

Sales Application Manual

... FOR DATA BASE SYSTEMS



TABLE OF CONTENTS

1. DATA BASE DESIGN	1
2. IMAGE 2100	2
3. TYPICAL APPLICATIONS	3
A. Equipment Records	
B. Accounts Receivable	
C. Freight Movement	
D. Back Order Files	
E. Insurance Records	
F. Materials Explosion	
G. Accounts Payable	
H. Credit	
4. REPORTS	4
5. PROCEDURES	5
6. EXPANDED MODE	6
7. MULTI-TERMINAL	7
8. COMPETITION	8
9. HOW TO ORDER	9

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

INFORMATION

MR. DALY

14762

PRECISION PARTS DIVISION

LATHE OPERATOR



Figure A-1

CONTENT

Class

Data

name	Daly
employee number	14762
location	precision parts div.
job	lathe operator
social security number	162-26-9157
number of deductions	4

•
•
•

•
•
•

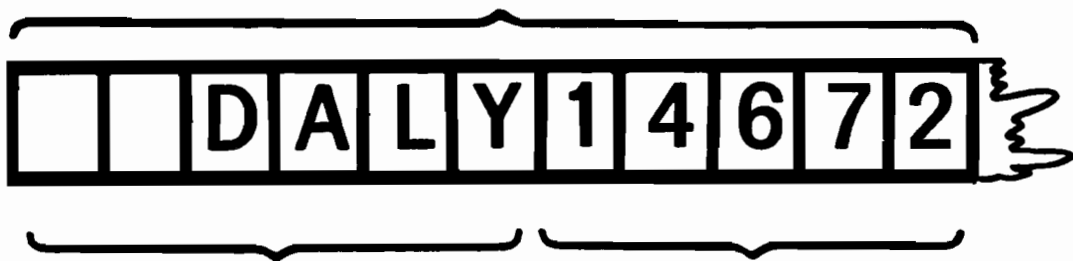
Figure A-2

FIELD

- A SET OF CONTIGUOUS LOCATIONS WITHIN A RECORD
- STORAGE FOR A PARTICULAR TYPE OF DATA

Figure A-3

RECORD



“name”
field

“employee no.”
field

Figure A-4

RECORD

- **UNIT OF DATA REFERENCED**
- **CONSISTS OF A SET OF CONTIGUOUS LOCATIONS (FIELDS)**

Figure A-5

FILES (DATA SETS)

- **A COLLECTION OF RECORDS HAVING LOGICAL INTERRELATIONSHIPS**
- **INTERRELATIONSHIPS ARE SPECIFIED BY LOGICAL STRUCTURE DATA**

Figure A-6

INDEX DATA

- BASIS FOR DISTINGUISHING RECORDS OF THE FILE (DATA SET)
- THE DATA IN THE RECORD THAT IS TESTED FOR SELECTION

Figure A-7

INDEX DATA

NAME ————— Daly

employee number

LOCATION ————— precision parts div.

JOB ————— lathe operator

social security number
number of deductions

•
•
•

Figure A-8

STRUCTURE DATA

- DATA ADDED TO A RECORD TO RELATE IT TO OTHER RECORDS
- THE "CHAINS" OR "LINKS" THAT ORGANIZE RECORDS INTO GROUPS

Figure A-9

STRUCTURE DATA

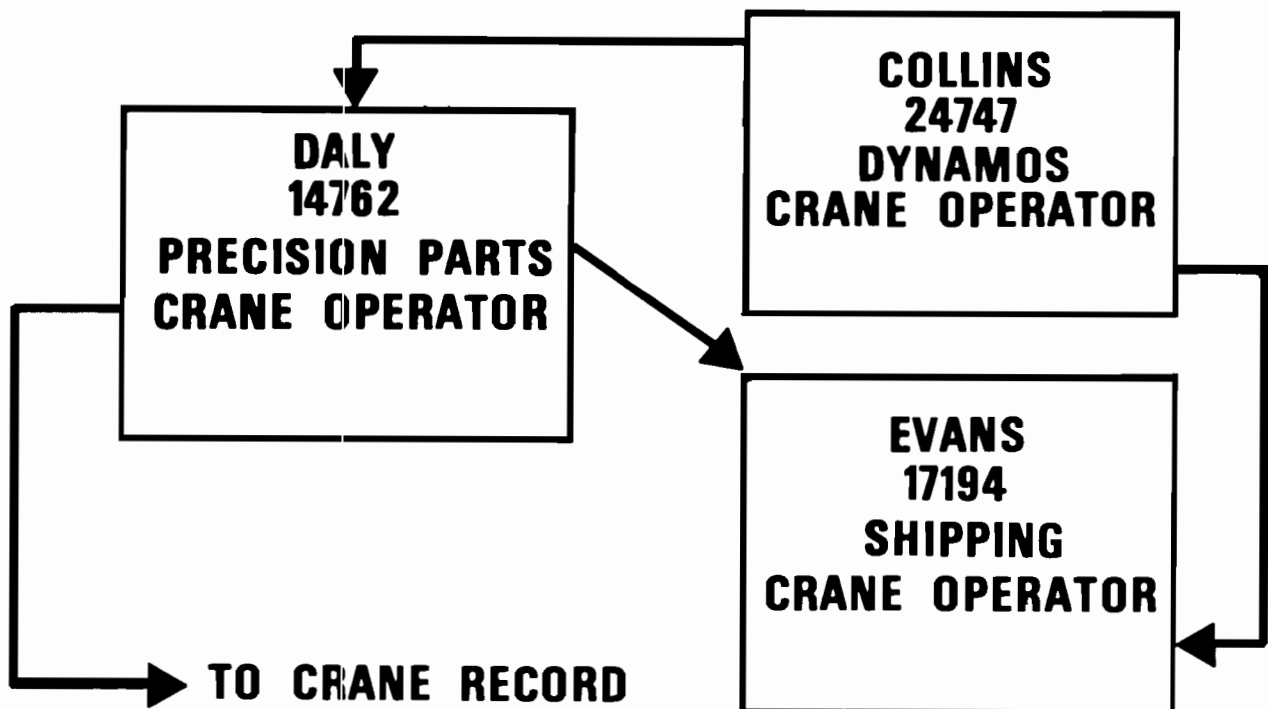


Figure A-10

DATA BASE

**A COLLECTION OF FILES (DATA SETS)
HAVING LOGICAL INTERRELATIONSHIPS**

Figure A-11

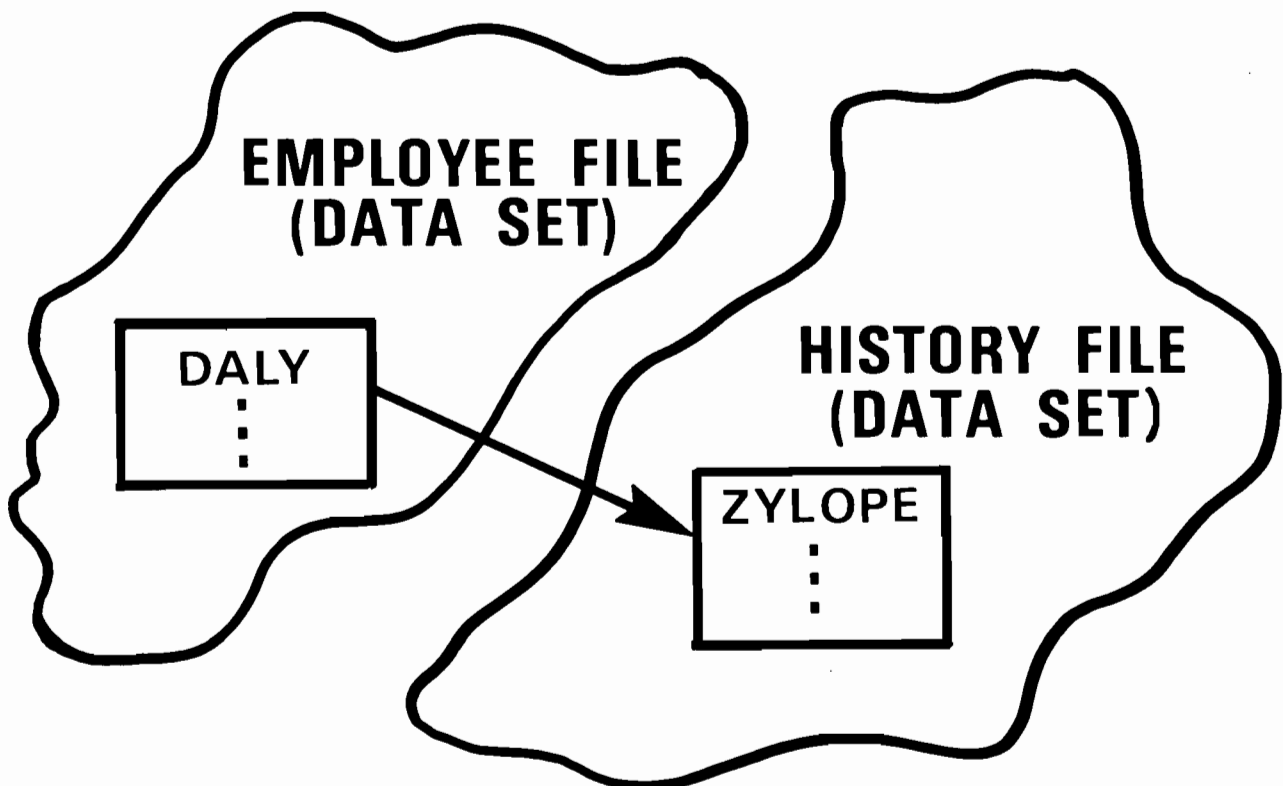


Figure A-12

KEY

**ONE OF THE SET OF AVAILABLE
ATTRIBUTES THAT MAY BE USED
IN A QUERY TO REFERENCE A FILE**

Figure A-13

KEY VALUE

**A SPECIFIC DATA VALUE FOR A KEY
TO BE USED IN SELECTING RECORDS
FROM THE FILE(S)**

Example:

KEY = PART

KEY VALUE = 2767A-001

Figure A-14

INDEXING TECHNIQUES

- **TABULAR**
- **CALCULATING**

Figure A-15

TABULAR INDEXING

File		Table	
DEWITT	P1	WILLIAMS	P2
WILLIAMS	P2	EVANS	P5
DALY	P3	HELLER	P4
HELLER	P4	DEWITT	P1
EVANS	P5	COLLINS	P6
COLLINS	P6	DALY	P3

Figure A-16

CALCULATED INDEXING

A MATHEMATICAL FUNCTION OF THE KEY SPACE INTO THE DATA SPACE

Figure A-17

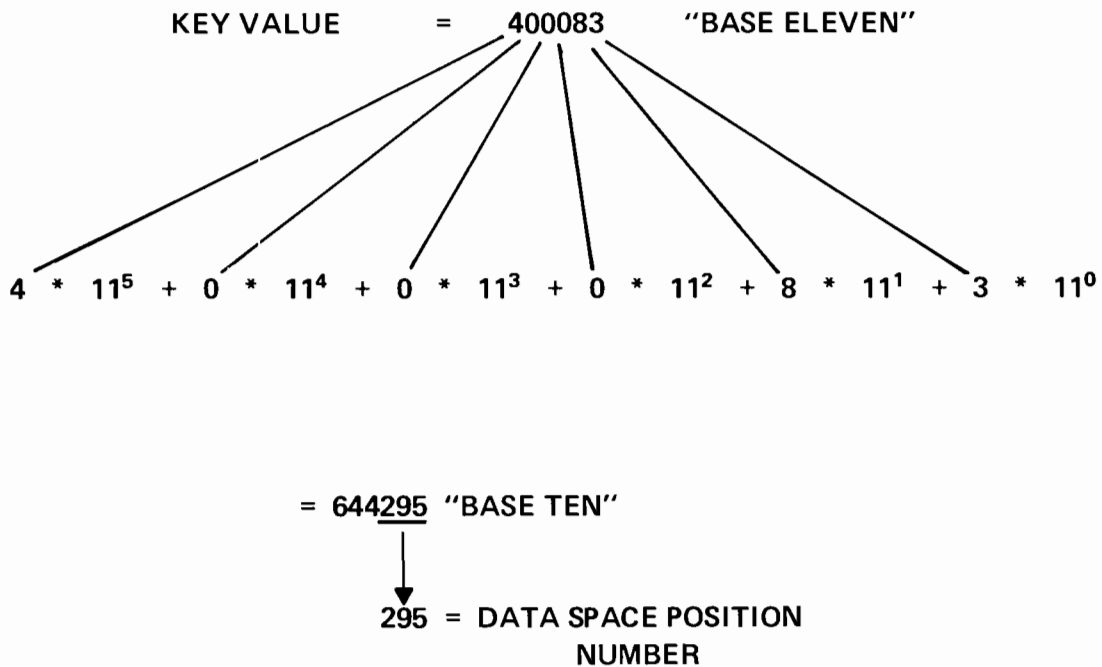


Figure A-18

Calculated Indexing and Conflicts

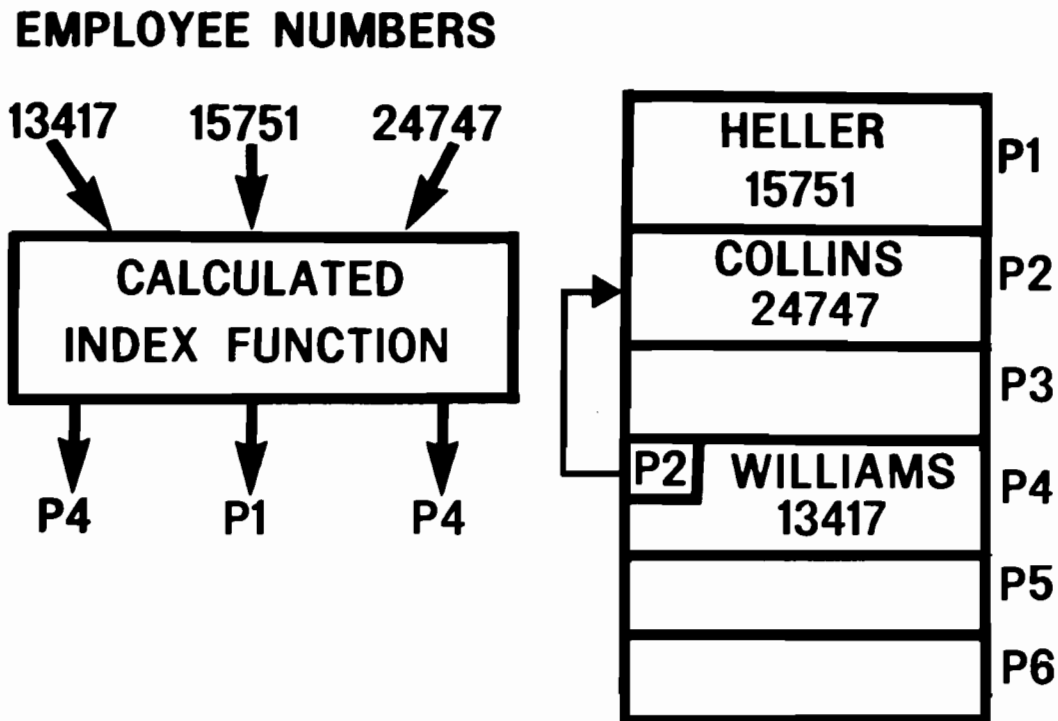


Figure A-19

QUERY

- **THE FORMAL STATEMENT OF A RETRIEVAL REQUEST**
- **SUPPLIES THE BASIC PARAMETERS FOR SEARCHING THE DATA BASE AND SELECTING RECORDS**

Figure A-20

QUERY EXAMPLE

Find all of the **LATHE**
operators working
in the **PRECISION PARTS**
division

Query
Parameters

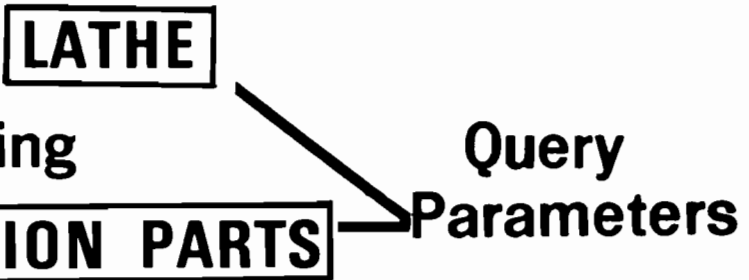


Figure A-21

KEYS & KEY VALUES

Find all **LATHE** → { Key=Job
Value=LA
operators working in
PRECISION PARTS → { Key=location
value=PP
division

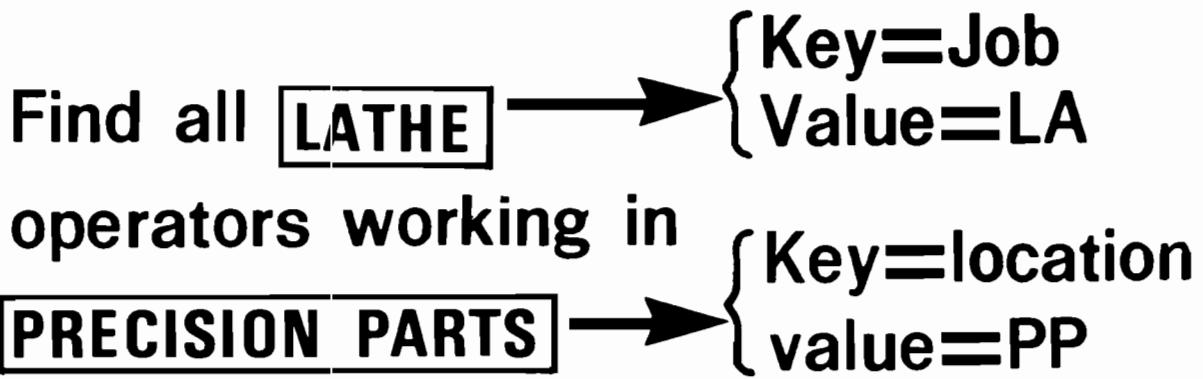


Figure A-22

SIMPLE QUERY

"FIND ALL ZYLOPE CORP. ORDERS"

<u>KEY</u>	<u>KEY VALUE</u>
CLIENT	ZYLOPE

FIND ALL (CLIENT=ZYLOPE) ORDERS

DATA BASE REQUIREMENT

- 1 – ORDER RECORDS HAVE CLIENT ATTRIBUTE
- 2 – CLIENT ATTRIBUTE IS INDEXED

Figure A-23

QUERY WITH CONJUNCTION

"HAS THE ZYLOPE CORP. ORDER FOR TRIVETS
BEEN SHIPPED?"

<u>KEYS</u>	<u>KEY VALUES</u>
CLIENT	ZYLOPE
ORDER ITEM	TRIVETS
STATUS	SHIPPED

has (CLIENT=ZYLOPE) order for
(ORDER-ITEM=TRIVETS) been
(STATUS=SHIPPED)?

- 1 – find all records such that CLIENT=ZYLOPE
and ORDER-ITEM=TRIVETS
- 2 – for each such record, is STATUS=SHIPPED

Figure A-24

A QUERY PROGRAM MUST:

- 1 – IDENTIFY THE KEYS AND KEY VALUE PAIRS
- 2 – DEFINE THE RELATIONSHIPS ON THEM
- 3 – ORDER THE UTILIZATION OF THE SEARCH FACILITY
- 4 – PROCESS EACH RECORD THAT RESULTS

Figure A-25

DESIGN QUESTIONS

- WHAT INFORMATION MUST BE STORED?
- HOW SHOULD THIS INFORMATION BE STORED?
- WHAT INTERRELATIONSHIP MUST EXIST BETWEEN UNITS OF THE INFORMATION?
- HOW WILL THE INFORMATION BE RETRIEVED?

Figure A-26



IMAGE 2100 MODULES

DBDS	DEFINITION SUBSYSTEM USED TO DEFINE YOUR DATA BASE
DBUS	UTILITY SUBSYSTEM USED TO BUILD, BACK-UP, AND RESTORE YOUR DATA BASE
DBMS	MANAGEMENT SUBSYSTEM USED TO ACCESS YOUR DATA BASE
DBQS	QUERY SUBSYSTEM USER INTERFACE

*NO PROGRAMMING

Figure B-1

IMAGE 2100 COMPARISON OF TERMS

• DATA BASE	DATA BASE
• DATA SET	FILE
• DATA ENTRY	DATA RECORD + LINKS
• DATA RECORD	DATA RECORD
• DATA ITEM	FIELD OR VARIABLE

Figure B-2

A QUERY PROGRAM MUST:

- 1 – IDENTIFY THE KEYS AND KEY VALUE PAIRS
- 2 – DEFINE THE RELATIONSHIPS ON THEM
- 3 – ORDER THE UTILIZATION OF THE SEARCH FACILITY
- 4 – PROCESS EACH RECORD THAT RESULTS

Figure A-25

DESIGN QUESTIONS

- WHAT INFORMATION MUST BE STORED?
- HOW SHOULD THIS INFORMATION BE STORED?
- WHAT INTERRELATIONSHIP MUST EXIST BETWEEN UNITS OF THE INFORMATION?
- HOW WILL THE INFORMATION BE RETRIEVED?

Figure A-26

IMAGE 2100 DATA BASE ORGANIZATION

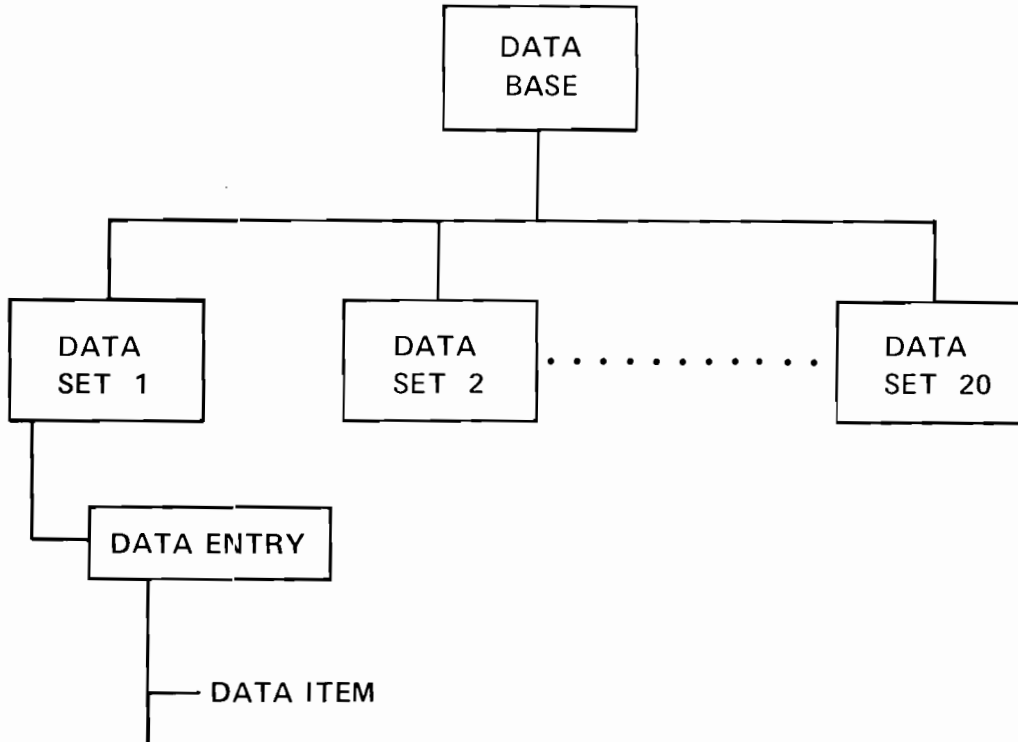


Figure B-3

IMAGE 2100 DATA SET TYPES

- DETAIL
- MANUAL MASTER
- MANUAL MASTER WITH LINKS
- AUTOMATIC MASTER

Figure B-4

DETAIL RECORD

A GROUPING OF CONTIGUOUS AND LOGICALLY RELATED FIELDS

EXAMPLE:

- CREDIT CARD RECORD
 - CARD #
 - NAME
 - MAXIMUM CREDIT
 - BALANCE DUE
 - CREDIT CLASS

ACCESS

- SEQUENTIAL
- DIRECT BY RELATIVE RECORD POSITION

Figure B-5

MANUAL MASTER

A GROUPING OF CONTIGUOUS AND LOGICALLY RELATED FIELDS WITH ONE FIELD SPECIFIED AS A KEY.

EXAMPLE:

- CREDIT CARD RECORD
 - CARD # <<KEY VALUE>>
 - NAME
 - MAXIMUM CREDIT
 - BALANCE DUE
 - CREDIT CLASS

ACCESS

- DIRECT BY KEY VALUE
- SEQUENTIAL
- DIRECT BY RELATIVE RECORD POSITION

Figure B-6

MANUAL MASTER WITH LINKS

A MANUAL MASTER RECORD WITH "LINKS" TO CONNECT IT TO LOGICALLY RELATED RECORDS IN OTHER DATA SETS.

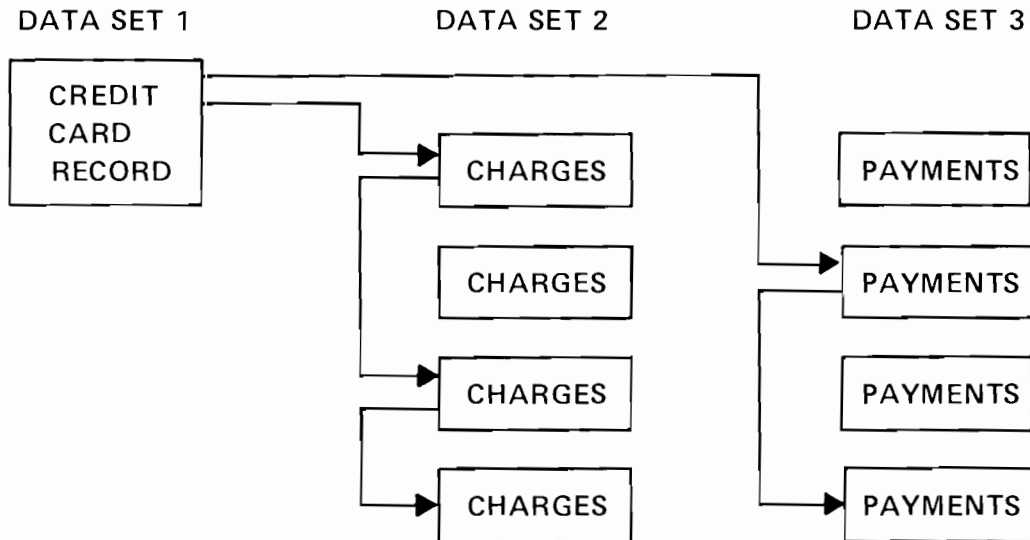
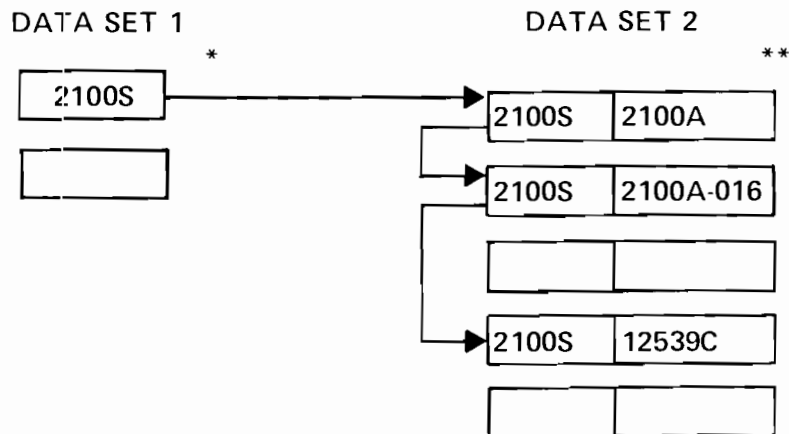


Figure B-7

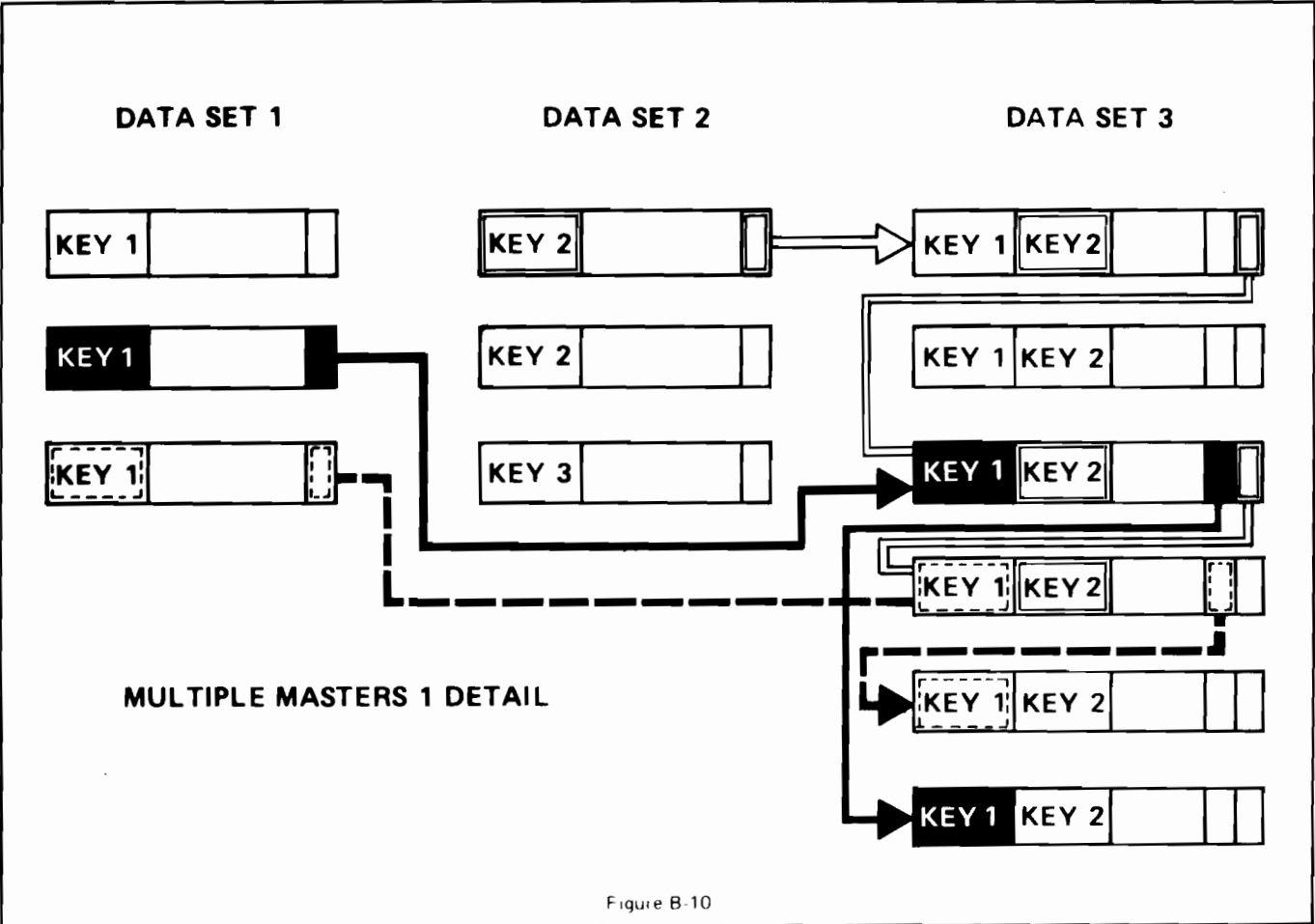
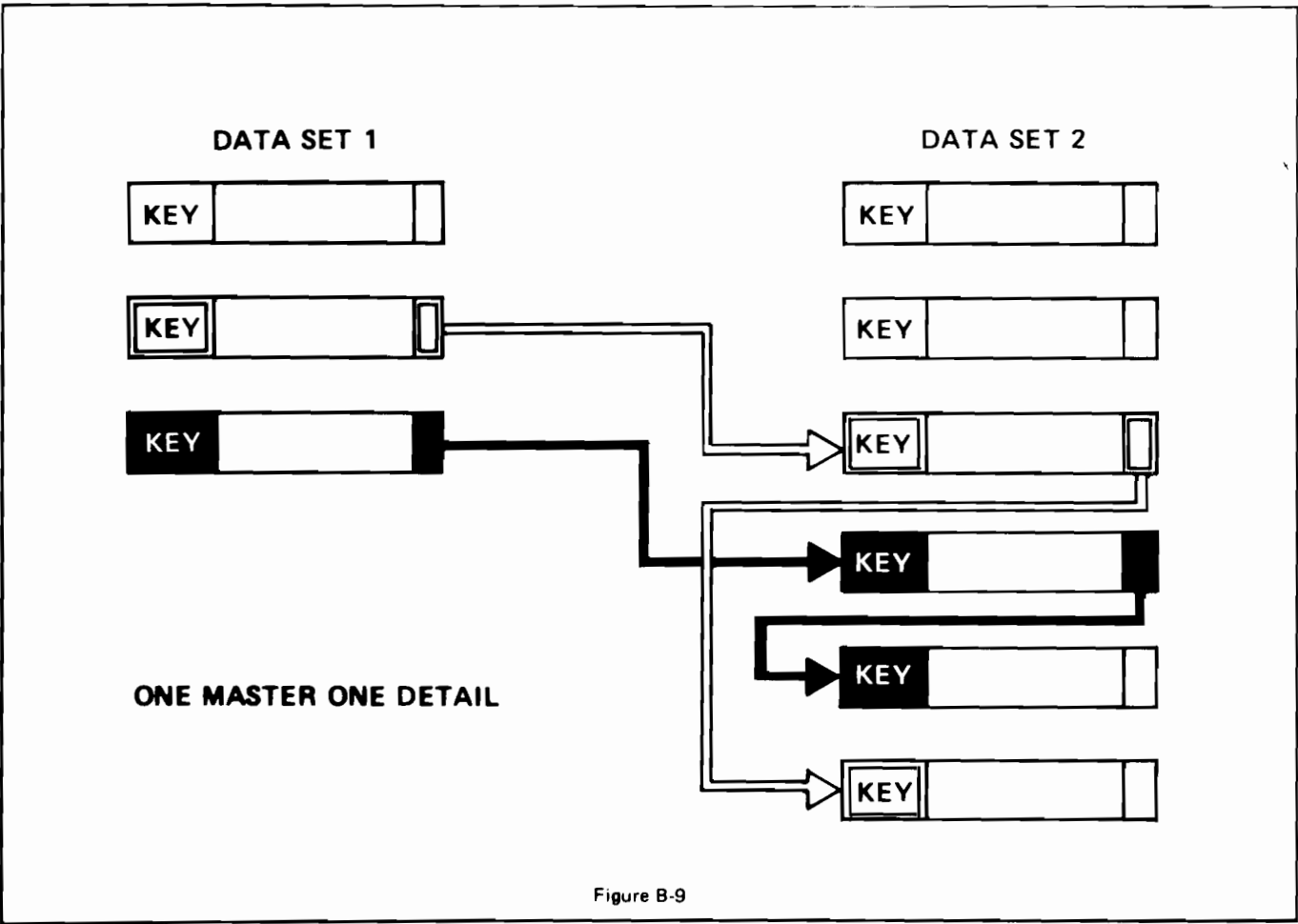
AUTOMATIC MASTER

THE STARTING POINT FOR A GROUP OF LOGICALLY (AND PHYSICALLY) CONNECTED DETAIL RECORDS.



*KEY ONLY
 **KEY AND DATA

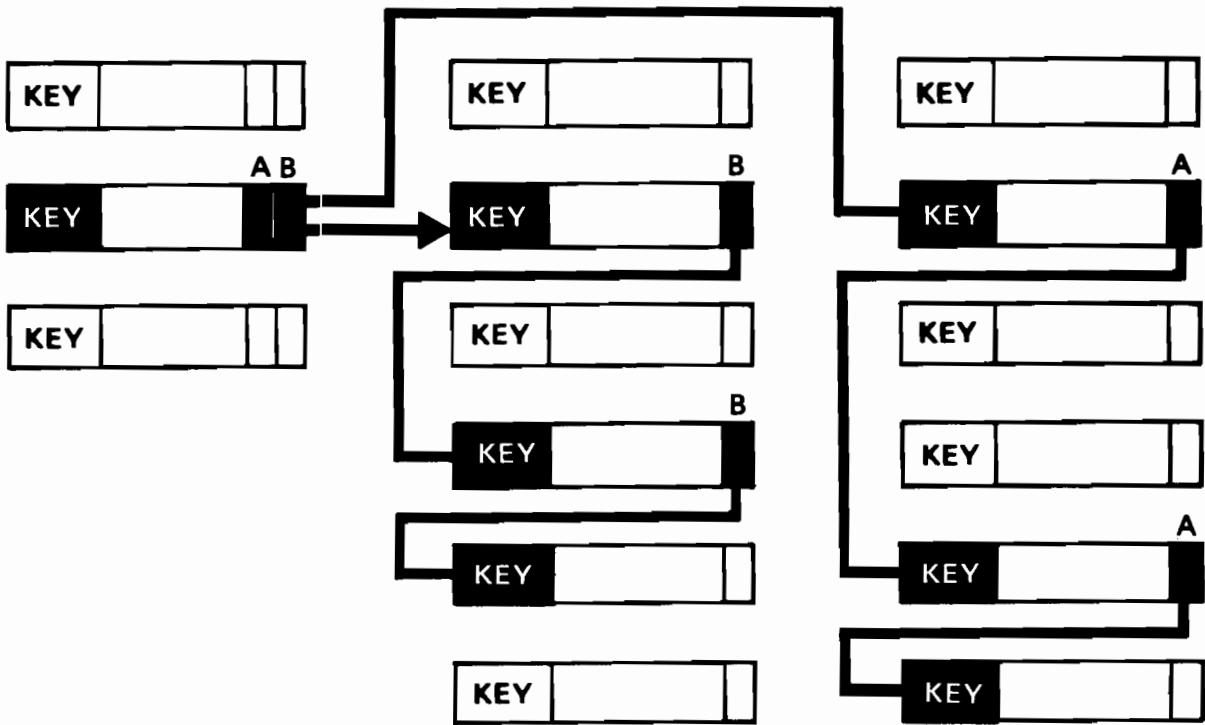
Figure B 8



DATA SET 1

DATA SET 2

DATA SET 3



ONE MASTER MANY DETAILS

Figure B-11

IMAGE 2100 SPECIFICATIONS



DATA BASE

- 2.5 MILLION BYTE CAPACITY ON 7901 OR 7900
- 5.0 MILLION BYTE CAPACITY ON 2883
- 20 DATA SETS
- 100 DATA ITEMS

DATA SET

- 2.5 MILLION BYTE CAPACITY ON 7901 OR 7900*
- 5.0 MILLION BYTE CAPACITY ON 2883*
- 100 DATA ITEMS*
- 32,000 RECORDS

*A DATA BASE COULD HAVE ONLY ONE DATA SET

Figure B-12

IMAGE 2100 SPECIFICATIONS

DATA ENTRY

DETAIL RECORDS

- DATA ITEMS = 100 - $\left\{ \begin{array}{l} \text{TOTAL ITEMS IN} \\ \text{OTHER DATA SETS} \end{array} \right\}$
- 5 DATA ITEMS MAY REFER BACK TO KEY VALUES IN MANUAL OR AUTOMATIC MASTER RECORDS
- DATA ITEMS MAY BE
 - BYTE ASCII
 - INTEGER 16 BIT
 - REAL 32 BIT

MANUAL MASTER

- DATA ITEMS = 100 - $\left\{ \begin{array}{l} \text{TOTAL ITEMS IN} \\ \text{OTHER DATA SETS} \end{array} \right\}$
- 1 DATA ITEM MUST BE A KEY VALUE
- DATA ITEMS MAY BE
 - BYTE ASCII
 - INTEGER 16 BIT
 - REAL 32 BIT

Figure B-13

IMAGE 2100 SPECIFICATIONS

MANUAL MASTER WITH LINKS

- DATA ITEMS = 100 - $\left\{ \begin{array}{l} \text{TOTAL ITEMS IN} \\ \text{OTHER DATA SETS} \end{array} \right\}$
- 1 DATA ITEM MUST BE A KEY VALUE
- DATA ITEMS MAY BE
 - BYTE ASCII
 - INTEGER 16 BIT
 - REAL 32 BIT
- THE MANUAL MASTER CAN HAVE LINKS TO A MAXIMUM OF 5 DETAIL DATA SETS.

AUTOMATIC MASTER

- DATA ITEMS = 1
- THE DATA ITEM IS THE KEY VALUE
- THE AUTOMATIC MASTER CAN HAVE LINKS TO A MAXIMUM OF 5 DETAIL DATA SETS.
- THE AUTOMATIC MASTER SERVES AS THE HEAD OF A CHAIN OF LINKED DETAIL DATA RECORDS.

Figure B-14

IMAGE 2100 SECURITY

DATA BASE

- INTEGER SECURITY CODE 1 TO 32,767

DATA ITEM

- PASSWORD WITH ACCESS LEVEL

EXAMPLE

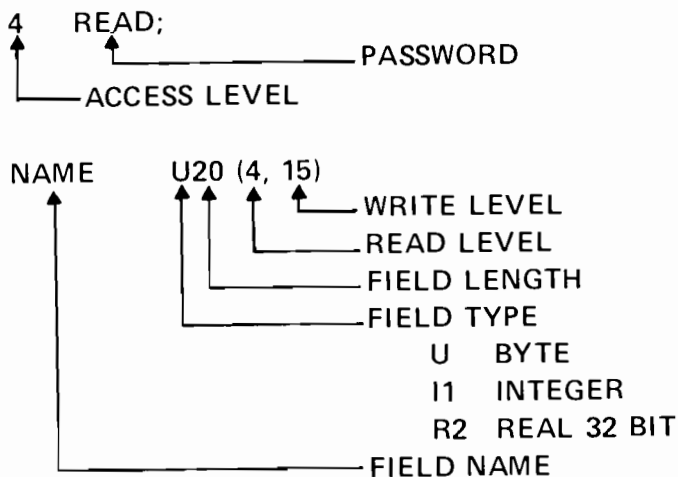


Figure B-15

QUERY 2100

REQUIREMENTS

- DATA BASE = NAME OF YOUR DATA BASE (ROOT FILE)
- LEVEL = YOUR PASSWORD
- SECURITY = YOUR EFMP SECURITY CODE
- MODE* = 1 READ ONLY
2 WRITE ONLY
3 READ/WRITE
5 OPEN A NEW DATA BASE AND THEN ALLOW BOTH READING AND WRITING.
- SELECT-FILE = THE NAME OF THE DOS III FILE USED TO STORE THE RECORD NUMBERS OF THE RECORDS SELECTED.
- SPEC-FILE = THE NAME OF A DOS III FILE USED TO STORE PROCEDURES.

*NOTE THE MODE MUST NOT EXCEED THE CAPABILITY ASSIGNED TO THE PASSWORD.

Figure B-16



IMAGE 2100
EXAMPLE 1

● **INDUSTRY**

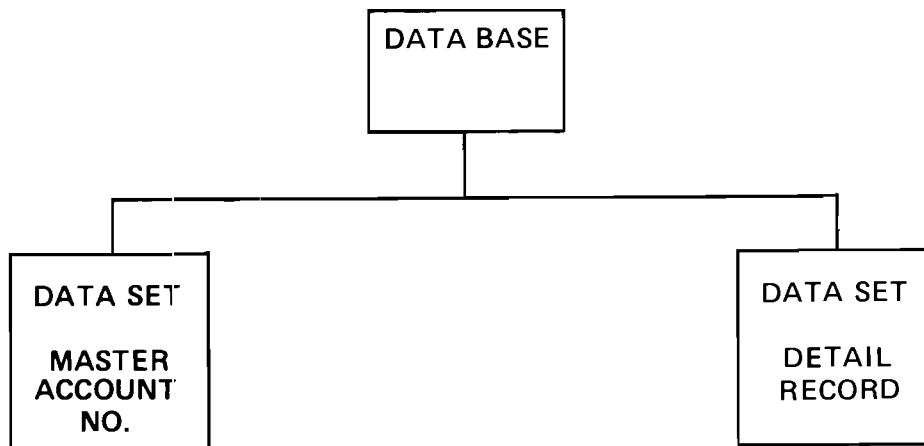
ANY COMPANY WHO RENTS OR LEASES EQUIPMENT TO OTHERS.

● **PROBLEM**

HOW DO YOU KEEP TRACK OF EQUIPMENT INSTALLED AT REMOTE LOCATIONS.

● **SOLUTION**

ACCESS YOUR DATA BASE BY ACCOUNT NUMBER, AND LINK ALL EQUIPMENT RECORDS WITH A COMMON ACCOUNT NUMBER TOGETHER.



:PROG, QUERY

QUERY 2100 READY

NEXT?

DATA-BASE = *EQT;

?LEVEL = INQ;

?SECURITY = 35;

?MODE = 1;

?SELECT-FILE = SEL;

NEXT?

FIND TELNO IS "415-257-4362" END;

0000 6 ENTRIES QUALIFIED

NEXT?

REPORT ALL;

NEXT?

EXIT;

@

IMAGE 2100
EXAMPLE 2

● **INDUSTRY**

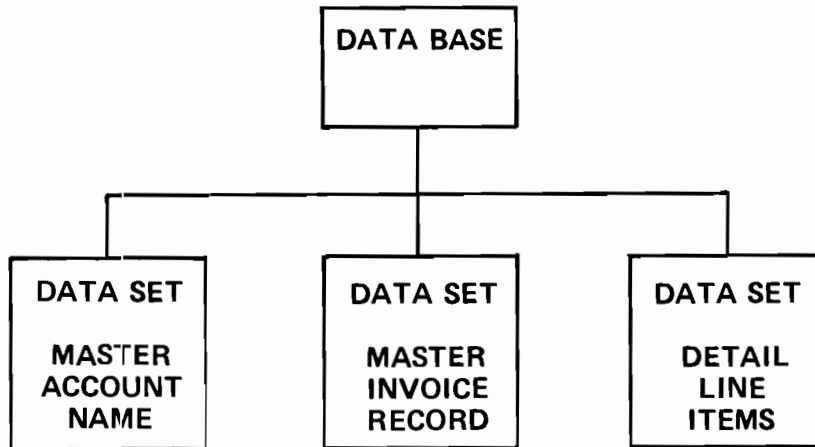
ANY COMPANY WHO INVOICES FOR FINISHED GOODS SOLD.

● **PROBLEM**

HOW TO MATCH CUSTOMER PAYMENTS (WHICH CAN REFLECT CREDITS ON INDIVIDUAL LINE ITEMS) TO STATEMENTS WHICH COVER MULTIPLE INVOICES WITH MULTIPLE LINE ITEMS.

● **SOLUTION**

SET UP A DATA SET OF INVOICE LINE ITEMS WITH EACH LINE ITEM LINKED TO OTHER LINE ITEMS WITH MATCHING INVOICE NUMBERS AND ACCOUNT NAMES.



HEWLETT-PACKARD IMAGE/2100 DATA BASE DEFINITION PROCESSOR

%CONTROL LIST, ERRORS = 5, ROOT, TABLE;
 EFMP PACK ID PN200; 35;
 BEGIN DATA BASE *A/R;
 LEVELS:

4 INQ;
 15 MAIN;

ITEMS:

KEY1, U20(4,15);
 KEY2, U10(4,15);
 ACC, U20(4,15);
 INV, U10(4,15);
 ITEM, U2(4,15);
 PART, U20(4,15);
 QTY, U6(4,15);
 PRICE, U8(4,15);

SETS:

NAME: ACCNT, A;
 ENTRY: KEY1(1);
 CAPACITY: 10;
 NAME: INVF, A;
 ENTRY: KEY2(1);
 CAPACITY: 30;
 NAME: LINE, DETAIL;
 ENTRY: ACC(ACCNT),
 INV(INVF),
 ITEM,
 PART,
 QTY,
 PRICE;
 CAPACITY: 100;

END.

DATA SET NAME	TYPE	FLD CNT	PATH CNT	ENTR LGTH	MED REC	CAPAC CT
ACCNT	A	1	1	10	6	10
INVF	A	1	1	5	6	30
LINE	D	6	2	33	5	100

NUMBER OF ERROR MESSAGES 0
 ITEM NAME COUNT: 8
 DATA SET COUNT: 3
 DATA BASE LENGTH: 36 SECTORS
 ROOT LENGTH: 2 SECTORS
 ROOT FILE *A/R CREATED.

:PROG, QUERY

QUERY 2100 READY

NEXT?

DATA-BASE = *AIR;

?LEVEL = INQ;

?SECURITY = 35;

?MODE = 1;

?SELECT-FILE = SEL;

NEXT?

FIND ACC IS "SOLE STEEL" END;

0000 4 ENTRIES QUALIFIED

NEXT?

REPORT ALL;

NEXT?

EXIT;

@

IMAGE 2100
EXAMPLE 3

● **INDUSTRY**

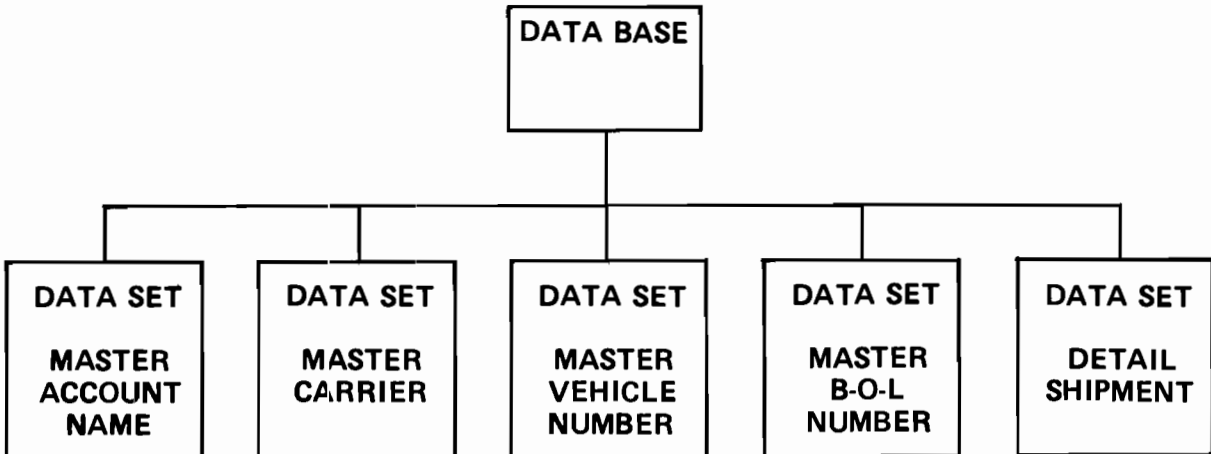
FREIGHT MOVEMENT

● **PROBLEM**

HOW DO YOU LOCATE A CUSTOMERS SHIPMENT WHEN YOU ARE NOT GIVEN A BILL OF LADING NUMBER.

● **SOLUTION**

ESTABLISH A DATA BASE OF SHIPMENT RECORDS WHICH ARE LOGICALLY LINKED TOGETHER BY ACCOUNT, CARRIER, AND VEHICLE NUMBER.



HEWLETT-PACKARD IMAGE/2100 DATA BASE DEFINITION PROCESSOR

\$CONTROL LIST, ERRORS = 5, ROOT, TABLE;
 EFMP PACK ID PN300; 35;
 BEGIN DATA BASE *FRT;
 LEVELS:

4 INQ;
 15 MAIN;

ITEMS:

KEY1, U20(4,15);
 KEY2, U8(4,15);
 KEY3, U4(4,15);
 KEY4, U20(4,15);
 ACC, U20(4,15);
 CARR, U8(4,15);
 ID, U4(4,15);
 BOL, U20(4,15);
 DESC, U20(4,15);

SETS:

NAME: K1,A;
 ENTRY: KEY1(1);
 CAPACITY: 20;
 NAME: K2,A;
 ENTRY: KEY2(1);
 CAPACITY: 10;
 NAME: K3,A;
 ENTRY: KEY3(1);
 CAPACITY: 40;
 NAME: K4,A;
 ENTRY: KEY4(1);
 CAPACITY: 50;
 NAME: SHIP,DETAIL;
 ENTRY: ACC(K1),
 CARR(K2),
 ID(K3),
 BOL(K4),
 DESC;
 CAPACITY: 100;

END.

DATA SET NAME	TYPE	FLD CNT	PATH CNT	ENTR LGTH	MED REC	CAPAC CT
K1	A	1	1	10	6	20
K2	A	1	1	4	6	10
K3	A	1	1	2	6	40
K4	A	1	1	10	6	50
SHIP	D	5	4	36	9	100

NUMBER OF ERROR MESSAGES 0
 ITEM NAME COUNT: 9
 DATA SET COUNT: 5
 DATA BASE LENGTH: 51 SECTORS
 ROOT LENGTH: 2 SECTORS
 ROOT FILE *FRT CREATED.

*FRT,35,MAIN;
 \$SET:SHIP

ACC IN COLUMNS 0001 THROUGH 0020 IS TYPE U
 CARR IN COLUMNS 0021 THROUGH 0028 IS TYPE U
 ID IN COLUMNS 0029 THROUGH 0032 IS TYPE U
 BOL IN COLUMNS 0033 THROUGH 0052 IS TYPE U
 DESC IN COLUMNS 0053 THROUGH 0072 IS TYPE U

12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012	HEWLETT PACKARD	UAL	3201024A-001-Q	2767A	LINE PRINTER
	HEWLETT PACKARD	TWA	4011024A-001-R	2610A	LINE PRINTER
	HEWLETT PACKARD	WESTERN	221024A-001-S	2614A	LINE PRINTER
	HEWLETT PACKARD	SEA-LAND	4001024A-002-A	2748B	TAPE READER
	HEWLETT PACKARD	LIONS	1651024A-003-B	7970B	MAG TAPE
	HEWLETT PACKARD	BEKINS	411024A-004-C	7900	DISC
	SOLE STEEL	UAL	4681025B-001-A	FAB PARTS	S.F.
	SOLE STEEL	TWA	6821025B-002-B	FAB PARTS	L.A.
	SOLE STEEL	UAL	3201025B-003-C	FAB PARTS	S.D.
	SOLE STEEL	UAL	9711025B-004-D	FAB PARTS	OAK
	WESTERN ELECTRIC	WESTERN	401025B-005-E	HEAD-SETS	SOUTHERN
	WESTERN ELECTRIC	WESTERN	221040A-001-A	33 ASR	LONG LINES
	WESTERN ELECTRIC	UAL	101050A-010-A	TAS SYSTEM	GEN TEL
	WESTERN ELECTRIC	BEKINS	151050B-020-B	800 SERIES	PBX PNB
	WESTERN ELECTRIC	LIONS	471050C-030-C	757 PBX	SYSTEM PNB
	GENERAL MOTORS	UAL	3801050C-040-D	50 AC	INDUCTION 50AX
	GENERAL MOTORS	UAL	1101060D-050-E	EXTENSION	BRACKETS
	GENERAL MOTORS	SEA-LAND	5501070E-060-F	104QW	WRAKE UNITS
	GENERAL MOTORS	WESTERN	501080E-070-G	104ZW10B	POWER SEATS
	GENERAL MOTORS	LIONS	601090F-010-A	AUTO TRANS	47A-204Z
	SAFEWAY	SEA-LAND	4001090F-020-B	CANNED PEACHES	50 CR
	SAFEWAY	SEA-LAND	5001090F-030-C	CANNED GRAPES	100 CR
	SAFEWAY	UAL	3201090F-040-D	FRESH APPRICOTS	
	SAFEWAY	UAL	851090F-050-E	FRESH ORANGES	
	SAFEWAY	TWA	1401090F-050-F	FRESH CABBAGE	
	SHEFCO	TWA	4011090F-050-G	T.V. MODEL	104Q-146
	SHEFCO	LIONS	2101090F-050-H	FREEZERS	AE-149268
	SHEFCO	UAL	2201090F-060-H	T.V. LAMPS	
	SEARS	SEA-LAND	5001090F-010-V	BICYCLES	810-Q-4271
	SEARS	SEA-LAND	5001090G-010-A	HOUSE PAINT	
	SEARS	LIONS	601090G-020-B	AUTO PARTS	
	SEARS	WESTERN	221090G-030-C	FRESH FLOWERS	
	ALS TV	SEA-LAND	5501090H-040-D	G.E. COLOR TV	
	ALS TV	TWA	1001090I-050-E	SONY TAPE DECKS	
	ALS TV	UAL	3201090J-060-F	RCA PORTABLE RADIOS	

\$END
 NUMBER OF ERRORS:0000
 DATA BASE SUCCESSFULLY BUILT OR UPDATED

:PROG, QUERY

QUERY 2100 READY

NEXT?

DATA-BASE = *FRT;

?LEVEL = INQ;

?SECURITY = 35;

?MODE = 1;

?SELECT-FILE = SEL;

NEXT;

FIND ACC IS "SHEFCO" END;

0000 3 ENTRIES QUALIFIED

NEXT?

REPORT ALL;

NEXT?

EXIT;

@

IMAGE 2100
EXAMPLE 4

● **INDUSTRY**

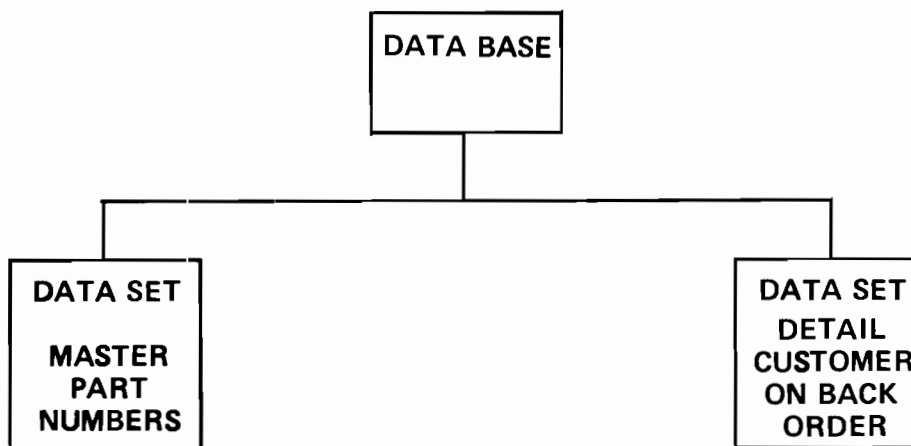
ANY COMPANY WHO SELLS FROM A FINISHED GOODS INVENTORY.

● **PROBLEM**

HOW DO YOU MAKE SURE YOUR BACK ORDER FILE IS UP TO DATE, AND THAT BACK ORDERS ARE FILLED IN THE CORRECT ORDER.

● **SOLUTION**

ESTABLISH A DATA SET OF BACK ORDER RECORDS WITH THE RECORDS LINKED BY PART NUMBER.



HEWLETT-PACKARD IMAGE/2100 DATA BASE DEFINITION PROCESSOR

\$CONTROL LIST, ERRORS = 5, ROOT, TABLE;

EFMP PACK ID PN400; 35;

BEGIN DATA BASE *INV;

LEVELS:

4 INQ;
15 MAIN;

ITEMS:

KEY, U20(4,15);
PN, U20(4,15);
CUST, U20(4,15);
QTY, U4(4,15);
CDE, U2(4,15);

SETS:

NAME: K5,A;
ENTRY: KEY(1);
CAPACITY: 20;
NAME: PARTS,DETAIL;
ENTRY: PN(K5),
CUST,
QTY,
CDE;
CAPACITY: 50;

END.

DATA SET NAME	TYPE	FLD CNT	PATH CNT	ENTR LGTH	MED REC	CAPAC CT
K5	A	1	1	10	6	20
PARTS	D	4	1	23	3	50

NUMBER OF ERROR MESSAGES 0

ITEM NAME COUNT: 5

DATA SET COUNT: 2

DATA BASE LENGTH: 15 SECTORS

ROOT LENGTH: 1 SECTORS

ROOT FILE *INV CREATED.

*INV,35,MAIN
\$SET:PARTS

PN IN COLUMNS 0001 THROUGH 0020 IS TYPE U
CUST IN COLUMNS 0021 THROUGH 0040 IS TYPE U
QTY IN COLUMNS 0041 THROUGH 0044 IS TYPE U
CDE IN COLUMNS 0045 THROUGH 0046 IS TYPE U

1234567890123456789012345678901234567890123456789012345678901234567890123456789012			
2100A	DE MARTINS ORCHARD	2	1
2100A	WESTERN FURNISHERS	1	1
2100A	PENINSULA SUPPLY	6	1
2121A	SOLE STEEL	1	1
2121A	CONSOLIDATED MFG.	3	1
2121A	CONGDON & CROME INC	1	1
2121A	CHANEYS	1	1
2120B	SAN JOSE SCHOOD DIST	2	1
2120B	RALPHS CARPENTERS	1	1
2120B	EDWARDS LUGGAGE	1	1
2100S	HUGHES	20	1
2100S	LOCKHEED	10	1
2100S	CONVAIR ASTRONAUTICS	5	1
2100S	SOLAR AIRCRAFT	4	1
2100S	RYAN AIRCRAFT	18	1
12960A	HUGHES	18	1
12960A	PENINSULA SUPPLY	6	1
12961A	XEROX	1	1
12961A	THE DATA COMPANY	1	1
12961A	PLEMONS CONSTRUCTION	1	1
12961A	DURACLEAN	1	1



\$END
NUMBER OF ERRORS:0000
DATA BASE SUCCESSFULLY BUILT OR UPDATED

:PROG, QUERY

QUERY 2100 READY

NEXT?

DATA-BASE = *INV;

?LEVEL = INQ;

?SECURITY = 35;

?MODE = 1;

?SELECT-FILE = SEL;

NEXT;

FIND PN IS "2100A" END;

0000 3 ENTRIES QUALIFIED

NEXT?

REPORT ALL;

NEXT?

EXIT;

@

IMAGE 2100
EXAMPLE 5

● **INDUSTRY**

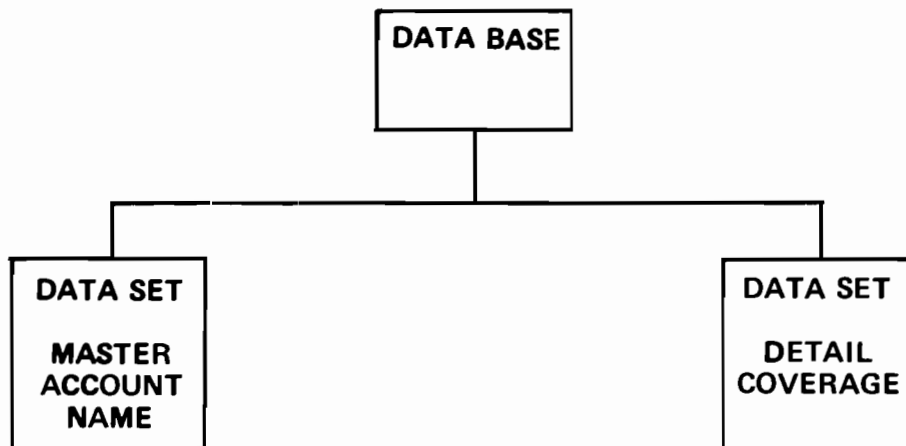
SMALL INSURANCE COMPANYS

● **PROBLEM**

**HOW DO YOU PROVIDE YOUR CUSTOMERS WITH THE
ADVANTAGES ON AN ON-LINE RECORD SYSTEM AT A
PRICE YOU CAN AFFORD.**

● **SOLUTION**

**USE IMAGE 2100 TO KEEP A RECORD OF EACH
CUSTOMERS INDIVIDUAL COVERAGE, AND LINK
ALL COVERAGE RECORDS BY CUSTOMER NUMBER.**



HEWLETT-PACKARD IMAGE/2100 DATA BASE DEFINITION PROCESSOR

\$CONTROL LIST, ERRORS = 5, ROOT, TABLE;
 EFMP PACK ID PN500; 35;
 BEGIN DATA BASE *INS;
 LEVELS:

4 IN0;
 15 MAIN;

ITEMS:

KEY, U20(4,15);
 ACC, U20(4,15);
 COV, U20(4,15);
 LIM, U8(4,15);

SETS:

NAME: K6,A;
 ENTRY: KEY(1);
 CAPACITY: 10;
 NAME: COVE,DETAIL;
 ENTRY: ACC(K6),
 COV,
 LIM;
 CAPACITY: 50;

END.

DATA SET NAME	TYPE	FLD CNT	PATH CNT	ENTR LGTH	MED REC	CAPAC CT
K6	A	1	1	10	6	10
COVE	D	3	1	24	3	50

NUMBER OF ERROR MESSAGES 0
 ITEM NAME COUNT: 4
 DATA SET COUNT: 2
 DATA BASE LENGTH: 14 SECTORS
 ROOT LENGTH: 1 SECTORS
 ROOT FILE *INS CREATED.

*INS,35,MAIN
\$SET:COVE

ACC IN COLUMNS 0001 THROUGH 0020 IS TYPE U
COV IN COLUMNS 0021 THROUGH 0040 IS TYPE U
LIM IN COLUMNS 0041 THROUGH 0048 IS TYPE U

12345678901234567890123456789012345678901234567890123456789012		
THORBURN	LIFE	500000
THORBURN	COMP	10000000
THORBURN	MEDICAL	75000
WILLARD	LIFE	250000
WILLARD	COMP	25000000
WILLARD	MEDICAL	450000
WILLARD	TOWING	6700
WILLARD	BOAT	30000
BAILEY	LIFE	4500000
BAILEY	FIRE	27500
BAILEY	THIEF	357500
BAILEY	LIB	2750000
BAILEY	FLOOD	1250000
JONES PLUMMING	LIFE	75000000
JONES PLUMMING	BUSINESS	75000000
JONES PLUMMING	LIB	375000
JONES PLUMMING	COMP	2750000
WEED	LIFE	1250000
WEED	CAR	350000
WEED	BOAT	450000

\$END

NUMBER OF ERRORS:0000

DATA BASE SUCCESSFULLY BUILT OR UPDATED

:PROG, QUERY

QUERY 2100 READY

NEXT?

DATA-BASE = *INS;

?LEVEL = INQ;

?SECURITY = 35;

?MODE = 1;

?SELECT-FILE = SEL;

NEXT?

FIND ACC IS "WILLARD" END;

0000 5 ENTRIES QUALIFIED

NEXT?

REPORT ALL;

NEXT?

EXIT;

@

IMAGE 2100
EXAMPLE 6

● **INDUSTRY**

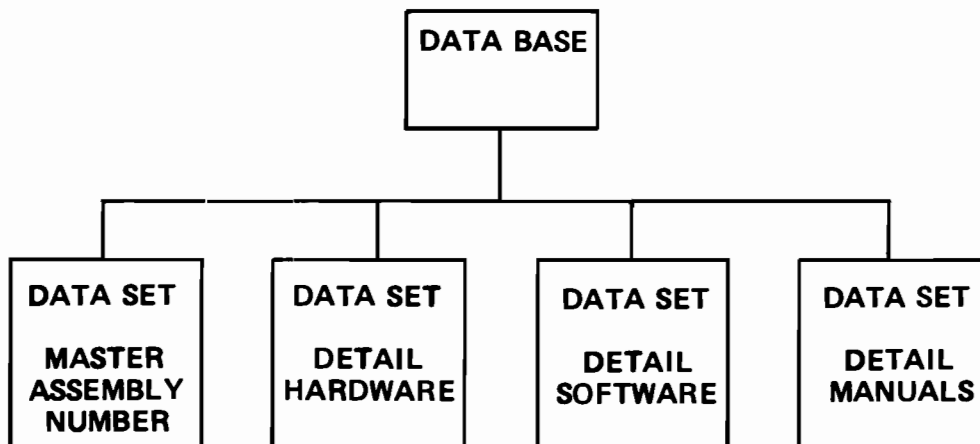
MANUFACTURING COMPANYS

● **PROBLEM**

**HOW TO MAINTAIN UP TO DATE LISTS OF THE ITEMS
THAT GO INTO A FINAL ASSEMBLY.**

● **SOLUTION**

**IMAGE 2100 CAN BE USED TO PROVIDE MATERIAL
PARTS EXPLOSIONS BY GROUPING RECORDS IN A
DETAIL DATA SET BY ASSEMBLY NUMBER.**



HEWLETT-PACKARD IMAGE/2100 DATA BASE DEFINITION PROCESSOR

\$CONTROL LIST, ERRORS = 5, ROOT, TABLE;
 EFMP PACK ID PN600; 35;
 BEGIN DATA BASE *BOM;
 LEVELS:

4 INQ;
 15 MAIN;

ITEMS:

KEY1, U10(4,15);
 HLIST, U10(4,15);
 SLIST, U10(4,15);
 MLIST, U10(4,15);
 DES1, U40(4,15);
 DES2, U40(4,15);
 DES3, U40(4,15);

SETS:

NAME: K1,A;
 ENTRY: KEY1(3);
 CAPACITY: 10;
 NAME: HARD,DETAIL;
 ENTRY: HLIST(K1),
 DES1;
 CAPACITY: 100;
 NAME: SOFT,DETAIL;
 ENTRY: SLIST(K1),
 DES2;
 CAPACITY: 100;
 NAME: MANL,DETAIL;
 ENTRY: MLIST(K1),
 DES3;
 CAPACITY: 100;

END.

DATA SET NAME	TYPE	FLD CNT	PATH CNT	ENTR LGTH	MED REC	CAPAC CT
K1	A	1	3	5	12	10
HARD	D	2	1	25	3	100
SOFT	D	2	1	25	3	100
MANL	D	2	1	25	3	100

NUMBER OF ERROR MESSAGES 0
 ITEM NAME COUNT: 7
 DATA SET COUNT: 4
 DATA BASE LENGTH: 69 SECTORS
 ROOT LENGTH: 2 SECTORS
 ROOT FILE *BOM CREATED.

*BOM,35,MAIN
\$SET:HARD

HLIST IN COLUMNS 0001 THROUGH 0010 IS TYPE U
DES1 IN COLUMNS 0011 THROUGH 0050 IS TYPE U

1234567890123456789012345678901234567890123456789012345678901234567890123456789012
2120A 2100A COMPUTER
2120A 12885A-001 8K MEMORY
2120A 12895A DMA
2120A 2752A 33ASR
2120A 12531C INTERFACE
2120A 2748A PAPER TAPE READER
2120A 12597A-002 INTERFACE
2120A 12960A K7! DISC SUBSYSTEM
2120A 12575H WINDER
2120A 2940-016,052,156 CABINET
2120B 2100S COMPUTER
2120B 2100S-208
2120B 2100S-239
2120B 7901 DISC SUBSYSTEM
2120B 13210A
2120B 13211A
2120B 13211A-001
2120B 2752A SYSTEM CONSOLE
2120B 29402B CABINET
2120B 29402B-001 POWER
2120B 29402B-010 SLIDES
2120B 12692B
2120B 12692B-003
2120B 12531C INTERFACE
2121A 2100S COMPUTER
2121A 7901 DISC SUBSYSTEM
2121A 13210A
2121A 13211A-001
2121A 2752A SYSTEM CONSOLE
2121A 29402B
2121A 29402B-001 CABINET
2121A 29402B-010 POWER
2121A 12692B
2121A 12692B-003
2121A 12531C INTERFACE
2121A 12869C

\$SET:SOFT

SLIST IN COLUMNS 0001 THROUGH 0010 IS TYPE U
DES2 IN COLUMNS 0011 THROUGH 0050 IS TYPE U

1234567890123456789012345678901234567890123456789012345678901234567890123456789012
2121A 24307A-60001 DSGEN
2121A 24307A-60002 DISCM
2121A 24307A-60003 EXEC MODUS
2121A 24307A-60004 JOB PROCESSOR
2121A 24307A-60005 BOOTSTRAP
2121A 24156F-60001 DISC DRIVER
2121A 24308A-60001 RELOCATING LOADER
2121A 24309A-60001 EFMP
2121A 24157C-60001 DVR05
2121A 20985F-60001 DVR00
2120A 24225F-60001 DSGEN

2120A	24225F-60002	DISCM
2120A	24225F-60003	EXEC MODU
2120A	24225F-60004	JOB PROCESSOR
2120A	24225F-60005	BOOTSTRAP
2120A	24156E-60001	DISC DRIVER
2120A	24155D-60001	RELOCATING LOADER
2120A	24227-60001	EFMP
2120A	24314A-60001	IN CORE BOOTSTRAP
2120A	24157B-60001	DVR05
2120A	24985D-60001	DVR00
2120B	24225F-60001	DSGEN
2120B	24225F-60002	DISCM
2120B	24225F-60003	EXEC MODU
2120B	24225F-60004	JOB PROCESSOR
2120B	24225F-60005	BOOTSTRAP
2120B	24156E-60001	DISC DRIVER
2120B	24155D-60001	RELOCATING LOADER
2120B	24227-60001	EFMP
2120B	24314A-60001	IN CORE BOOTSTRAP
2120B	24157B-60001	DVR05
2120B	24985D-60001	DVR00

\$SET:MANL

MLIST IN COLUMNS 0001 THROUGH 0010 IS TYPE U
 DES3 IN COLUMNS 0011 THROUGH 0050 IS TYPE U

1234567890123456789012345678901234567890123456789012345678901234567890123456789012		
2121A	02100-90130	DOS III MANUAL
2120A	5951-1375	GENERATION
2120B	5951-1375	GENERATION
2120A	02116-91779	DOS-M MANUAL
2120B	02116-91779	DOS-M MANUAL

\$END

NUMBER OF ERRORS:0000
 DATA BASE SUCCESSFULLY BUILT OR UPDATED

:PROG, QUERY

QUERY 2100 READY

NEXT?

DATA-BASE = *BOM;

?LEVEL = INQ;

?SECURITY = 35;

?MODE = 1;

?SELECT-FILE = SEL;

NEXT?

FIND HLIST IS "2120A" END;

00010 ENTRIES QUALIFIED

NEXT?

REPORT ALL;

NEXT?

FIND SLIST IS "2120A" END;

00011 ENTRIES QUALIFIED;

NEXT?

REPORT ALL;

NEXT?

EXIT;

@



IMAGE 2100
EXAMPLE 7

● **INDUSTRY**

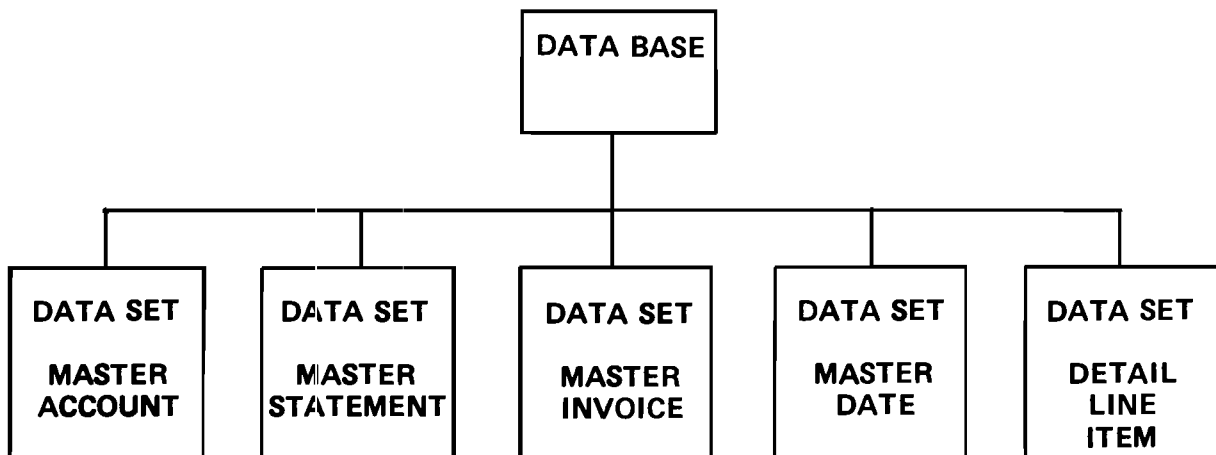
GENERAL

● **PROBLEM**

HOW DO YOU KEEP TRACK OF YOUR PAYABLES.

● **SOLUTION**

**USE IMAGE 2100 TO KEEP TRACK OF ALL INVOICE
LINE ITEMS WITH EACH RECORD LINKED TO ALL
OTHERS WITH THE SAME ACCOUNT, STATEMENT,
INVOICE, OR DATE.**



HEWLETT-PACKARD IMAGE/2100 DATA BASE DEFINITION PROCESSOR

\$CONTROL LIST, ERRORS = 5, ROOT, TABLE;
 EFMP PACK ID PN700; 35;
 BEGIN DATA BASE *A/P;
 LEVELS:

4 INQ;
 15 MAIN;

ITEMS:

KEY1, U10(4,15);
 KEY2, U10(4,15);
 KEY3, U10(4,15);
 KEY4, U10(4,15);
 ACC, U10(4,15);
 STAT, U10(4,15);
 INV, U10(4,15);
 DATE, U10(4,15);
 VALUE, U10(4,15);

SETS:

NAME: K1,A;
 ENTRY: KEY1(1);
 CAPACITY: 20;
 NAME: K2,A;
 ENTRY: KEY2(1);
 CAPACITY: 20;
 NAME: K3,A;
 ENTRY: KEY3(1);
 CAPACITY: 20;
 NAME: K4,A;
 ENTRY: KEY4(1);
 CAPACITY: 20;
 NAME: DATA,DETAIL;
 ENTRY: ACC(K1),
 STAT(K2),
 INV(K3),
 DATE(K4),
 VALUE;
 CAPACITY: 100;

END.

DATA SET NAME	TYPE	FLD CNT	PATH CNT	ENTR LGTH	MED REC	CAPAC CT
K1	A	1	1	5	6	20
K2	A	1	1	5	6	20
K3	A	1	1	5	6	20
K4	A	1	1	5	6	20
DATA	D	5	4	25	9	100

NUMBER OF ERROR MESSAGES 0
 ITEM NAME COUNT: 9
 DATA SET COUNT: 5
 DATA BASE LENGTH: 36 SECTORS
 ROOT LENGTH: 2 SECTORS
 ROOT FILE *A/P CREATED.

*A/P,35,MAIN
\$SET:DATA

ACC IN COLUMNS 0001 THROUGH 0010 IS TYPE U
STAT IN COLUMNS 0011 THROUGH 0020 IS TYPE U
INV IN COLUMNS 0021 THROUGH 0030 IS TYPE U
DATE IN COLUMNS 0031 THROUGH 0040 IS TYPE U
VALUE IN COLUMNS 0041 THROUGH 0050 IS TYPE U

1234567890123456789012345678901234567890123456789012345678901234567890123456789012					
ROBERTS	1024A-001	10000A	6-15-73	500000	
ROBERTS	1024A-001	10000B	6-15-73	43000	
ROBERTS	1024B-002	10010A	7-25-73	50000	
ROBERTS	1024B-002	10020C	7-30-73	250000	
S.J.GAS	3762Q	2762	5-15-73	475000	
S.J.GAS	3762Q	3245	6-15-73	10000	
FAIRWOOD	A-8762	2400Q-001	6-5-73	34500	
FAIRWOOD	A-8762	2410R-002	7-25-73	1250000	
FAIRWOOD	3462A-001	2500X-010	8-14-73	50000	
SOLE	0800R	1462-J	6-15-73	345000	
SOLE	0805R	1465-K	6-25-73	125000	
SOLE	0900A	2437-A	7-15-73	2451000	

\$END

NUMBER OF ERRORS:0000
DATA BASE SUCCESSFULLY BUILT OR UPDATED

:PROG, QUERY

QUERY 2100 READY

NEXT?

DATA-BASE = *AIP;

?LEVEL = INQ;

?SECURITY = 35;

?MODE = 1;

?SELECT-FILE = SEL;

NEXT?

FIND ACC IS "ROBERTS" AND

? STAT IS "1024A-001" END;

0000 2 ENTRIES QUALIFIED

NEXT?

REPORT ALL;

NEXT?

EXIT;

@

IMAGE 2100
EXAMPLE 8

● **INDUSTRY**

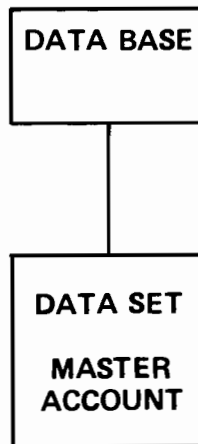
SMALL BANKS, OR CREDIT BUREAUS. MEDIUM SIZE STORES WITH THEIR OWN CREDIT OFFICES.

● **PROBLEM**

HOW TO KEEP AN UP TO DATE RECORD OF THE CUSTOMERS CHARGE HISTORY, AND TO EASILY AND RAPIDLY ACCESS THAT DATA.

● **SOLUTION**

USE IMAGE 2100 TO STORE CUSTOMER RECORDS RANDOMLY BY CREDIT CARD NUMBER.



HEWLETT-PACKARD IMAGE/2100 DATA BASE DEFINITION PROCESSOR

\$CONTROL LIST, ERRORS = 5, ROOT, TABLE;
 EFMP PACK ID PN800; 35;
 BEGIN DATA BASE *BANK;
 LEVELS:

4 ING;
 15 MAIN;

ITEMS:

CARD, U12(4,15);
 NAME, U30(4,15);
 MAX, U6(4,15);
 BAL, U6(4,15);
 ID, U2(4,15);

SETS:

NAME: CHARG, MANUAL;
 ENTRY: CARD(0),
 NAME,
 MAX,
 BAL,
 ID;
 CAPACITY: 50;

END.

DATA SET NAME	TYPE	FLD CNT	PATH CNT	ENTR LGTH	MED REC	CAPAC CT
CHARG	M	5	0	28	3	50

NUMBER OF ERROR MESSAGES 0
 ITEM NAME COUNT: 5
 DATA SET COUNT: 1
 DATA BASE LENGTH: 14 SECTORS
 ROOT LENGTH: 1 SECTORS
 ROOT FILE *BANK CREATED.

*BANK,35,MAIN
\$SET:CHARG

CARD IN COLUMNS 0001 THROUGH 0012 IS TYPE U
NAME IN COLUMNS 0013 THROUGH 0042 IS TYPE U
MAX IN COLUMNS 0043 THROUGH 0048 IS TYPE U
BAL IN COLUMNS 0049 THROUGH 0054 IS TYPE U
ID IN COLUMNS 0055 THROUGH 0056 IS TYPE U

123456789012345678901234567890123456789012345678901234567890123456789012
96-823224 ARTHUR GARY THORBURN 50000 8500 1
96-824624 MICHAEL JEAN GABET 30000 12500 1
96-742168 KARL HELNESS 30000 25000 1
96-823221 JOHN ELWARD 50000 5000 1
96-283224 REUBEN GALLEGUILLOS 50000 4500 1
96-422328 PAM SAYLOR 50000 22500 1
96-177692 ART PRICE 50000 34500 1
96-542132 BRUCE NOEL 50000 27500 1
96-376281 WALT MURISON 50000 10400 1
96-428742 CHUCK ULREY 50000 3800 1

END
NUMBER OF ERRORS:0000
DATA BASE SUCCESSFULLY BUILT OR UPDATED

:PROG, QUERY

QUERY 2100 READY

NEXT?

DATA BASE = *BANK;

?LEVEL = INQ;

?SECURITY = 35;

?MODE = 1;

?SELECT-FILE = SEL;

NEXT?

FIND CARD IS "96-823224" END;

0000 1 ENTRIES QUALIFIED

NEXT?

REPORT ALL;

NEXT?

EXIT;

@

A SAMPLE REPORT ON THE *BANK DATA BASE

REPORT

H1, "CREDIT CARD REPORT", 49, SPACE B5;

H2, "CARD", 6;

H2, "NAME", 22;

H2, "MAX", 55;

H2, "NOW", 65, SPACE A5;

S1, CARD;

D, CARD, 14;

D, NAME, 48;

D, MAX, 58, E1;

D, BAL, 68, E1;

TF, BAL, 68, ADD, E1, SPACE B5;

E1, "\$,\$\$\$99";

END;

CREDIT CARD REPORT

CARD	NAME	MAX	NOW
96-177692	ART PRICE	\$500,00	\$345,00
96-283224	REUBEN GALLEGUILLOS	\$500,00	\$45,00
96-376281	WALT MURISON	\$500,00	\$104,00
96-422326	PAM SAYLOR	\$500,00	\$225,00
96-428742	CHUCK ULREY	\$500,00	\$38,00
96-542132	BRUCE NOEL	\$500,00	\$275,00
96-742168	KARL HELNFESS	\$300,00	\$250,00
96-823221	JOHN ELWARD	\$500,00	\$50,00
96-823224	ARTHUR GARY THORBURN	\$500,00	\$85,00
96-824624	MICHAEL JEAN GABET	\$300,00	\$125,00

1,542,00

PROCEDURES

THE ABILITY TO STORE "FIND" AND "REPORT"
STATEMENTS IN A FILE FOR LATER USE.

EXAMPLE:

```
?CREATE NAME = #TEL;  
?FIND TELNO IS " " END;
```

```
:PROG, QUERY
```

```
.  
. .  
. .  
. .  
. .
```

```
FIND NAME = #TEL;
```





```

0001 FTN,L
0002 PROGRAM U100
0003 C
0004 C DESCRIPTION OF FIELDS
0005 C ISORT 1024 EFMP TEMP BUFFER
0006 C ISLGT 1 IMAGE COUNTER
0007 C IRUN 1175 ROOT FILE RUN TABLE
0008 C IOPN 128 EFMP OPEN FILE TABLE
0009 C
0010 C IBASE 3 DATA BASE (ROOT FILE) NAME
0011 C ILEVEL 3 IMAGE LEVEL WORD (PASSWORD)
0012 C ISCOD 1 EFMP SECURITY CODE (INTEGER)
0013 C IMODE 1 IMAGE MODE CODE
0014 C ISTAT 4 RETURNED STATUS
0015 C IDSET 3 DATA SET NAME (ALSO EFMP FILE NAME)
0016 C IBUF 31 USER DATA RECORD
0017 C IARG 6 SEARCH ITEM VALUE (KEY)
0018 C JANS 2 ANSWER TO QUESTION
0019 C JTEST 1 TEST FOR ANSWER "YE"S
0020 C
0021 COMMON ISORT(1024),ISLGT(1),IRUN(1175),IOPN(128)
0022 C
0023 COMMON IBASE(3),ILEVL(3),ISCOD(1),IMODE(1),ISTAT(4)
0024 C
0025 COMMON IDSET(3),IBUF(31),IARG(6),JANS(2)
0026 C
0027 C
0028 DATA JTEST/2HYE/
0029 C
0030 C OPEN DATA BASE
0031 C
0032 100 WRITE(1,110)
0033 READ(1,120)IBASE
0034 WRITE(1,130)
0035 READ(1,120)ILEVL
0036 WRITE(1,140)
0037 READ(1,*)ISCOD(1)
0038 WRITE(1,150)
0039 READ(1,*)IMODE(1)
0040 C
0041 C
0042 110 FORMAT("DATA BASE ?")
0043 120 FORMAT(3A2)
0044 130 FORMAT("LEVEL ?")
0045 140 FORMAT("SECURITY CODE ?")
0046 150 FORMAT("MODE ?")
0047 C
0048 C
0049 CALL DBOPN(IBASE,ILEVL,ISCOD,IMODE,ISTAT)
0050 C
0051 C CHECK FOR ERROR
0052 C
0053 IF (ISTAT(1) .EQ. 0) GO TO 200

```



```
0054 C
0055 C      ERROR
0056 C
0057 C      WRITE(1,160)ISTAT(1)
0058 160      FORMAT("ERROR ", I5, " ON OPEN")
0059 C      PAUSE
0060 C
0061 C      NO ERROR
0062 C
0063 C      READ ROUTINE
0064 C          1.  GET DATA SET NAME
0065 C          2.  GET KEY
0066 C          3.  READ RECORD ( MANUAL MASTER ASSUMED )
0067 C          4.  WRITE RECORD TO CRT
0068 C          5.  ASK IF MORE IS WANTED
0069 C          6.  GET ANSWER
0070 C          7.  DO 1,2,3,4,5,6, OR CLOSE DATA BASE
0071 C
0072 200      WRITE(1,210)
0073 C          READ(1,220)IDSET
0074 C          WRITE(1,230)
0075 C          READ(1,240)IARG
0076 C
0077 C
0078 210      FORMAT("DATA SET ?")
0079 220      FORMAT(3A2)
0080 230      FORMAT("ENTER KEY VALUE")
0081 240      FORMAT(6A2)
0082 C
0083 C      SET IMODE TO INDICATE KEYED READ
0084 C
0085 C      IMODE = 4
0086 C
0087 C      IMAGE READ
0088 C
0089 C      CALL DBGET(IDSET,IMODE,ISTAT,IBUF,IARG)
0090 C
0091 C      CHECK FOR ERROR
0092 C
0093 C      IF (ISTAT(1) .EQ. 0) GO TO 260
0094 C
0095 C      ERROR
0096 C
0097 C      WRITE(1,250)ISTAT(1)
0098 250      FORMAT("ERROR ", I5, " ON READ")
0099 C      CALL DBCLS(0,0)
0100 C      PAUSE
0101 C
0102 C      NO ERROR
0103 C
0104 260      CALL EXEC(2,1B,IBUF(4),28)
0105 C
0106 C      ASK FOR MORE
```

```
0107 C
0108 WRITE(1,280)
0109 280 FORMAT("MORE ?")
0110 READ(1,290)JANS
0111 290 FORMAT(2A2)
0112 IF (JANS(1) .EQ. JTEST) GO TO 200
0113 C
0114 C DONE
0115 C
0116 CALL DBCLS(0,0)
0117 END
```

** NO ERRORS*

PAGE 0004 U100

(FTN4--RELEASE 24177B--JULY, 1971)

ENDS

FELOCATING LOADER

NAME/ENTRY ADDR

M100	23666
*U100	23666
FMTIO	24400
*.RIO.	24644
*.IIO.	24632
*.XIO.	24661
*.XAY.	25015
*.RAY.	25046
*.IAY.	25057
*.DIO.	25165
*.BIO.	25260
*.IOI.	24765
*.IOR.	24724
*.IAR.	25112
*.RAR.	25070
*.DTA.	25344
*NEWIO	25276
*OLDIO	25303
PAUSE	25612
*.PAUS	25612
*.STOP	25650
CLRIO	25756
*CLRIO	25756
DBOPN	25761
*DBOPN	25766
DBGET	26555
*DBGET	26607
DBCLS	27277
*DBCLS	27302
.FLUN	27640
*.FLUN	27640
FLIB	27661
*.FAD	27661
*.FSB	27670
*.FMP	27677
*.FDV	27706
*.FIX	27715
*.FLOAT	27724
.XFER	27764
*.XFER	27776
*.DFER	27764
DBLE	30030
*DBLE	30032
SNGL	30065
*SNGL	30066
FRMTR	30133
*.FRMN	30441
*.LS2F	30433
*.INPN	30463
*.DTAN	30455
.OPSY	32673
*.OPSY	32673
.EAU.	32733
*.MPY	32733
*.DIV	32740

*.DLD	32745
*.DST	32752
.ENTR	33003
*.ENTR	33012
*.ENTP	33003
PHICM	33073
*PHICM	33073
*CMPCT	33115
RUNTB	33116
*RUNTB	33116
PHIL	33176
*PHIL	33203
HASH	33547
*HASH	33551
PHIS1	33574
*PHIS1	33601
DUMRX	33711
*\$LIBR	33711
*\$LIBX	33736
.ZRLB	33771
*.ZRLB	33771
.XPAK	34032
*.XPAK	34032
.PACK	34223
*.PACK	34223
.XCOM	34337
*.XCOM	34337
PHIMV	34410
*PHIMV	34410
*MOVCT	34427

```

0001 FTN.L
0002 PROGRAM MTI
0003 C
0004 C THIS PROGRAM DEMONSTRATES MULTI-TERMINAL INPUT TO AN
0005 C IMAGE 2100 DATA BASE. THIS PROGRAM USES TCS FOR THE
0006 C SCHEDULING OF INPUT REQUESTS, AND THE EXPANDED MODE
0007 C OF IMAGE TO ACCESS THE DATA BASE.
0008 C
0009 C THE PROGRAM ACCEPTS A KEY VALUE FROM A TERMINAL, AND
0010 C PASSES THE KEY TO IMAGE. THE IMAGE DATA BASE ACCESSED
0011 C IS A MANUAL MASTER RECORD WITH ONE FIELD SPECIFIED
0012 C AS A KEY, AN WITH NO LINKS TO ANY OTHER DATA SETS.
0013 C THE DATA BASE WAS SPECIFIED USING DBDS, AND BUILD BY DBUS.
0014 C
0015 C THIS PROGRAM EXPECTS A USER TERMINAL TO TERMINATE
0016 C THE JOB BY TYPING IN AN **. THIS IS NECESSARY IN
0017 C ORDER FOR IMAGE TO PROPERLY CLOSE THE DATA SET.
0018 C
0019 C ***** WARNING DO NOT TERMINATE WITH A :OFF *****
0020 C
0021 C
0022 C THE RECORD STRUCTURE IS ASSUMED TO BE:
0023 C
0024 C 1. 03 WORDS FOR THE MEDIA RECORD
0025 C 2. CARD 12 ASCII CHARACTERS (KEY)
0026 C 3. MAX 06 ASCII CHARACTERS
0027 C 4. BAL 06 ASCII CHARACTERS
0028 C 5. ID 02 ASCII CHARACTERS
0029 C
0030 C
0031 C DESCRIPTION OF FIELDS AND VARIABLES
0032 C
0033 C ISORT 1024 EFMP TEMP BUFFER
0034 C ISLGT 1 IMAGE COUNTER
0035 C IRUN 1175 ROOT FILE RUN TABLE
0036 C IOPN 128 EFMP OPEN FILE TABLE
0037 C
0038 C IBASE 3 DATA BASE (ROOT FILE) NAME
0039 C ILEV 3 IMAGE LEVEL WORD (PASSWORD)
0040 C ISCOD 1 EFMP SECURITY CODE (INTEGER)
0041 C IMODE 1 IMAGE MODE CODE
0042 C ISTAT 4 IMAGE RETURNED STATUS
0043 C IDSET 3 DATA SET NAME (ALSO EFMP FILE NAME)
0044 C
0045 C IBUF 160 USER BUFFER AREA
0046 C 40 WORDS * 4 USERS MAX
0047 C IARG 6 SEARCH ITEM VALUE (KEY)
0048 C
0049 C
0050 C JPQ 180 TCS PENDING Q ARRAY
0051 C JSTAT 1 TCS STATUS
0052 C JPTR 1 POINTER INTO USER I/O BUFFER AREA
0053 C JRET 1 RETURN FROM READ WITHOUT WAIT

```

```

0054 C          JRET2          1          RETURN FOOOM WRITE WITHOUT WAIT
0055 C          JLU            4          ARRAY FOR LOGICAL UNIT NUMBERS
0056 C          LU             1          LOGICAL UNIT NUMBER RETURNED BY TCS
0057 C          JTLOG         1          TRANSMISSION LOG RETURNED BY TCS
0058 C          JPAR          1          TERMINAL NUMBER CODE.
0059 C
0060 C          JFN            3          DOS ERROR FILE NAME
0061 C          JEN            1          END TEST
0062 C
0063 C
0064 C          COMMON  ISORT(1024),ISLGT(1),IRUN(1175),IOPN(128)
0065 C
0066 C          COMMON  IBASE(3),ILEVL(3),ISCOD(1),IMODE(1),ISTAT(4)
0067 C
0068 C          COMMON  IDSET(3),IBUF(160),IARG(6)
0069 C
0070 C          COMMON  JPQ(180),JSTAT(1),JLU(4)
0071 C
0072 C          DIMENSION  JFN(3)
0073 C
0074 C          DATA JFN(1),JFN(2),JFN(3)/2HIM,2HER,2HR /
0075 C          DATA JEN/2H**/
0076 C
0077 C          START SYSTEM BY GETTING BASIC INFO FROM SYSTEM CONSOLE
0078 C
0079 C          WRITE(1,100)
0080 100        FORMAT("DATA BASE")
0081          READ(1,110)IBASE
0082 110        FORMAT(3A2)
0083          WRITE(1,120)
0084 120        FORMAT("LEVEL")
0085          READ(1,130)ILEVL
0086 130        FORMAT(3A2)
0087          WRITE(1,140)
0088 140        FORMAT("SECURITY CODE")
0089          READ(1,*)ISCOD(1)
0090          WRITE(1,150)
0091 150        FORMAT("MODE")
0092          READ(1,*)IMODE(1)
0093          WRITE(1,160)
0094 160        FORMAT("DATA SET")
0095          READ(1,170)IDSET
0096 170        FORMAT(3A2)
0097 C
0098 C          OPEN THE IMAGE 2100 DATA BASE
0099 C
0100          CALL DBOPN(IBASE,ILEVL,ISCOD,IMODE,ISTAT)
0101 C
0102 C          CHECK FOR ERROR
0103 C
0104          IF (ISTAT(1) .EQ. 0) GO TO 200
0105 C
0106 C          ERROR ON OPEN

```

```
0107 C
0108 WRITE(1,180)ISTAT(1)
0109 180 FORMAT("ERROR ON IMAGE OPEN",/,"ERROR NUMBER ",I4, " ")
0110 GO TO 5000
0111 C
0112 C NO ERROR SET UP FOR MULTI-TERMINAL INPUT
0113 C
0114 200 WRITE(1,210)IDSET
0115 210 FORMAT("DATA BASE DATA SET ",3A2, " OPENED")
0116 C
0117 C ASSIGN RETURN POINTS FOR TCS
0118 C
0119 ASSIGN 500 TO JRET1
0120 ASSIGN 1000 TO JRET2
0121 C
0122 C FIND OUT HOW MANY TERMINALS (MAX IS 4)
0123 C
0124 WRITE(1,220)
0125 220 FORMAT("HOW MANY TERMINALS")
0126 READ(1,*)J2
0127 C
0128 C SET UP TCS PENDING Q ARRAY
0129 C
0130 CALL ATP(82,JPQ,20,KSTAT,0,0,0)
0131 C
0132 C GET LOGICAL UNIT NUMBERS
0133 C
0134 DO 240 J1 = 1,J2
0135 WRITE(1,230)J1
0136 230 FORMAT("ENTER LOGICAL UNIT FOR INPUT NO. ", I4)
0137 READ(1,*)JLU(J1)
0138 240 CONTINUE
0139 C
0140 C AT THIS TIME WE ARE READY TO START.
0141 C
0142 WRITE(1,250)
0143 250 FORMAT("STARTING MULTI-TERMINAL OPERATION")
0144 C
0145 C START MULTIPLE READ WITHOUT WAIT ON TERMINALS
0146 C
0147 300 JPTR = 1
0148 DO 310 J1 = 1,J2
0149 CALL ATP(1,20400B+JLU(J1),IBUF(JPTR),40,J1,JRET1)
0150 JPTR = JPTR + 40
0151 310 CONTINUE
0152 C
0153 C SUSPEND PROGRAM
0154 C
0155 CALL ATP(53)
0156 C
0157 C
0158 C
0159 C
```




```
0160 C
0161 C
0162 C THIS SECTION OF CODE IS EXECUTED WHEN TCS RETURNS FROM A
0163 C READ WITHOUT WAIT. CALL STATUS TO SEE WHO IS FINISHED
0164 C
0165 C
0166 C
0167 C
0168 500 CALL ATP(79,JSTAT,JPAR,LU,JTLOG)
0169 C
0170 C ESTABLISH USER BUFFER
0171 C
0172 C JPTR = (JPAR * 40) - 39
0173 C
0174 C TEST TO SEE IF INPUT IS THE TERMINATION CODE
0175 C
0176 C IF (IBUF(JPTR) .NE. JEN) GO TO 506
0177 C
0178 C TERMINATION CHARACTER
0179 C
0180 C WRITE(1,505)LU
0181 505 FORMAT("TERMINAL ", I3, " HAS TERMINATED OPERATION")
0182 CALL DBCLS(0,0)
0183 GO TO 5000
0184 C
0185 C MOVE INPUT (ASSUMED TO BE A KEY) INTO IMAGE IARG (12 CHAR)
0186 C
0187 506 IARG(1) = IBUF(JPTR)
0188 DO 510 J1 = 1,5
0189 IARG(J1 + 1) = IBUF(JPTR + J1)
0190 510 CONTINUE
0191 C
0192 C SET MODE TO INDICATE A KEYED READ
0193 C
0194 C IMODE = 4
0195 C
0196 C IMAGE READ OF A MANUAL MASTER RECORD WITH NO LINKS
0197 C
0198 C CALL DBGET(IDSET,IMODE,ISTAT,IBUF(JPTR),IARG)
0199 C
0200 C CHECK FOR ERROR CODES
0201 C
0202 C IF (ISTAT(1) .EQ. 0) GO TO 520
0203 C
0204 C IMAGE RETURNED AN ERROR CODE
0205 C DO A DOS 3 FILE SECTOR READ OF A ERROR MESSAGE FILE
0206 C AND PUT THE ERROR MESSAGE INTO THE USER I/O BUFFER
0207 C
0208 C CALL EXEC(14,102B,IBUF(JPTR+3),28,JFN,ISTAT(1))
0209 C
0210 C NO ERROR. START A WRITE WITHOUT WAIT
0211 C
0212 520 CALL ATP(2,20000B+LU,IBUF(JPTR+3),28,IPAR,JRET2)
```

```
0213 C
0214 C     SUSPEND
0215 C
0216 C     CALL ATP(53)
0217 C
0218 C
0219 C
0220 C
0221 C
0222 C
0223 C     THIS SECTION OF CODE IS EXECUTED WHEN A WRITE WITHOUT
0224 C     WAIT IS COMPLETE
0225 C
0226 C     ISSUE A STATUS CALL TO SEE WHO IS DONE
0227 C
0228 C 1000 CALL ATP(79,JSTAT,JPAR,LU,JTLOG)
0229 C
0230 C     SET UP USER BUFFER
0231 C
0232 C     JPTR = (JPAR * 40) - 39
0233 C
0234 C     START READ WITHOUT WAIT
0235 C
0236 C     CALL ATP(1,20400B+LU,IBUF(JPTR),40,JPAR,JRET1)
0237 C
0238 C     SUSPEND
0239 C
0240 C     CALL ATP(53)
0241 C
0242 C     THAT IS ALL THERE IS TO IT
0243 C
0244 C 5000 WRITE(1,5010)
0245 C 5010 FORMAT("PROGRAM END")
0246 C     END
```

** NO ERRORS*

PAGE 0006 MTI

(FTN4--RELEASE 24177B--JULY, 1971)

0247

ENDS

RELOCATING LOADER

NAME/ENTRY ADDR

UTI	24356
*MTI	24356
.EAU.	25534
*.MPY	25534
*.DIV	25541
*.DLD	25546
*.DST	25553
FMTIO	25604
*.RIO.	26050
*.IIO.	26036
*.XIO.	26065
*.XAY.	26221
*.RAY.	26252
*.IAY.	26263
*.DIO.	26371
*.BIO.	26464
*.IOI.	26171
*.IOR.	26130
*.IAR.	26316
*.RAR.	26274
*.DTA.	26550
*NEWIO	26502
*OLDIO	26507
CLRIO	27016
*CLRIO	27016
DBOPN	27021
*DBOPN	27026
ATP	27615
*ATP	30140
DBCLS	32157
*DBCLS	32162
DBGET	32520
*DBGET	32552
DUMRX	33242
*\$LIBR	33242
*\$LIBX	33267
.FLUN	33322
*.FLUN	33322
FLIB	33343
*.FAD	33343
*.FSB	33352
*.FMP	33361
*.FDV	33370
*IFIX	33377
*FLOAT	33406
.XFER	33446
*.XFER	33460
*.DFER	33446
DBLE	33512
*DBLE	33514
SNGL	33547
*SNGL	33550
FRMTR	33615
*.FRMN	34123
*.LS2F	34115

*.INPN	34145
*.DTAN	34137
.OPSY	36355
*.OPSY	36355
.ENTR	36415
*.ENTR	36424
*.ENTP	36415
PHICM	36505
*PHICM	36505
*CMPCT	36527
RUNTB	36530
*RUNTB	36530
PHIS1	36610
*PHIS1	36615
PHIL	36725
*PHIL	36732
HASH	37276
*HASH	37300
.ZRLB	37323
*.ZRLB	37323
.XPAK	37364
*.XPAK	37364
.PACK	37555
*.PACK	37555
.XCOM	37671
*.XCOM	37671
PHIMV	37742
*PHIMV	37742
*MOVCT	37761

COMPETITION

DEFINITION OF TERMS

SEQUENTIAL	=	READ RECORDS ONE AT A TIME (SAME AS MAG TAPE)
DIRECT	=	READ A PARTICULAR RECORD BY ITS POSITION WITHIN THE FILE
KEY OR ISAM	=	READ A PARTICULAR RECORD BASED ON AN ALPHANUMERIC FIELD CON- TAINED WITHIN THE RECORD
LINK	=	LOGICAL ASSOCIATE RECORDS BY ONE OR MORE KEY FIELDS
FIELD PROTECTION	=	BE ABLE TO ASSIGN A READ AND WRITE ACCESS TO A FIELD WITHIN A RECORD
SECURITY	=	THE ABILITY TO ASSIGN A PASSWORD WHICH HAS UNIQUE READ AND WRITE CAPABILITIES
UTILITIES	=	PROGRAMS WHICH ALLOW THE NON- PROGRAMMER TO DEFINE, BUILD, AND BACK-UP DATA-BASES
QUERY	=	A PROGRAM WHICH ALLOWS THE NON- PROGRAMMER TO ACCESS THE DATA BASE

	IMAGE	IBM 3-6/3-10	DEC	D.G.	G.A.	INTERDATA	LOCKHEED
SEQUENTIAL	✓	✓	✓	✓	✓	✓	✓
DIRECT	✓	✓	✓	✓	✓	✓	✓
KEY (ISAM)	✓	RPG	RPG				RPG
LINK	✓						
FIELD PROTECTION	✓						
SECURITY	✓						
UTILITIES	✓	✓					
QUERY	✓						

**● MAKE SURE
CUSTOMER KNOWS COMPLETE SYSTEM COST**

HARDWARE \$ _____

OPERATING SYST. _____

DATA BASE PKG _____

TOTAL SYSTEM COST \$ _____



HOW TO ORDER

MANUALS

- DECIMAL STRING ARITHMETIC ROUTINES 02100-90140
- DESIGNING IMAGE 2100 DATA BASES 02100-90138
- QUERY ON-LINE ACCESS TO IMAGE 2100 DATA BASES 02100-90142
- USING IMAGE 2100 DATA BASES 02100-90139

SOFTWARE

- 24376A IMAGE 2100 SOFTWARE

