR ACTIVITY	NO. ONLINE TERMINALS	NO. OF I/O CHAR.	KEY INFORMATION IDENTIFIER	ESTIMATED TIME TO ENTER DATA	HOURLY THRUPUT TARGET	RESPONSE TIME TARGET
O.E. SUPER	CHK ORDERS	1	100	ORDER-NO or COMP-ID	15-25 secs	LOW VOLUME	2-5 secs
O.E. CLERKS	ADD/UPDATE ORDERS	10	150	ORDER-NO or COMP-ID	35-50 secs	60/HOUR EACH	1-3 secs
WAREHOUSE SUPER	READ/UPDATE	1	100	ITEM-NO	15-25 secs	LOW VOLUME	2-5 secs
RECEIVING CLERKS	UPDATE INVENTORY	2	100	ITEM-NO	25-40 secs	60/HOUR EACH	1-3 secs
DEPT. MGRS	CASUAL INQ/ REPORTS	4	200	COMP-ID, ORDER-NO.	15-40 secs	LOW VOLUME	2-5 secs
SHIPPING CLERKS	VERIFY PARTS SHIPPED	2	100	ITEM-NO	15-30 secs	60/HOUR EACH	2-5 secs
ACCTS RECEIVABLE	READ AND PRINT BILLING INFORMATION (INVOICES)	—	—	COMP-ID	—	BATCH	—
ACCTS PAYABLE	PRINT CHECKS	—	—	COMP-ID	—	BATCH	—

DESIGN AID #4

ITEM/FUNCTION MATRIX

FUNCTIONS

ITEM NAME	TYPE/SIZE	SEARCH ITEM	SORT ITEM?	UPDATE FREQUENCY								

DESIGN AID #4

ITEM/FUNCTION MATRIX

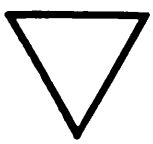
ITEM NAME	TYPE/SIZE	SEARCH ITEM	SORT ITEM?	UPDATE FREQUENCY	FUNCTIONS: OE SUPER	OE CLERK	WAREHOUSE SUPER	RECEIVING CLERK	DEPARTMENT MGRS	SHIPPING CLERK	ACCOUNTS REC/PA.
COMP-ID	X 8	X			X	X					X
CO-NAME	X 50				X	X					X
VAL	Z 10			HIGH	X						X
AMT-LD	Z 10			HIGH	X						X
ORD-NO	X 8	X				X				X	X
ITM-NO	X 8	X				X	X	X		X	X
CUR-CNT	Z 6			HIGH			X	X			
UNT-CST	Z 6			HIGH			X	X			
TAX	Z 6			HIGH			X	X			
REC-CDE	X 4						X				
PAC-CDE	X 4						X				
SHP-CDE	X 2						X				
SLS-PRC	Z 6						X				
DOL-LD	Z 10			HIGH			X				
ORD-DTE	X 6		X			X					
REC-DTE	X 6					X					
ORD-AMT	Z 10					X					
SHP-DTE	X 6		X							X	
NO-ITM	J									X	

ALL

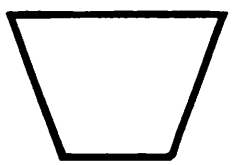
DESIGN AID #5

DATA BASE MODEL

MASTER DATA SETS



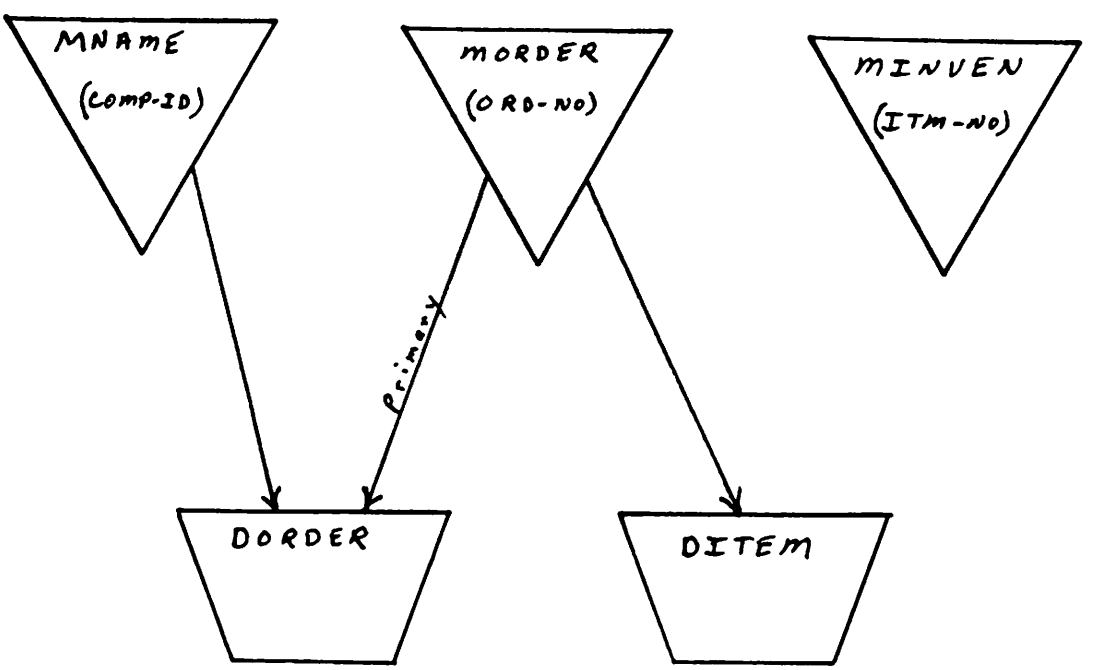
DETAIL DATA SETS



DESIGN AID #5
DATA BASE MODEL

MASTER DATA SETS

DETAIL DATA SETS



DESIGN AID #6

Data Base Directory

DATA BASE NAME _____

DATA SET NAME			ITEM		TYPE /	READ/WRITE	KEY	SORT	P/C	CAP	DESCRIPTION
#	MASTER	DETAIL		NAME	DESIG	SECURITY					

DESIGN AID #6

Data Base Directory

DATA BASE NAME "ORDENT"

DATA SET NAME			ITEM		TYPE / DESIG	READ/WRITE SECURITY	KEY	SORT	P/C	CAP	DESCRIPTION
#	MASTER	DETAIL	#	NAME							
1	MNAME				M					100	Customer Name File
			1	COMP-ID	X8		X		1		Truncated Company Name
			2	CO-NAME	X50						Full Company Name
			3	VAL	Z10						Value Current Orders
			4	AMT-LD	Z10						Account Receivables
2	MORDER				A					100	ORDERS OUTSTANDING
			5	ORD-NO	X8		X		2		Order Number
3	MINV EN				M					100	INVENTORY FILE
			6	ITM-NO	X8		X		0		Item Number
			7	CUR-CNT	Z6						No. of Items on Hand
			8	UNT-CST	Z6						Unit Cost of Item
			9	TAX	Z6						Total Tax on Item
			10	REC-CDE	X4						Reception by Inventory
			11	PAC-CDE	X4						Packing Code
			12	SHP-CDE	X2						Shipping Code
			13	SLS-PRC	Z6						Item Sales Price
			14	DOL-LD	Z10						Item Dollar Load
4		DORDER							2	100	CUSTOMER'S ORDERS CHAINS
			1	COMP-ID	X8		X				Truncated Company Name
			5	ORD-NO	X8		X2				Order Number
			15	ORD-DTE	X6			S2			Date Order Placed
			16	REC-DTE	X6						Date Order Received
			17	ORD-AMT	Z10						Amount of Order
5		DITEM							1	100	ITEMS CURRENTLY ON ORDER
			5	ORD-NO	X8		X				Order Number
			6	ITM-NO	X8						Item NoNumber
			18	SHP-DTE	X6			S			Item Shipping Date
			19	NO-ITM	J						Number of Items on Order

DESIGN AID #7
DETAILED ACTIVITY AGAINST THE DATA BASE
(IDEA SCRIPT FILES)
WORKSHEET

TRANSACTION INFORMATION									TRANSACTION STEPS								
USER	SCRIPT FILE NAME	USER TRANS NO.	NO PROCS (TERMS)	PAUSE MIN.	PAUSE MAX.	NO. ITERATIONS	NO. I/O CHARS	STACK SIZE	STEP NO.	DATA BASE NAME	OPEN MODE	DATA SET NAME	CALL TYPE	GET MODE	KEY NAME	INITIAL ARGUMENT VALUE	NO. OF GETS

DESIGN AID #7
DETAILED ACTIVITY AGAINST THE DATA BASE
(IDEA SCRIPT FILES)
WORKSHEET

TRANSACTION INFORMATION									TRANSACTION STEPS								
USER	SCRIPT FILE NAME	USER TRANS NO.	NO PROCS (TERMS)	PAUSE MIN.	PAUSE MAX.	NO. ITERATIONS	NO. I/O CHARS	STACK SIZE	STEP NO.	DATA BASE NAME	OPEN MODE	DATA SET NAME	CALL TYPE	GET MODE	KEY NAME	INITIAL ARGUMENT VALUE	NO OF GET
OE-CLERK	ORDSCRPT	66	10	35	50	50	150	4	1	ORDTST	1	MNAME	LOCK	—	—	—	—
									2	"		MNAME	PUT	—	COMP-ID	COMP6666	—
									3	"		DORDER	PUT		COMP-ID	COMP 6666	—
									4			DITEM	PUT		ORD-NO	ORD 6666	—
									5			—	UNLK	—	—	—	—
REC-CLERK		77	2	25	40	50	100	4	1	ORDTST	1	MINVEN	LOCK	—	—		—
									2	"	—	MINVEN	GETU	7	ITM-NO	ITM 777	—
									3		—	—	UNLK				—
SHP-CLERK		[illegible]	[illegible]	15	30	50	100	4	1	ORDTST	1	DITEM	LOCK		—		
									2			DITEM	GETU	5	ORD-NO	ORD888	5
									3			—	UNLK				

B-02.22

IMAGE DATA-BASE EVALUATIVE ANALYZER

A DATA BASE PERFORMANCE MEASUREMENT TOOL

OVERVIEW OF IDEA

- **ESTIMATES DATA BASE:**
  - LOAD TIME
  - RESPONSE TIME
  - THROUGHPUT
- **PROVIDES INITIAL DESIGN FEEDBACK**
- **PROVIDES FUTURE CHANGE IMPACT**
- **INTERACTIVELY PROMPTS YOU TO CREATE SCRIPT FILES**
- **GENERATES OWN TEST DATA**
- **EXECUTES FROM A SINGLE TERMINAL OR MULTIPLE TERMINALS**
- **PRODUCES SUMMARY OR DETAILED REPORTS**
  - THROUGHPUT RATES
  - RESPONSE TIMES
- **RESIDES IN THE HEWLETT-PACKARD CONTRIBUTED LIBRARY**
- **RUNS ON THE HP 3000 SERIES I, II, AND III COMPUTERS**

IDEA REQUIREMENTS

- DESIGN MUST BE COMPLETE
- DATA BASE MUST BE CREATED (EMPTY)
- UNDERSTANDING OF THE DATA BASE APPLICATION
- UNDERSTANDING OF IMAGE CALL PROCEDURES
- DETAILED ACTIVITY DETERMINED AGAINST THE DATA BASE (SCRIPT FILES)

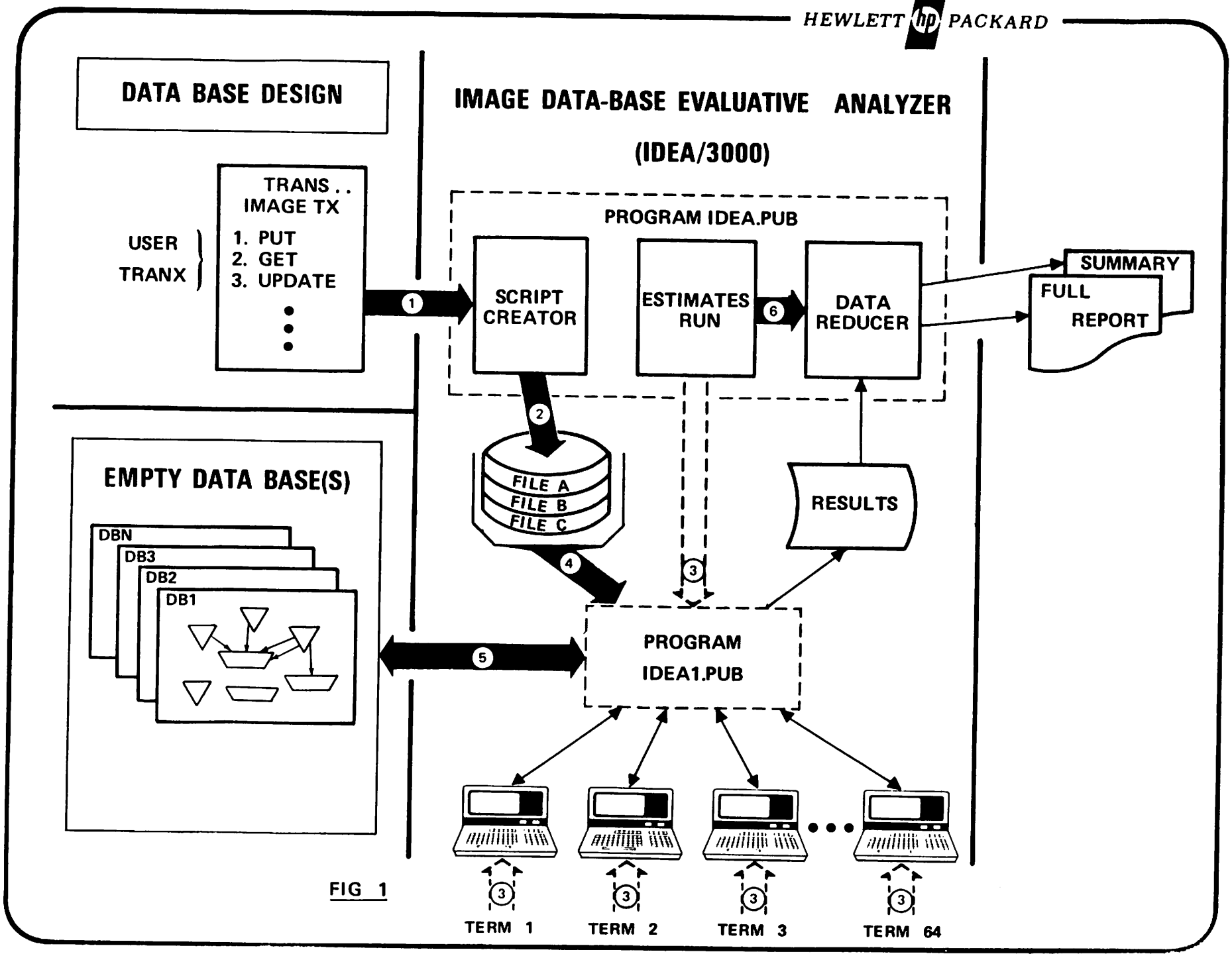

DATA BASE DESIGN
IMAGE DATA-BASE EVALUATIVE ANALYZER
(IDEA/3000)
TRANS . .
IMAGE TX
1. PUT
2. GET
3. UPDATE
USER
TRANX
PROGRAM IDEA.PUB
SCRIPT CREATOR
ESTIMATES RUN
DATA REDUCER
SUMMARY
FULL REPORT
EMPTY DATA BASE(S)
DBN
DB3
DB2
DB1
FILE A
FILE B
FILE C
RESULTS
PROGRAM IDEA1.PUB
TERM 1
TERM 2
TERM 3
TERM 64

FIG 1

SAMPLE SCRIPT CREATION

```
:RUN IDEA


SELECT:(IDEA, Version 3, May 1978)
1 = CREATE SCRIPT
2 = LOAD RUN
3 = TEST RUN
4 = TIMING RUN
5 = REPORT
6 = ESTIMATE LOAD TIME
7 = ERASE DATA BASE
8 = LIST SCRIPTS
EXIT
1
RECORD NUMBER 1
01  TRANSACTION NUMBER(0<N<97)? 66
02  NUMBER OF PROCESSES? 10
03  MINIMUM PAUSE: 32
04  MAXIMUM PAUSE? 50
05  NUMBER OF ITERATIONS: 25
06  NUMBER OF I/O-CHARS: 150
07  STACK SIZE(0 TO 16K)? 4
WHICH LINE IS INCORRECT(NONE-CR)? C/R

RECORD NUMBER 2
01  TRANSACTION NUMBER  66
STEP NUMBER 01
02  DATA-BASE NAME(REF) =  ?
02  DATA-BASE NAME(REF) = ORDTST
03  OPEN-MODE? 1
04  DATASET NAME= ?
04  DATASET NAME= MASTER
05  GET,GETU,GETD,PUT,LOCK OR UNLK: LOCK
WHICH LINE IS INCORRECT(NONE-CR)? C/R

RECORD NUMBER 3
01  TRANSACTION NUMBER  66
STEP NUMBER 02
02  DATA-BASE NAME(REF) =  ?  ORDTST
02  DATA-BASE NAME(REF) =  C/R
04  DATASET NAME= ? MASTER
04  DATASET NAME=  C/R
05  GET,GETU,GETD,PUT,LOCK OR UNLK: PUT
07  KEY NAME? CUMP-ID
08  INITIAL VALUE(AT MOST 20 CHARS)? CUMP6666
WHICH LINE IS INCORRECT(NONE-CR)? C/R

RECORD NUMBER 4
01  TRANSACTION NUMBER  66
STEP NUMBER 03
02  DATA-BASE NAME(REF) =  ?
02  DATA-BASE NAME(REF) =  C/R
04  DATASET NAME= ? MASTER
04  DATASET NAME= ORDERS
05  GET,GETU,GETD,PUT,LOCK OR UNLK: PUT
07  KEY NAME? CUMP-ID
08  INITIAL VALUE(AT MOST 20 CHARS)? CUMP6666
WHICH LINE IS INCORRECT(NONE-CR)? CR
```

SAMPLE SCRIPT FILE

```
661000350050000015
6600000000000000000  ORDTST  00XACT
6600000000000000000  ORDTST  01LOCKMNAME                                04**********************0150
6600000000000000000  ORDTST  01PUT MNAME                                01**********************0001
6600000000000000000  ORDTST  01PUT DORDER      COMP-ID                  01COMP6666                0001
6600000000000000000  ORDTST  01PUT DITEM       COMP-ID                  01COMP6666                0001
6600000000000000000  ORDTST                   ORD-NO                   01ORD6666                 0001
770200250040000015                            00XACT                   04**********************0100
7700000000000000000  ORDTST  01LOCKMINVEN                               01**********************0001
7700000000000000000  ORDTST  01GETUMINVEN                               07ITM777                  0001
7700000000000000000  ORDTST  01UNLKMINVEN                               01**********************0001
880200150030000015                            00XACT                   04**********************0100
8800000000000000000  ORDTST  01LOCKDITEM                                01**********************0001
8800000000000000000  ORDTST  01GETUDITEM       ORD-NO                   05ORD888                  0005
8800000000000000000  ORDTST  01UNLKDITEM                                01**********************0001
```

SAMPLE OF IDEA OPTIONS

```
:RUN IDEA


SELECT:(IDEA, Version 3, May 1976)
1 = CREATE SCRIPT
2 = LOAD RUN
3 = TEST RUN
4 = TIMING RUN
5 = REPORT
6 = ESTIMATE LOAD TIME
7 = ERASE DATA BASE
8 = LIST SCRIPTS
EXIT
(2)

SCRIPT FILE NAME? (ORDSCRPT)
STARTING MASTER LOAD PROCESS
STARTING DETAIL LOAD PROCESS
LOADING COMPLETE

SELECT:(IDEA, Version 3, May 1976)
1 = CREATE SCRIPT
2 = LOAD RUN
3 = TEST RUN
4 = TIMING RUN
5 = REPORT
6 = ESTIMATE LOAD TIME
7 = ERASE DATA BASE
8 = LIST SCRIPTS
EXIT
(3)

SCRIPT FILE NAME? ORDSCRPT
TRANSACTION 66 CHECKED
TRANSACTION 77 CHECKED
TRANSACTION 88 CHECKED
TEST RUN O.K.  DO YOU WANT A TIMING RUN(Y/N)? (N)


SELECT:(IDEA, Version 3, May 1976)
1 = CREATE SCRIPT
2 = LOAD RUN
3 = TEST RUN
4 = TIMING RUN
5 = REPORT
6 = ESTIMATE LOAD TIME
7 = ERASE DATA BASE
8 = LIST SCRIPTS
EXIT
(4)

SCRIPT FILE NAME? (ORDSCRPT)
 14 PROCESSES REQUIRED
12345678901234
STARTING TIMING RUN
END OF TIMING RUN
SORTING IDEALOG
STARTING REPORT
```

SAMPLE OF IDEA OPTIONS (Cont.)

```
1

SELECT:(IDEA, Version 3, May 1976)
1 = CREATE SCRIPT
2 = LOAD RUN
3 = TEST RUN
4 = TIMING RUN
5 = REPORT
6 = ESTIMATE LOAD TIME
7 = ERASE DATA BASE
8 = LIST SCRIPTS
EXIT
6
WHICH DATA BASE? ORDTST

APPROX. LOAD/DBLOAD TIME =  0  HRS    21  MINS

NOTE:  This is an estimated time
based on 0.15 sec/PUT and 0.1 sec
per update per chain in a DETAIL.
Chain sorting times are excluded.


SELECT:(IDEA, Version 3, May 1978)
1 = CREATE SCRIPT
2 = LOAD RUN
3 = TEST RUN
4 = TIMING RUN
5 = REPORT
6 = ESTIMATE LOAD TIME
7 = ERASE DATA BASE
8 = LIST SCRIPTS
EXIT
7
WHICH DATA BASE? ORDTST
ERASED

SELECT:(IDEA, Version 3, May 1978)
1 = CREATE SCRIPT
2 = LOAD RUN
3 = TEST RUN
4 = TIMING RUN
5 = REPORT
6 = ESTIMATE LOAD TIME
7 = ERASE DATA BASE
8 = LIST SCRIPTS
EXIT
8
LISTING O..
3

SCRIPT FILE NAME? ORDSCRPT
TRANSACTION 66 CHECKED
```

SAMPLE IDEA OUTPUT

MON. OCT 16, 1978 8:32 AM
IDEA V.0 SERII

SUMMARY REPORT
(AVERAGES)

PAGE 1

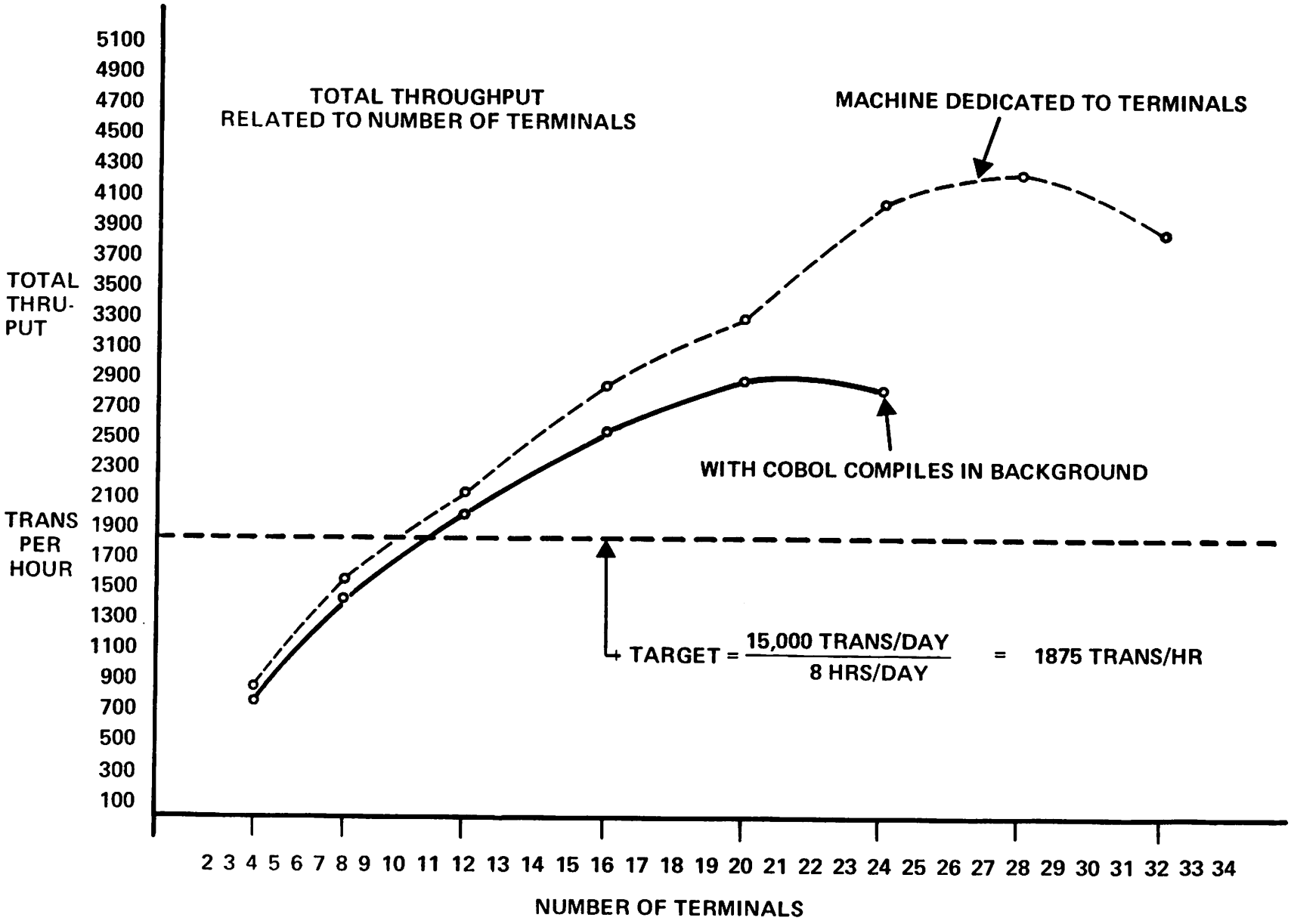
XACT/ PROCESS	NO OF TIMES	RECORDING TIME (A)	THINK-TIME	TOTAL-DELAY (B)	RESPONSE TIME (C)	CYCLE-TIME (A+B+C)	TRANSACTIONS PER HOUR
66/ 1	24	.0 SECS	43.1 SECS	43.9 SECS	.9 SECS	44.9 SECS	80
66/ 2	26	.0 SECS	41.5 SECS	41.8 SECS	.5 SECS	42.4 SECS	85
66/ 3	24	.0 SECS	43.8 SECS	44.3 SECS	1.0 SECS	45.4 SECS	79
66/ 4	25	.0 SECS	42.1 SECS	42.3 SECS	.6 SECS	43.0 SECS	84
66/ 5	25	.0 SECS	42.9 SECS	43.3 SECS	.8 SECS	44.2 SECS	81
66/ 6	25	.0 SECS	42.2 SECS	42.6 SECS	.9 SECS	43.5 SECS	83
66/ 7	25	.0 SECS	42.2 SECS	42.4 SECS	1.7 SECS	44.2 SECS	81
66/ 8	25	.0 SECS	41.7 SECS	42.4 SECS	.6 SECS	43.1 SECS	84
66/ 9	25	.0 SECS	42.5 SECS	43.2 SECS	.9 SECS	44.3 SECS	81
66/10	25	.0 SECS	42.3 SECS	42.7 SECS	.8 SECS	43.6 SECS	83
77/ 1	36	.0 SECS	32.6 SECS	32.8 SECS	.9 SECS	33.8 SECS	107
77/ 2	36	.0 SECS	32.8 SECS	33.0 SECS	.8 SECS	33.9 SECS	106
88/ 1	47	.0 SECS	21.8 SECS	22.0 SECS	.9 SECS	23.0 SECS	157
88/ 2	44	.0 SECS	23.3 SECS	23.7 SECS	1.0 SECS	24.8 SECS	145

			AVG RESPONSE-TIME FOR TERMINAL MIX		.9 SECS		

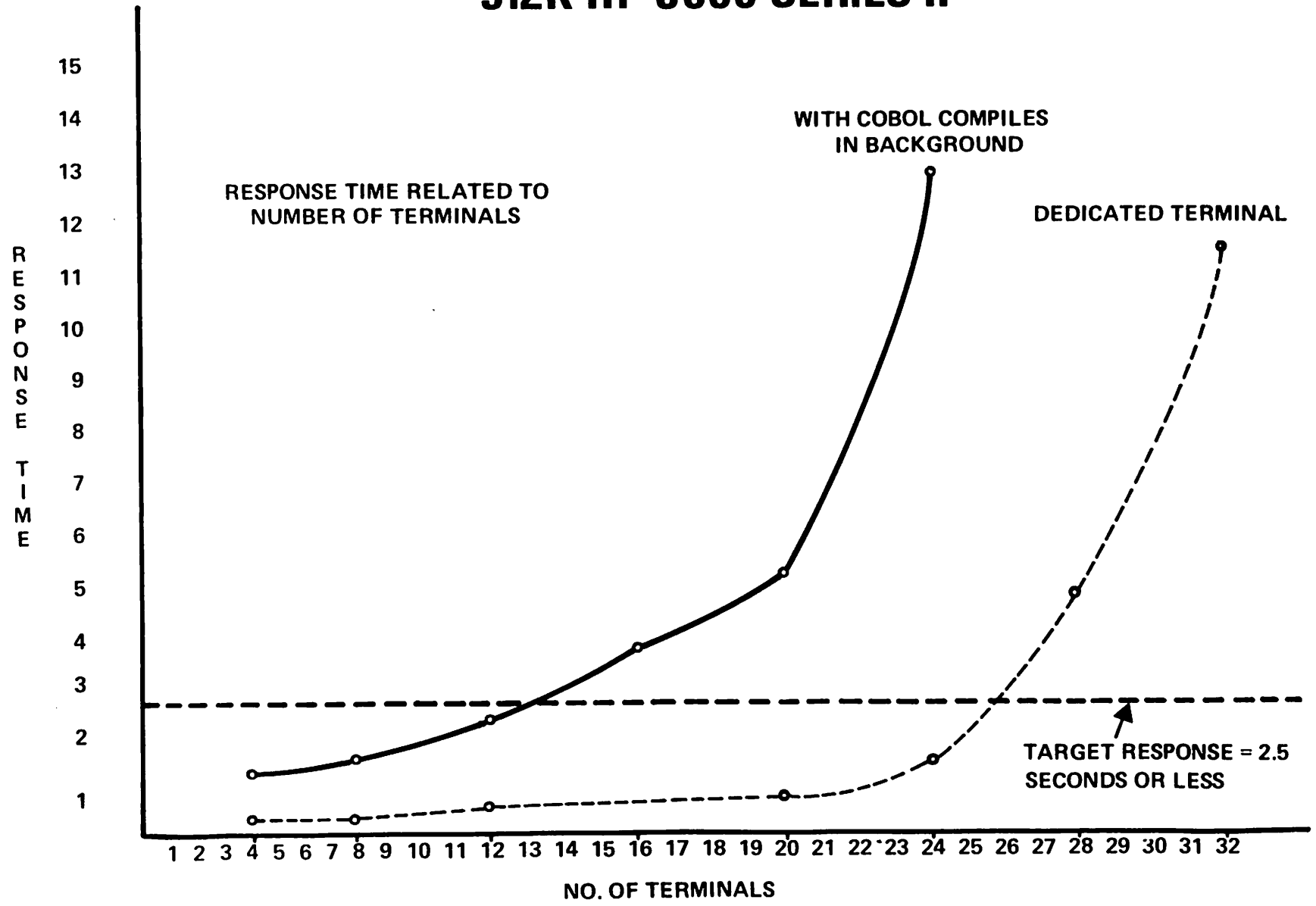
			TOTAL HOURLY THRU-PUT FOR THE MIX				1336

HEWLETT hp PACKARD

512K HP 3000 SERIES II



512K HP 3000 SERIES II



DISADVANTAGES OF PERFORMANCE MEASUREMENT

- **DOES NOT TAKE INTO CONSIDERATION THE APPLICATION OVERHEAD SUCH AS:**
  - APPLICATION PROGRAM SOURCE LANGUAGE
  - DATA EDITING ROUTINES
  - NUMBER CRUNCHING
- **DATA BASE IS USUALLY NOT FULLY LOADED**
- **DATA BASE ACTIVITY IS USUALLY "BEST GUESS" SITUATION**

ADVANTAGES OF PERFORMANCE MEASUREMENT

- NOT A SIMULATION - PERFORMS THE ACTUAL DISK I/O
- FEASIBILITY OF DESIGN MAY BE DETERMINED WITHOUT WRITING APPLICATION PROGRAMS
- DEDICATED COMPUTER SYSTEM NOT REQUIRED
- AIDS IN DETERMINING HARDWARE CONFIGURATION REQUIREMENTS
- ESTIMATES "BEST CASE" THROUGHPUT AND RESPONSE TIME
- MAY BE USED TO MODEL CURRENT DATA BASE ACTIVITY AND THEN THE IMPACT OF FUTURE DESIGN CHANGES

SUMMARY

- DATA BASE DESIGNER COULD BENEFIT FROM A COMMON DATA BASE DESIGN METHODOLOGY

- DATA BASE DESIGNERS NEED A TOOL TO MEASURE PERFORMANCE BEFORE APPLICATION IMPLEMENTATION

- HEWLETT-PACKARD HAS BOTH !

IMAGE DATA BASE DESIGN AIDS

INTRODUCTION

One of the concerns of the data base designer is the lack of a data base design methodology. The following data base design aids have been developed to provide some direction and a logical approach to assist the data base designer in this most important phase in the implementation of a data base.

The approach taken here is by no means the only way to design a data base. It is simply a logical approach starting by gathering the general overall information about the host computer system and the data base applications and then working down to the detail of the actual DBMS calls against the data base. The main benefit of using these design aids is that they force the designer to gather the pertinent information related to the data base application so intelligent design decisions can be made.

It is assumed that the users of these design aids have a good understanding of IMAGE.

DATA BASE DESIGN AIDS

Preliminary Investigations

Before beginning the data base design process, the data base designer should first become familiar with the organizational structure, business functions, and objectives of the company and the specific group for whom the data base is being designed. This is very important especially if this data base will be used by upper or functional management to assist in the day to day decision making process.

Design Aid #1 - Computer/Data Base Environment

The purpose of this design aid is to get an overall picture of the environment in which the data base will be residing. This aid provides information on the company, the computer hardware, and the major activities or applications that will be using the computer system and the data base.

Design Aid #2 - User Function Flow Diagram

The purpose of this design aid is to identify the data base users and their respective functional activities relating to a specific data base. The forms used and the number of users invovled in each type of function are shown on a flow diagram. This design aid is useful because it provides an indication of the overall activities being applied against a data base.

The key point to remember here is that each arrow touching the side of the data base symbol represents a transaction against the data base. This transaction may consist of a single DBMS call or a series of DBMS calls. There will be an opportunity to identify these specific calls later, in design aid number 7, after the data base design has been established.

Design Aid #3 - User Functions and Requirements

The purpose of this design aid is to describe the data base functions in more detail. The objectives of the data base should be listed to remind the designer of what the data base is to provide.

Each of the arrows (functions) touching the side of the data base symbol in design aid #2 should now be described further. This includes identifying each user or batch program and describing the data base function or activity, the number of online terminals, the number of I/O characters transmitted to and from the terminals, the key information identifier (the major search item that a data base user will utilize to identify or locate a data base record, e.g., order number, part number, etc.), the estimated time to enter the data, the hourly throughput target, and the response time target.

Design Aid #4 - Item/Function Matrix

The primary purpose of this design aid is to identify all the items in the data base and then relate them to the functions described in the preceding design aids.

In addition, the item type and size are included because these are useful later in the data base directory. Also, indicating whether the item is a search item (key) is important because the search item is the basis for a manual or automatic master data set. Furthermore, indicating that an item is a sort item assumes a detail data set chain will be sorted by that item. The update frequency of an item can be useful to identify the high activity items which can affect the design of the data base by possibly locating those high activity items together in the same data set.

This is the last design aid before the actual data base design decisions are made. Additional data usage statistics may be appropriate before the final design decisions are made.

Design Aid #5 - Data Base Model

The purpose of this design aid is to provide a visual representation of the data base by showing the data set relationships. This is where the design decisions are made! The data base designer must now "earn his keep" by assimilating all the information gathered in the previous design aids as well as drawing on prior data processing experience to come up with a design that satisfies the needs of the data base users.

If there are any words of wisdom to assist the designer in making these important design decisions, they are, "There is no perfect design!". The data base designer is usually well aware of the fact that the needs and objectives of a company or organization within a company are constantly changing.

Design Aid #6 - Data Base Directory

The purpose of this design aid is to provide a means of writing down the structure of the data base. Later, this structure can easily be translated into the data base SCHEMA using the TEXT EDITOR.

The information requested is self explanatory.

Design Aid #7 - Detailed Activity Against the Data Base

The purpose of this design aid is to indicate the detailed activity against the data base for the purpose of identifying the DBMS calls required to complete a transaction. This information alone may help the experienced designer to estimate the load on the system. This design aid is primarily used as a worksheet for a performance measurement tool called IDEA which is an acronym for IMAGE Database Evaluative Analyzer.

Two versions of IDEA (Series I and Series II) which were written by HP System Engineers are currently available in the Contributed Library (the Series II version will work with the Series III). A new enhanced version of IDEA has been made available to our field system engineers who specialize in performance measurement consulting.

NEW FEATURES AND LIMITATIONS OF IDEA VERSION #3

1. Each transaction can reference more than 1 data base.

2. Keys must be of type U or X with a maximum length of 254 bytes.

3. Within any transaction, the product of NUMBER OF ITERATIONS and NUMBERS of PROCESSES must be less than 32768.

4. IDEA permits simulation of up to 60 terminals. The actual limit may be less due to the number of data bases and data sets involved and to the system configuration.

5. The maximum record size is 512 words.

6. Modifying the script file must be done under the EDITOR. Remember to KEEP the modified file UNNUMBERED.

7. TIMING RUN processes do not "give up" at the first functional failure. They perform "retries" designed to force success so that they may continue with minimal impact on performance. If, for example, a directed GET fails, the data set is "rewound" and a serial GET performed. This fails only if the data set is empty, in which case the TIMING RUN is terminated.

   NOTE: These "retries" are logged and appear on the REPORT.

8. For each directed GET, a random number between 1 and 100 is used as the address of an entry to be a read. The probability that a "retry" will be required depends, in this case, on the capacity and fullness of the data set being accessed.

9. The processing which handle a transaction are created with a user specified stack size.

10. IDEA can be run remotely.

11. Multiple data bases may be accessed within a single transaction.

12. Key lengths can exceed 20 characters.

13. Mode 2,4, and 5 "GETS" (with or without update or delete) can be multiple GETS.

14. No longer necessary to always perform a data base load.

15. The loading is about 7 times as fast.

16. Data bases can be erased by IDEA.

17. The only "files" the user has to "build" are the data bases.

18. A test run can be made to check the script.

19. A "firsttime" pass through the script is also performed by each process prior to proceeding to the timing portion of the timing run. Puts and deletes are bypassed during this pass. IDEA monitors this on the screen by displaying one of the digits 1,2,3,4,5,6,7,8,9,0 cyclically, and in that order, each time a newly activated process completes this first-time pass. It only creates and activates another one after the preceeding one has performed successfully.

20. IDEA handles all files and data bases so that none of them are left lying around. If the program should abort for any reason, this will generally not be true.

21. The timing processes all terminate when any one of them terminates. They do this without logging any more timing records.

22. An impatient user can also force early termination by entering Control-Y at any time after the timing run has begun. The response to the Control-Y may be quite slow due to the resource cleanup which transpires.

23. The logging file (IDEALOG) is sorted only once into the sort file (IDEASORT)/ Both files have 108 byte records with a blocking factor of 7.

24. The timing processes all close the input script file to release the system resources tied up by leaving them open.

25. The timing processes append share the IDEALOG file in multi-access mode. This minimizes the number of resources needed to support the logging function.

26. IDEA and the timing processes communicate via a job control word (JCW). Local rins are used to control access to the user's terminal, and the JCW when writing II.

27. A local rin is also used to queue up each timing process until they all have been successfully activated.

28. The timing process does not give up at the first functional failure; it performs recovery style retries suitable to the function. Directed gets, for example, are implemented by the generation of a random number between 1 and 100 which is used as the address of the record to be read. If this fails, for any reason, the data set is "rewound" and a serial get is performed. This will succeed unless the data set is empty, in which case the timing run is terminated.

Helpful Hints

1. It is best to run IDEA stand-alone. This permits you to obtain timing data not impacted by other processes. It also maximizes the probability that the system resources required for a given TIMING RUN will be avaiable.

2. If you wish to test a given script with a varying number of processes, start with the maximum and modify the script for lower values on subsequent runs. In this manner, the starting script can be used to LOAD the data base once so that all subsequent runs can be performed without reloading.

3. Scripts with PUTs and/or DELETEs are likely to encounter problems during TIMING RUNs which may lead to early termination. For PUT scripts, problems include full data sets, duplicate masters, absence of a required chain head.

4. For DELETE scripts, problems include empty data sets or attempting to delete a master with related detail entries.

5. Processes are launched for each transaction in the same order as the transactions are defined in the script. By entering the transactions with the slowest cycle time (including THINK time) first, all processes will get into play as early as possible. This will make the resulting statistics most meaningful and with a minimum number of iterations.