Bureau of Mineral Resources, Geology & Geophysics

R E C O R D

RECORD 1989/7

COPY SERVICE DATABASE

USER MANUAL

Tony Boston

Geoscience Computing and Database Branch

1989/7
Copy 4

RECORD 1989/7

COPY SERVICE DATABASE

USER MANUAL

Tony Boston

Geoscience Computing and Database Branch

CONTENTS

1 Introduction

The Copy Service database has been set up using the ORACLE DBMS on the DG MV 20000. Its purpose is to store information on and associated costs of material available for reproduction by the copy service. Data is stored on BMR bulletins, reports and records, petroleum and geophysical reports and logs as well as geological and geophysical maps and sections. The logical record structure of the database is given in appendix 1 and consists of 11 tables. The schema used to create the database is given in appendix 2.

This is a brief guide to the use of the data entry form for the Copy Service Database. This form allows the user to query, update, insert and delete records from the database. Appendix 3 provides complete documentation of the data entry form created using SQL*FORMS including all imbedded triggers.

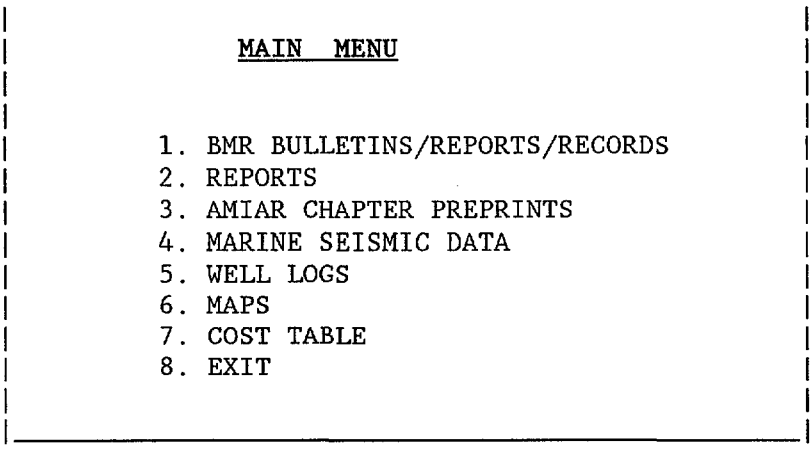
When a keyboard symbol appears in the following instructions, it means that you press that key, eg [NL] means press the carriage-return key. [Shift-F9] means that you must press the 'shift' key before pressing the F9 key.

To run the data entry form enter :

COPYS [NL]

After an introductory message, a screen appears which prompts for the ORACLE username and password. Enter your username, Copy_service, hit [NL], then enter the password followed by [NL] to log into ORACLE.

A main menu then appears as follows :



Choose the number corresponding to your desired option, then hit [NL]. For example, if you enter '2', the block for data entry into the AMIAR table appears. The [C1] and [C2] keys can be used to travel backwards and forwards between blocks. To view a list of all the screens in the form use [shift-F2]. A complete list of all keys used in the forms package and their function is given at the end of this guide.

An example of each screen in the form and detailed information on their use is given in appendices 4 and 5 respectively.

2 Data Entry

2.1 Querying the Database

Move to the block which corresponds with the table you wish to query.

To initiate a query of the database, press function key [F2], 'enter query'. Move to the field(s) to be queried using the [NL] and [HOME] keys, and enter query data. For example, if you wish to query the well logs table for logs from the well Grunter 1, press [F2] and enter this name in the well name field. Alternatively, the '%' symbol may be used for general searches. For example, **Grunter%** entered in the well name field would search for all records with well names starting with Grunter.

To execute the query, press function key [F1]. After a small delay, during which the word 'Working' appears at the bottom of the screen, the retrieved data appears on the screen. If many records are retrieved by a query, they may be searched through by using the arrow keys.

2.2 Updating the Database

To update data retrieved to a form as the result of a query, modify the data as it appears on the screen. If many records have been retrieved, you may step through them with the arrow keys and make any necessary changes.

The changes do not affect the database itself until you commit by pressing the function key [F3] or leave the current screen. In the latter case, you are asked if you want to commit the changes you have made. Answer 'y' for yes only if you are sure the changes you have made are correct. If the record is sucessfully committed, the message 'Transaction complete--1 record processed' appears at the bottom of the screen.

2.3 Inserting Records into the Database

To enter new data into the database, the screen must be blank. The screen is ready for data input when it is first viewed, or when the create record function key [F4] is pressed. In general, it is best to press the create record key [F4] before a new record is entered. As for updated records, new records must be committed to the database using function key [F3] or, when prompted, on exiting the screen.

When entering multiple records which differ from each other in only a minor way, after the first record has been entered and committed, press create record [F4] to clear the screen, then the duplicate record key, [shift-F4] to copy the last record onto the screen. Then you may alter the fields as necessary and commit the next record to the database.

2.4 Deleting Records from the Database

To delete a record from the database, first summon the record onto the screen by querying the database. Next press the delete record key, [shift-F9] which will clear the record from the screen. Then press [F3] to commit the change to the database.

2.5 Exiting the Form

To exit the form at any time, press the cancel/exit key [F11]. If prompted with the message 'Do you want to commit the changes you have made', then generally type 'n' for no unless you are sure that you want to make changes to the database. It is better to have to re-enter data than to accidentally corrupt the database.

Notes on keyboard:

[NL]	: New line key. Use to move forward between fields in a block.
[HOME]	: Home key. Use to move backwards between fields in a block.
[C1]	: Use to travel backwards between blocks.
[C2]	: Use to travel forward between blocks which usually correspond to ORACLE tables.
[C4]	: Use to scroll through a screen-full of records.
[SHIFT-F2]	: Displays block menu.
[F3]	: Commits changes to the database.
[F4]	: Clears block ready for entry of next record.
[SHIFT-F4]	: Displays last record on the screen.
[F11]	: Use to exit from Form or cancel current operation.
ARROW KEYS	: Use to scroll through records after a query, and move about fields in a block.

APPENDIX 1 : LOGICAL RECORD STRUCTURE OF COPY SERVICE DATABASE

BMR BULLETINS/ REPORTS/RECORDS

Pub Number
Volume Part Num
Number of Pages
Number of Plates
Length of Plates
Price

REPORTS

BMR File Number
Govt Printer Num
Title
Number of Pages
Number of Plates
Length of Plates

REPORT AUTHORS

BMR File Number
Author

AMIAR PREPRINTS

Title
Year
Number of pages
Num Copied

SEISMIC DATA

Survey Number
Line Number
Product Code
Price
Length of Section

SURVEY

Survey Number
Survey Name

SHOT POINT DATA

Product Code
Part
Part Start
Part Stop

LOG DETAILS

Well Name
Log Type
BMR File Number
Govt Print Num
In Copy Service
Length of Log

GEOLOGICAL MAPS

Sheet Name
Scale
Coverage
Price

GEOPHYSICAL MAPS

Sheet Name
Scale
Grid Reference Num
BMR Reference Num
Release Date
Data Type
Contoured

COSTS

Cost Type
Cost

```
rem
rem
rem
rem
rem
rem     APPENDIX 2 : SCHEMA FOR THE COPY SERVICE DATABASE.
rem
rem             Date    : May, 1988
rem
rem             Author : Tony Boston
rem
rem
rem
rem

rem
rem     Create Space Definitions.
rem

create space CSDB_1
        Datapages (Initial 200,
                   Increment 50,
                   Maxextents 9999,
                   Pctfree 10)
        Indexpages(Initial 50,
                   Increment 25,
                   Maxextents 9999)
        Partition A;

create space CSDB_2
        Datapages (Initial 50,
                   Increment 25,
                   Maxextents 9999,
                   Pctfree 10)
        Indexpages(Initial 20,
                   Increment 10,
                   Maxextents 9999)
        Partition A;

create space CSDB_3
        Datapages (Initial 10,
                   Increment 5,
                   Maxextents 9999,
                   Pctfree 10)
        Indexpages(Initial 10,
                   Increment 5,
                   Maxextents 9999)
        Partition A;
```

```
rem
rem     Create the views csgeodx1 and csgeodx2 based on information
rem     in GEODX database relevant to the copy service. That is, BMR
rem     Bulletins, Reports and Records which have Publication IDs of
rem     183, 185 and 191 respectively.
rem

create view csgeodx1(gdxnum,title,pubnum,volpartnum,status,pubdate)
as select cd_article,tx_title,id_publication,
          no_volpart,nvl(cd_status,' '),dt_publication
    from   geodx.source,geodx.article,geodx.bmrpub
    where  (id_publication = 183 or id_publication = 185 or
          id_publication = 191)
    and    geodx.article.id_source = geodx.source.id_source
    and    geodx.source.id_source  = geodx.bmrpub.id_source
    and    geodx.article.id_source = geodx.bmrpub.id_source

create view csgeodx2(gdxnum,author,order_sequence)
as select cd_article,name_author,order_sequence
    from   geodx.article_author,geodx.author
    where  geodx.article_author.id_author = geodx.author.id_author;

rem
rem     Create two views of data in the PEDIN database for use by
rem     the Copy Service. These were created by the PEDIN DBA manager.
rem     Two synonyms were then created for these views.
rem

create view copy_service_well_logs
        (uno,well_name,bmr_file_num,location,state)
as select wells.uno,name,file_no,on_off,state
from wells,wdata1
where wells.uno = wdata1.uno and wdata1.code = 'WL';

create view copy_service_logs(uno,log_type,top,bottom)
as select uno,testtype,top,bottom
from wdata1
where code='WL';

create synonym well for bruce.copy_service_well_logs;

create synonym log for copy_service_logs;
```

```
rem
rem     Create a table to hold some of the Publication details of
rem     BMR Bulletins, Reports and Records in the Copy Service.
rem

create table pub_details
       (pubnum number(5) not null,
        Volpartnum char(20) not null,
        Num_pages number(3),
        Num_plates number(3),
        Length_of_plates number(3),
        Price number(6,2))
space CSDB_2;

create unique index Pubnum_in
on Pub_Details(Pubnum,Volpartnum);

rem
rem     Create the Report Table to hold data on Well Completion
rem     Reports, Seismic Survey Reports, Marine Survey Reports and
rem     Geophysical Survey Reports.
rem

create table Report
        (BMR_File_Num char(8) not null,
         Govt_Print_Num number(5) not null,
         Title char(240) not null,
         Num_Pages number(3),
         Num_Plates number(3),
         Length_of_Plates number(3))
space    CSDB_1;

create unique index File_num_in_1
on Report(BMR_File_Num);

create unique index Govt_Print_Num_in_1
on Report(Govt_Print_Num);

create unique index Title_in
on Report(Title);
```

```
rem
rem     Create a table for all authors of reports.
rem

create table Author
        (BMR_File_Num char(8) not null,
         Author char(30) not null)
space   CSDB_2;

create index File_Num_in_2
on Author(BMR_File_Num);

create index Author_in
on Author(Author);

rem
rem     Create the AMIRP Table to hold data on AMIR pre-print chapters.
rem

create table Amirp
        (Title char(50) not null,
         Year number(4) not null,
         Num_Pages number(3))
space   CSDB_2;

create unique index Title_Year
on AMIRP(Title,Year);

rem
rem     Create a Survey Table to hold the Seismic Survey Names
rem     and Numbers.
rem

create table SurveyData
        (Survey_Num number(3) not null,
         Survey_Name char(50) not null)
space   CSDB_2;

create unique index Survey_in_2
on SurveyData(Survey_Num);

create unique index Survey_name_in
on SurveyData(Survey_Name);
```

```
rem
rem     Create the Seismic Table to hold data on Seismic Survey
rem     Sections.
rem

create table SeismicData
        (Survey_Num number(3) not null,
         Line_num char(15) not null,
         Product_code char(11) not null,
         Price number(6,2),
         Length_of_Section number(3))
space   CSDB_2;

create index Survey_in_1
on SeismicData(Survey_Num);

create unique index Prod_in_1
on SeismicData(Product_Code);

rem
rem     Create the Shot Point Data table to contain data on the number
rem     of parts in the line and the shot points of those parts.
rem

create table ShotPointData
        (Product_Code char(11) not null,
         Part char(5),
         Part_Start char(10),
         Part_Stop  char(10))
space   CSDB_2;

create index Prod_in_2
on ShotPointData(Product_Code);

rem
rem     Create the Log Table to hold data on Schlumberger Logs.
rem

create table Log_details
        (Well_name char(30) not null,
         Log_Type char(20) not null,
         BMR_file_num char(8),
         Govt_Print_num number(5),
         In_cs char(1),
         Length_of_Log number(3))
space   CSDB_2;
```

```
create index File_num_in_3
on Log_details(BMR_File_Num);

create index Govt_Print_num_in_2
on Log_details(Govt_Print_Num);

create unique index Well_name_Log_type_in
on Log_details(Well_Name,Log_Type);

rem
rem     Create the Map Tables to hold data on B.M.R. Maps.
rem

create table GeoMap
        (Sheet_name char(22) not null,
         Scale char(7),
         Coverage char(1),
         Price number(6,2))
space CSDB_1;

create index Sheet_name_in_1
on GeoMap(Sheet_name);

create table AeroMap
        (Sheet_name char(22) not null,
         Scale char(7),
         Grid_Ref_Num char(6),
         BMR_Ref_Num char(15),
         Release_date date,
         Data_type char(20),
         Contoured char(1))
space CSDB_1;

create index sheet_name_in_2
on AeroMap(Sheet_name);

create unique index Grid_Ref_num_in
on AeroMap(Grid_ref_num);

rem
rem     Create the Cost Table to hold data on Copy Charges.
rem

create table Cost
        (Cost_Type char(20) not null,
         Cost number(6,2))
space   CSDB_3;
```

APPENDIX 3 : **DATA ENTRY FORM - DESIGNER DOCUMENTATION**

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Application-level Triggers

DO_NOTHING STEP (1)

```
#EXEMACRO NULL;
Actions:
If step not successful: stop with error (No Message)
```

Q_NEXTBLOCK STEP (1)

```
#EXEMACRO NXTBLK; CLRBLK; EXEQRY; PRVBLK;
Actions:
If step not successful: stop with error (No Message)
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Control Block: menu/MAIN MENU
Starts on Page: 10

Field: 1 CHOICE Number: Length 1
Page: 10 Line: 18 Column: 39
Non-table Non-queryable Value range from 1 to 8
Automatic Help Help Message: NOTE : ORACLE IS CASE SENSITIVE.
PLEASE BE CONSISTENT WHEN
ENTERING AbCde.. DATA

KEY-NXTFLD STEP (1)

```
#EXEMACRO CASE CHOICE IS
    WHEN '1' THEN GOBLK CSGEODX1;
    WHEN '2' THEN GOBLK REPORT;
    WHEN '3' THEN GOBLK AMIRP;
    WHEN '4' THEN GOBLK SURVEY;
    WHEN '5' THEN GOBLK WELL;
    WHEN '6' THEN GOBLK GEOMAP;
    WHEN '7' THEN GOBLK COST;
    WHEN '8' THEN EXIT;
    WHEN OTHERS THEN NULL;
END CASE;
```

Actions:
If step not successful: stop with error (No Message)

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

1 record Block: csgeodx1/BMR BULLETINS/REPORTS/RECORDS
Starts on Page: 1
Table: CSGEODX1

Block-level Triggers

```
                    KEY-COMMIT STEP (1)
#EXEMACRO NULL;
Actions:
If step not successful: stop with error (No Message)
                    KEY-DELREC STEP (1)
#EXEMACRO NULL;
Actions:
If step not successful: stop with error (No Message)
                    KEY-ENTQRY STEP (1)
#EXEMACRO ENTQRY; EXETRG Q_NEXTBLOCK;
Actions:
If step not successful: stop with error (No Message)
                    KEY-EXEQRY STEP (1)
#EXEMACRO EXETRG DO_NOTHING;
Actions:
If step not successful: stop with error (No Message)
                    KEY-NXTREC STEP (1)
#EXEMACRO NXTREC; EXETRG Q_NEXTBLOCK;
Actions:
If step not successful: stop with error (No Message)
                    KEY-PRVREC STEP (1)
#EXEMACRO PRVREC; EXETRG Q_NEXTBLOCK;
Actions:
If step not successful: stop with error (No Message)
```

Field: 1 PUBNUM Number: Length 3
Page: 1 Line: 4 Column: 23
Non-enterable Queryable
No Help Message

Field: 2 VOLPARTNUM Character: Length 20
Page: 1 Line: 4 Column: 59
Non-enterable Queryable
No Help Message

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

```
Block: csgeodx1/BMR BULLETINS/REPORTS/RECORDS
       (continued)
POST-CHANGE STEP (1)
       SELECT SUBSTR(:CSGEODX1.TITLE,79,78)
       INTO DISPLAYTITLE
       FROM DUAL
       Actions:
       If step not successful: stop with error (No Message)
                                        STEP (2)
       SELECT PRICE
       INTO :CSGEODX1.TEMP
       FROM PUB_DETAILS
       WHERE :CSGEODX1.VOLPARTNUM = PUB_DETAILS.VOLPARTNUM
         AND :CSGEODX1.PUBNUM = PUB_DETAILS.PUBNUM
         AND PRICE IS NOT NULL
       Actions:
       If step IS successful: stop trigger (No Message)
                                        STEP (3)
       SELECT NVL(NUM_PAGES,0)*COST
       INTO :CSGEODX1.TEMP
       FROM PUB_DETAILS,COST
       WHERE :CSGEODX1.PUBNUM = PUB_DETAILS.PUBNUM
        AND  :CSGEODX1.VOLPARTNUM = PUB_DETAILS.VOLPARTNUM
        AND  COST_TYPE = 'COST PER PAGE'
       Actions:
       If step not successful: stop with error (No Message)
                                        STEP (4)
       SELECT NVL(LENGTH_OF_PLATES,0)*COST + :CSGEODX1.TEMP
       INTO :CSGEODX1.TEMP
       FROM PUB_DETAILS,COST
       WHERE :CSGEODX1.PUBNUM = PUB_DETAILS.PUBNUM
         AND :CSGEODX1.VOLPARTNUM = PUB_DETAILS.VOLPARTNUM
         AND COST_TYPE = 'DYELINE PER METRE'
       Actions:
       If step not successful: stop with error (No Message)

Field: 3 TITLE                                  Character: Length 240
       Page: 1   Line: 8   Column: 2                       Displayed 78
       Non-updatable
       No Help Message
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Block: csgeodx1/BMR BULLETINS/REPORTS/RECORDS
(continued)

Field: 4 STATUS — Character: Length 2
Page: 1 Line: 10 Column: 15
Non-enterable Queryable
No Help Message

Field: 5 PUBDATE — Number: Length 234
Displayed 10
Page: 1 Line: 10 Column: 40
Non-enterable Queryable
No Help Message

Field: 6 GDXNUM — Character: Length 8
Non-displayed

Field: 7 TEMP — Money(R): Length 7
Page: 1 Line: 10 Column: 67
Non-table Non-enterable

Field: 8 DISPLAYTITLE — Character: Length 78
Page: 1 Line: 9 Column: 2
Non-table Non-enterable

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

7 record Block: csgeodx2/csgeodx2
Starts on Page: 1
Table: CSGEODX2

Block-level Triggers

KEY-COMMIT STEP (1)
#EXEMACRO NULL;
Actions:
If step not successful: stop with error (No Message)
KEY-DELREC STEP (1)
#EXEMACRO NULL;
Actions:
If step not successful: stop with error (No Message)
KEY-NXTBLK STEP (1)
#EXEMACRO NXTBLK; CLRBLK;
Actions:
If step not successful: stop with error (No Message)

Field: 1 GDXNUM Character: Length 8
Non-displayed Copied from CSGEODX1.GDXNUM

Field: 2 AUTHOR Character: Length 60
Page: 1 Line: 1 Column: 2
Non-updatable
Help Message: Enter value for : AUTHOR

Field: 3 ORDER_SEQUENCE Number: Length 234
Page: 1 Line: 1 Column: 69 Displayed 2
Non-enterable Queryable
No Help Message

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

14 record Block: GDXDETAILS/BMR BULLETINS/REPORTS/RECORDS DATA
Starts on Page: 2
Table: PUB_DETAILS

Block-level Triggers

POST-QUERY STEP (1)

```
SELECT (NVL(:GDXDETAILS.NUM_PAGES,0)*COST)
INTO :GDXDETAILS.COST
FROM COST
WHERE COST_TYPE = 'COST PER PAGE'
Actions:
If step not successful: stop trigger (No Message)
```

STEP (2)

```
SELECT (NVL(:GDXDETAILS.LENGTH_OF_PLATES,0)*COST) +
       :GDXDETAILS.COST
INTO :GDXDETAILS.COST
FROM COST
WHERE COST_TYPE = 'DYELINE PER METRE'
Actions:
If step not successful: stop trigger (No Message)
```

Field: 1 PUBNUM Number: Length 3
Page: 2 Line: 1 Column: 3
Copied from CSGEODX1.PUBNUM
Help Message: Enter value for : PUBLICATION NUMBER

Field: 2 VOLPARTNUM Character: Length 20
Page: 2 Line: 1 Column: 11
Copied from CSGEODX1.VOLPARTNUM
Help Message: Enter value for : VOLPARTNUM

Field: 3 NUM_PAGES Number(R): Length 3
Page: 2 Line: 1 Column: 36
Help Message: Enter value for : NUM_PAGES

Field: 4 NUM_PLATES Number(R): Length 3
Page: 2 Line: 1 Column: 43
Help Message: Enter value for : NUM_PLATES

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Block: GDXDETAILS/BMR BULLETINS/REPORTS/RECORDS DATA
(continued)

Field: 5 LENGTH_OF_PLATES Number(R): Length 3
Page: 2 Line: 1 Column: 51
Help Message: Enter value for : LENGTH_OF_PLATES

Field: 6 PRICE Money(R): Length 7
Page: 2 Line: 1 Column: 59
Help Message: Enter value for : NUM_COPIED

```
                        KEY-NXTFLD STEP (1)
SELECT NVL(:GDXDETAILS.NUM_PAGES,0)*COST
INTO :GDXDETAILS.COST
FROM COST
WHERE COST_TYPE = 'COST PER PAGE'
Actions:
If step not successful: stop with error and display
"TRIGGER STEP FAILED, CHECK COST TABLE WHERE COST_TYPE
= 'COST PER PAGE'"
                                    STEP (2)
SELECT NVL(:GDXDETAILS.LENGTH_OF_PLATES,0)*COST +
       :GDXDETAILS.COST
INTO :GDXDETAILS.COST
FROM COST
WHERE COST_TYPE = 'DYELINE PER METRE'
Actions:
If step not successful: stop with error and display
"TRIGGER STEP FAILED, CHECK COST TABLE WHERE COST =
'DYELINE PER METRE'"
                                    STEP (3)
#EXEMACRO NXTFLD;
Actions:
If step not successful: stop with error (No Message)
                                    STEP (4)
SELECT 'X'
FROM DUAL
Actions:
If step not successful: stop with error and display
"PRESS [F3] TO COMMIT RECORD TO THE DATABASE."
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Block: GDXDETAILS/BMR BULLETINS/REPORTS/RECORDS DATA
(continued)

Field: 7 COST Money(R): Length 7
Page: 2 Line: 1 Column: 71
Non-table Non-enterable

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

```
1 record Block: REPORT/REPORTS
Starts on Page: 3
         Table: REPORT
       Default: Order by
                ORDER BY BMR_FILE_NUM
```

Block-level Triggers

```
                          KEY-ENTQRY STEP (1)
                #EXEMACRO CLRBLK; ENTQRY; EXETRG Q_NEXTBLOCK;
                Actions:
                If step not successful: stop with error (No Message)
                          KEY-EXEQRY STEP (1)
                #EXEMACRO EXEQRY; EXETRG Q_NEXTBLOCK;
                Actions:
                If step not successful: stop with error (No Message)
                          KEY-NXTBLK STEP (1)
                #EXEMACRO GOBLK AUTHOR; CLRBLK;
                Actions:
                If step not successful: stop with error (No Message)
                          KEY-NXTREC STEP (1)
                #EXEMACRO NXTREC; EXETRG Q_NEXTBLOCK;
                Actions:
                If step not successful: stop with error (No Message)
                          KEY-PRVREC STEP (1)
                #EXEMACRO PRVREC; EXETRG Q_NEXTBLOCK;
                Actions:
                If step not successful: stop with error (No Message)
                          PRE-DELETE STEP (1)
                DELETE FROM AUTHOR
                WHERE BMR_FILE_NUM = :REPORT.BMR_FILE_NUM
                Actions:
                If step not successful: stop with error (No Message)

Field: 1 BMR_FILE_NUM                          Character: Length 8
       Page: 3   Line: 4   Column: 20
       Mandatory
       Help Message: Enter value for : BMR_FILE_NUM
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

```
Block: REPORT/REPORTS
       (continued)
                                  POST-CHANGE STEP (1)
       SELECT (NVL(:REPORT.NUM_PAGES,0)*COST)
       INTO REPORT.TEMP
       FROM COST
       WHERE COST_TYPE = 'COST PER PAGE'
       Actions:
       If step not successful: stop with error (No Message)
                                        STEP (2)
       SELECT (NVL(:REPORT.LENGTH_OF_PLATES,0)*COST) +
              :REPORT.TEMP
       INTO REPORT.TOTALCOST
       FROM COST
       WHERE COST_TYPE = 'DYELINE PER METRE'
       Actions:
       If step not successful: stop with error (No Message)

Field: 2 GOVT_PRINT_NUM                          Number: Length 5
       Page: 3   Line: 4   Column: 60
       Mandatory
       Help Message: Enter value for : GOVT_PRINT_NUM


Field: 3 TITLE                                Character: Length 100
       Page: 3   Line: 6   Column: 20                    Displayed 60
       Mandatory
       Help Message: Enter value for : TITLE


Field: 4 NUM_PAGES                            Number(R): Length 3
       Page: 3   Line: 8   Column: 20
       Help Message: Enter value for : NUM_PAGES


Field: 5 NUM_PLATES                           Number(R): Length 3
       Page: 3   Line: 8   Column: 60
       Help Message: Enter value for : NUM_PLATES
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Block: REPORT/REPORTS
(continued)

Field: 6 LENGTH_OF_PLATES Number(R): Length 3
Page: 3 Line: 10 Column: 20
Help Message: Enter value for : LENGTH_OF_PLATES

```
                              KEY-NXTFLD STEP (1)
SELECT (NVL(:REPORT.NUM_PAGES,0)*COST)
INTO REPORT.TEMP
FROM COST
WHERE COST_TYPE = 'COST PER PAGE'
Actions:
If step not successful: stop with error and display
"TRIGGER1 FAILED!!!!"
                                       STEP (2)
SELECT (NVL(:REPORT.LENGTH_OF_PLATES,0)*COST) +
       :REPORT.TEMP
INTO REPORT.TOTALCOST
FROM COST
WHERE COST_TYPE = 'DYELINE PER METRE'
Actions:
If step not successful: stop with error and display
"TRIGGER FAILED!"
                                       STEP (3)
#EXEMACRO NXTFLD;
Actions:
If step not successful: stop with error (No Message)
```

Field: 7 TOTALCOST Money(R): Length 6
Page: 3 Line: 10 Column: 60
Non-table Non-enterable

Field: 8 TEMP Money(R): Length 7
Non-table Non-displayed

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

```
 9 record Block: AUTHOR/AUTHOR
 Starts on Page: 3
          Table: AUTHOR
        Default: Order by
                 ORDER BY AUTHOR


Field: 1 BMR_FILE_NUM                          Character: Length 8
       Page: 3   Line: 1   Column: 20
       Copied from REPORT.BMR_FILE_NUM
       Help Message: Enter value for : BMR_FILE_NUM


Field: 2 AUTHOR                                Character: Length 30
       Page: 3   Line: 1   Column: 50
       Help Message: Enter value for : AUTHOR
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

15 record Block: AMIRP/AMIAR CHAPTER PREPRINTS
Starts on Page: 4
Table: AMIRP

Field: 1 YEAR Number: Length 4
Page: 4 Line: 1 Column: 5
Mandatory Value range from 1900 to 2200
Help Message: Enter value for : YEAR

Field: 2 TITLE Character: Length 48
Page: 4 Line: 1 Column: 11
Mandatory
Help Message: Enter value for : TITLE

Field: 3 TOTAL_COST Money(R): Length 7
Page: 4 Line: 1 Column: 70
Non-table Non-enterable

Field: 4 NUM_PAGES Number(R): Length 3
Page: 4 Line: 1 Column: 63
Help Message: Enter value for : NUM_PAGES

POST-CHANGE STEP (1)

```
SELECT (NVL(:AMIRP.NUM_PAGES,0)*COST)
INTO :AMIRP.TOTAL_COST
FROM COST
WHERE COST_TYPE = 'COST PER PAGE'
Actions:
If step not successful: stop with error (No Message)
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

```
 1 record Block: SURVEY/MARINE SEISMIC DATA
 Starts on Page: 5
          Table: SURVEYDATA
        Default: Order by
                 ORDER BY SURVEY_NUM
```

Block-level Triggers

```
                          KEY-NXTBLK STEP (1)
                 #EXEMACRO NXTBLK; CLRBLK;
                 Actions:
                 If step not successful: stop with error (No Message)
                          PRE-DELETE STEP (1)
                 DELETE FROM SEISMICDATA
                 WHERE SEISMICDATA.SURVEY_NUM = :SURVEY.SURVEY_NUM
                 Actions:
                 If step not successful: stop with error (No Message)

Field: 1 SURVEY_NUM                                  Number: Length 3
       Page: 5   Line: 5   Column: 6
       Mandatory
       Help Message: Enter value for : SURVEY_IND


Field: 2 SURVEY_NAME                              Character: Length 50
       Page: 5   Line: 5   Column: 25
       Mandatory
       Help Message: Enter value for : SURVEY_NAME
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

5 record Block: SEISMICDATA/SEISMICDATA
Starts on Page: 5
Table: SEISMICDATA

Block-level Triggers

KEY-NXTBLK STEP (1)
#EXEMACRO NXTBLK; CLRBLK;
Actions:
If step not successful: stop with error (No Message)

PRE-DELETE STEP (1)
DELETE FROM SHOTPOINTDATA
WHERE SHOTPOINTDATA.PRODUCT_CODE =
:SEISMICDATA.PRODUCT_CODE
Actions:
If step not successful: stop with error (No Message)

Field: 1 SURVEY_NUM Number: Length 3
Page: 5 Line: 1 Column: 6
Copied from SURVEY.SURVEY_NUM Mandatory
Help Message: Enter value for : SURVEY_NUM

Field: 2 LINE_NUM Character: Length 15
Page: 5 Line: 1 Column: 14
Mandatory
Help Message: Enter value for : LINE_NUM

Field: 3 PRODUCT_CODE Character: Length 11
Page: 5 Line: 1 Column: 30
Mandatory
Help Message: Enter value for : PRODUCT_CODE

Field: 4 LENGTH_OF_SECTION Number(R): Length 3
Page: 5 Line: 1 Column: 45
Help Message: Enter value for : LENGTH_OF_SECTION

POST-CHANGE STEP (1)
SELECT (COST*NVL(:SEISMICDATA.LENGTH_OF_SECTION,0))
INTO :SEISMICDATA.DYELINE_COST
FROM COST
WHERE COST_TYPE = 'DYELINE PER METRE'
Actions:
If step not successful: stop with error (No Message)

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Block: SEISMICDATA/SEISMICDATA
(continued)

STEP (2)

```
SELECT (COST*NVL(:SEISMICDATA.LENGTH_OF_SECTION,0))
INTO :SEISMICDATA.FILM_COST
FROM COST
WHERE COST_TYPE = 'FILM PER METRE'
Actions:
If step not successful: stop with error (No Message)
```

Field: 5 PRICE Money(R): Length 7
Page: 5 Line: 1 Column: 53
Help Message: Enter value for : PRICE

Field: 6 DYELINE_COST Money(R): Length 7
Page: 5 Line: 1 Column: 61
Non-table Non-enterable

Field: 7 FILM_COST Money(R): Length 7
Page: 5 Line: 1 Column: 69
Non-table Non-enterable

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

4 record Block: SHOTPOINTDATA/SHOTPOINTDATA
Starts on Page: 5
Table: SHOTPOINTDATA

Field: 1 PRODUCT_CODE Character: Length 11
Page: 5 Line: 1 Column: 5
Copied from SEISMICDATA.PRODUCT_CODE Mandatory
Help Message: Enter value for : PRODUCT_CODE

Field: 2 PART Character: Length 5
Page: 5 Line: 1 Column: 20
Help Message: Enter value for : PART

Field: 3 PART_START Character: Length 10
Page: 5 Line: 1 Column: 28
Help Message: Enter value for : PART_START

Field: 4 PART_STOP Character: Length 10
Page: 5 Line: 1 Column: 41
Help Message: Enter value for : PART_STOP

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

1 record Block: WELL/PEDIN WELL LOGS
Starts on Page: 6
Table: WELL

Block-level Triggers

KEY-COMMIT STEP (1)
#EXEMACRO NULL;
Actions:
If step not successful: stop with error (No Message)
KEY-ENTQRY STEP (1)
#EXEMACRO ENTQRY; EXETRG Q_NEXTBLOCK;
Actions:
If step not successful: stop with error (No Message)
KEY-EXEQRY STEP (1)
#EXEMACRO EXEQRY; EXETRG Q_NEXTBLOCK;
Actions:
If step not successful: stop with error (No Message)
KEY-NXTREC STEP (1)
#EXEMACRO NXTREC; EXETRG Q_NEXTBLOCK;
Actions:
If step not successful: stop with error (No Message)
KEY-PRVREC STEP (1)
#EXEMACRO PRVREC; EXETRG Q_NEXTBLOCK;
Actions:
If step not successful: stop with error (No Message)

Field: 1 UNO Character: Length 8
Page: 6 Line: 6 Column: 2
Help Message: Enter value for : UNO

Field: 2 WELL_NAME Character: Length 30
Page: 6 Line: 6 Column: 14
Help Message: Enter value for : WELL_NAME

Field: 3 BMR_FILE_NUM Character: Length 8
Page: 6 Line: 6 Column: 46
Help Message: Enter value for : BMR_FILE_NUM

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Block: WELL/PEDIN WELL LOGS
(continued)

Field: 4 LOCATION Character: Length 3
Page: 6 Line: 6 Column: 56
Help Message: Enter value for : LOCATION

Field: 5 STATE Character: Length 4
Page: 6 Line: 6 Column: 66
Help Message: Enter value for : STATE

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

12 record Block: LOG/LOG
Starts on Page: 6
Table: LOG

Block-level Triggers

key-commit STEP (1)
#EXEMACRO NULL;
Actions:
If step not successful: stop with error (No Message)

Field: 1 UNO Character: Length 8
Non-displayed Copied from WELL.UNO

Field: 2 LOG_TYPE Character: Length 16
Page: 6 Line: 1 Column: 14
Help Message: Enter value for : LOG_TYPE

Field: 3 TOP Number(R): Length 10
Page: 6 Line: 1 Column: 37
Help Message: Enter value for : TOP

Field: 4 BOTTOM Number(R): Length 10
Page: 6 Line: 1 Column: 50
Help Message: Enter value for : BOTTOM

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

15 record Block: LOG_DETAILS/COPY SERVICE WELL LOGS
Starts on Page: 9
Table: LOG_DETAILS

Field: 1 WELL_NAME Character: Length 30
Page: 9 Line: 1 Column: 2 Displayed 20
Mandatory
Help Message: Enter value for : WELL_NAME

Field: 2 LOG_TYPE Character: Length 20
Page: 9 Line: 1 Column: 23
Mandatory
Help Message: Enter value for : LOG_TYPE

Field: 3 BMR_FILE_NUM Character: Length 8
Page: 9 Line: 1 Column: 43
Help Message: Enter value for : BMR_FILE_NUM

Field: 4 FILM_COST Money(R): Length 7
Page: 9 Line: 1 Column: 73
Non-table Non-enterable

Field: 5 IN_COPYS Character: Length 1
Page: 9 Line: 1 Column: 51
Automatic Help Help Message: Enter 'Y', if log is in Copy
Service.

Field: 6 GOVT_PRINT_NUM Number(R): Length 5
Page: 9 Line: 1 Column: 53
Help Message: Enter value for : GOVT_PRINT_NUM

Field: 7 LENGTH_OF_LOG Number(R): Length 3
Page: 9 Line: 1 Column: 59
Help Message: Enter value for : LENGTH_OF_LOG

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

```
Block: LOG_DETAILS/COPY SERVICE WELL LOGS
       (continued)
                                POST-CHANGE STEP (1)
       SELECT NVL(:LOG_DETAILS.LENGTH_OF_LOG,0)*COST
       INTO :LOG_DETAILS.DYELINE_COST
       FROM COST
       WHERE COST_TYPE = 'DYELINE PER METRE'
       Actions:
       If step not successful: stop with error (No Message)
                                          STEP (2)
       SELECT NVL(:LOG_DETAILS.LENGTH_OF_LOG,0)*COST
       INTO :LOG_DETAILS.FILM_COST
       FROM COST
       WHERE COST_TYPE = 'FILM PER METRE'
       Actions:
       If step not successful: stop with error (No Message)

Field: 8 DYELINE_COST                          Money(R): Length 7
       Page: 9   Line: 1   Column: 65
       Non-table Non-enterable
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

4 record Block: GEOMAP/MAPS
Starts on Page: 7
Table: GEOMAP

Field: 1 SHEET_NAME Character: Length 22
Page: 7 Line: 1 Column: 3
Mandatory
Help Message: Enter value for : SHEET_NAME

Field: 2 SCALE Character: Length 7
Page: 7 Line: 1 Column: 31
Help Message: Enter value for : SCALE

Field: 3 COVERAGE Character: Length 1
Page: 7 Line: 1 Column: 43
Help Message: Enter value for : COVERAGE

Field: 4 PRICE Money(R): Length 7
Page: 7 Line: 1 Column: 57
No Help Message

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

1 record Block: AEROMAP/AEROMAP
Starts on Page: 7
Table: AEROMAP

Block-level Triggers

```
                    POST-QUERY STEP (1)
#EXEMACRO EXETRG GETCOST;
Actions:
If step not successful: stop with error (No Message)
                      GETCOST STEP (1)
SELECT COST
INTO :AEROMAP.DYELINE_COST
FROM COST
WHERE COST_TYPE = 'DYELINE STANDARD MAP'
Actions:
If step not successful: stop with error (No Message)
                                STEP (2)
SELECT COST
INTO :AEROMAP.FILM_COST
FROM COST
WHERE COST_TYPE = 'FILM STANDARD MAP'
Actions:
If step not successful: stop with error (No Message)
                                STEP (3)
SELECT COST
INTO :AEROMAP.CON_DYE_COST
FROM COST
WHERE COST_TYPE = 'DYELINE CONTOUR MAP'
Actions:
If step not successful: stop with error (No Message)
                                STEP (4)
SELECT COST
INTO :AEROMAP.CON_FILM_COST
FROM COST
WHERE COST_TYPE = 'FILM CONTOUR MAP'
Actions:
If step not successful: stop with error (No Message)
```

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Block: AEROMAP/AEROMAP
(continued)

Field: 1 SHEET_NAME Character: Length 22
Page: 7 Line: 12 Column: 16
Mandatory
Help Message: Enter value for : SHEET_NAME

KEY-NXTFLD STEP (1)
#EXEMACRO EXETRG GETCOST; NXTFLD;
Actions:
If step not successful: stop with error (No Message)

Field: 2 SCALE Character: Length 7
Page: 7 Line: 12 Column: 57
Help Message: Enter value for : SCALE

Field: 3 GRID_REF_NUM Character: Length 6
Page: 7 Line: 14 Column: 16
Help Message: Enter value for : GRID_REF_NUM

Field: 4 BMR_REF_NUM Character: Length 15
Page: 7 Line: 14 Column: 57
Help Message: Enter value for : BMR_REF_NUM

Field: 5 RELEASE_DATE Date: Length 9
Page: 7 Line: 16 Column: 16
Automatic Help Help Message: DATE MUST BE IN THE FORMAT
DD-MMM-YY, EG 01-JAN-88.

Field: 6 CONTOURED Character: Length 1
Page: 7 Line: 16 Column: 37
Mandatory
Automatic Help Help Message: Enter 'Y' for contoured maps, 'N'
for uncontoured maps.

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

Block: AEROMAP/AEROMAP
(continued)

Field: 7 DATA_TYPE Character: Length 20
Page: 7 Line: 16 Column: 57
Help Message: Enter value for : DATA_TYPE

Field: 8 FILM_COST Money(R): Length 7
Page: 7 Line: 18 Column: 57
Non-table Non-enterable

Field: 9 DYELINE_COST Money(R): Length 7
Page: 7 Line: 18 Column: 37
Non-table Non-enterable

Field: 10 CON_DYE_COST Money(R): Length 7
Page: 7 Line: 19 Column: 37
Non-table Non-enterable

Field: 11 CON_FILM_COST Money(R): Length 7
Page: 7 Line: 19 Column: 57
able Non-enterable

SQL*FORMS Application CSDBASE owned by COPY SERVICE
as at 12-DEC-88

15 record Block: COST/COST TABLE
Starts on Page: 8
Table: COST

Field: 1 COST_TYPE Character: Length 20
Page: 8 Line: 1 Column: 19
Help Message: WARNING : COST TYPES SHOULD NORMALLY NOT BE ALTERED.

Field: 2 COST Money(R): Length 7
Page: 8 Line: 1 Column: 47
Help Message: Enter value for : COST

APPENDIX 4 : **DATA ENTRY SCREEN LAYOUT**

COPY SERVICE DATABASE

MAIN MENU
---- ----

1. BMR BULLETINS/REPORTS/RECORDS
2. REPORTS
3. AMIAR CHAPTER PREPRINTS
4. MARINE SEISMIC DATA
5. WELL LOGS
6. MAPS
7. COST TABLE
8. EXIT

ENTER CHOICE : _

Char Mode: Replace Page 10 Count: *0

BMR BULLETINS/REPORTS/RECORDS

PUBLICATION NUMBER ___ VOL/PART NUMBER _____

(BULLETIN = 183, REPORT = 185, RECORD = 191)

TITLE

_____

_____

STATUS __ PUBLICATION DATE _____ COST _____

AUTHOR ORDER SEQUENCE

_____ __

_____ __

_____ __

_____ __

_____ __

_____ __

_____ __

_____

Char Mode: Replace Page 1 Count: *0

GEODX BULLETINS/REPORTS/RECORDS DETAILS

PUB. NUMBER	VOLUME AND PART NUMBER	NUMBER PAGES	NUMBER PLATES	PLATE LENGTH	PRICE	COPY COST
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___

Char Mode: Replace Page 2 Count: *0

REPORTS

BMR FILE NUMBER _____ GOVT PRINTER NUMBER ____

TITLE _____

NUMBER OF PAGES ___ NUMBER OF PLATES ___

LENGTH OF PLATES ___ TOTAL COST _____

BMR FILE NUMBER	AUTHOR
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Char Mode: Replace Page 3 Count: *0

AMIR PREPRINT CHAPTERS

YEAR	TITLE	NUM PAGES	TOTAL COST

MARINE SEISMIC DATA

SURVEY NUMBER ____ SURVEY NAME _____

SURVEY NUMBER	LINE NUMBER	PRODUCT CODE	LENGTH OF SECTION	PRICE	DYELINE COST	FILM COST
____	_____	_____	____	_____	_____	_____
____	_____	_____	____	_____	_____	_____
____	_____	_____	____	_____	_____	_____
____	_____	_____	____	_____	_____	_____
____	_____	_____	____	_____	_____	_____

PRODUCT CODE	PART	PART START	PART FINISH
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____

Char Mode: Replace Page 5 Count: *0

WELL LOGS IN THE PEDIN DATABASE

PEDIN WELL NUMBER	WELL NAME	BMR FILE NUMBER	LOCATION	STATE

LOG TYPE	TOP (METRES)	BOTTOM (METRES)

Char Mode: Replace Page 6 Count: *0

WELL LOGS

WELL NAME	LOG TYPE	BMR FILE NUMBER	IN CS	GOVT PR NO	LENGTH OF LOG	DYELINE COST	FILM COST

Char Mode: Replace Page 9 Count: *0

GEOLOGICAL COMPILATION SHEETS AND GEOCHEMICAL MAPS

SHEET NAME	SCALE	COVERAGE	PRICE

GRAVITY, RADIOMETRIC AND TOTAL MAGNETIC INTENSITY MAPS

SHEET NAME _____ SCALE _____

GRID REF NUM _____ BMR REF NUM _____

RELEASE DATE _____ DATA TYPE _____

CONTOURED _ DYELINE COST _____ FILM COST _____

Char Mode: Replace Page 7 Count: *0

COPY-SERVICE CHARGES

COST TYPE	COST

APPENDIX 5 : DATA ENTRY SCREEN USER DOCUMENTATION

1 BMR BULLETINS/REPORTS/RECORDS

This screen accesses data in the GEODX database and can be used for query only. Records cannot be updated, inserted or deleted. Records can be queried by any field except author. Generally queries will be by title or vol/part number(eg 88/1234). To initiate a query, press [F2], then enter the appropriate publication number, 183 for a Bulletin, 185 for a Report or 191 for a Record, followed by the title or vol/part number of the required publication. Titles may be abbreviated using the % symbol, for example, entering 191 in the publication field followed by Gold% in the title field would retrieve all Records with titles beginning with the word Gold.

To execute the query, press [F1], and all records which satisfy the query will be retrieved. The authors appear in the author block automatically, and the order sequence indicates the 1st, 2nd, 3rd etc authors. Records can be searched through using the up and down arrow keys. If a record has already been entered in the Copy Service Database the its cost will appear in the cost field.

To move to the next screen, press [C2] twice.

2 BMR BULLETINS/REPORTS/RECORDS DETAILS

When this screen is entered, the publication number and volume/part number are copied from the previous screen. Data can be inserted in the number of pages, number of plates, plate length, and price fields as appropriate. The copy cost is calculated automatically based on the number of pages and length of plates. After a record has been entered, the cursor skips to the publication number field and the user is asked to press [F3] to commit the record to the database.

It is envisaged that data will be entered into this screen each time a request for a BMR publication has been processed. In this way the database will be gradually built up.

3 REPORTS

The report screen is designed for input of data on Well Completion, PSSA, Seismic Survey, and Geophysical Survey Reports.

Enter data in the upper block from left to right, top to bottom. Data must be entered into the BMR file number, Govt printer number and Title fields for each record. Be consistent in the use of upper and lower case for all character fields and also in the style used for entry of BMR file number. Records with inconsistent data will not be retrieved during a query. For example, the BMR file number should be entered consistently as 88/1234 not 88-1234 or any other variation. The total cost is calculated automatically based on the number of pages and length of plates.

The [C2] key can be used to enter the lower block on the screen. The BMR file number is automatically copied and authors can be entered one at a time. Authors should be entered in a consistent manner, eg surname followed by initials as in 'Bowen K.C.'.

Always commit data to the database after it has been entered by pressing [F3]. You should then get a message at the bottom of the screen, 'Transaction completed -- N records processed'.

4 AMIAR CHAPTER PREPRINTS

The cost of these publications is automatically calculated based on the number of pages copied.

Enter records from left to right and press the [NL] key after each field has been entered. Press the down arrow key to start entering a new record. Press [F3] to commit the records to the database.

5 MARINE SEISMIC DATA

To query this screen, first query the upper block for survey number or survey name. Move to the next block using the [C2] key. The survey number is automatically copied to the next block. Press [F1] to query this block and data should then appear in the other fields on the screen. To obtain shot point data on a particular line, place the cursor on the desired line, press [C2] and its product code is copied into the next block. Then press [F1] to query this block.

Please note that if a record is deleted from the upper block on the screen, then all records from the next block with the same survey number are also deleted.

6 WELL LOGS IN THE PEDIN DATABASE

The PEDIN database can be accessed for query only. Press [F2], then enter a query which will usually be for the well name or the BMR file number. The query will retrieve all logs recorded for that well and the interval over which they were run. If more than 12 logs were run for a particular well, press [C2] to go to the next block and [C4] to bring up the rest of the logs for that well.

This screen can be used to obtain more information on logs, especially on those that are not held by the copy service.

7 WELL LOGS

This screen accesses a table which contains all well logs held at the BMR. Only a proportion of these are held by the copy service. The well name, log type and BMR file number of these logs are already in the table. If the log is held in the copy service, then enter 'Y' for yes in the IN CS field. The government printer number and length of log should also be entered for these logs, and the costs of a dyeline and film reproduction are calculated automatically.

8 GEOLOGICAL COMPILATION SHEETS AND GEOCHEMICAL MAPS

The names of 1:100,000 sheets in Australia have already been entered into the sheet name field. The other fields should be updated as these maps are requested. Coverage is indicated by a 'P' or 'C' in the coverage field, for partial and complete coverage of the map area.

9 GRAVITY, RADIOMETRIC AND TOTAL MAGNETIC INTENSITY MAPS

The names of 1:250,000 sheets in Australia have already been entered in the sheet name field. The other fields should be updated as these maps are requested. If a map is contoured, enter 'Y' in the Contoured field and a greater cost will be calculated for dyeline and film reproductions.

To create a new record with the same sheet name, query the database for that name, press [F4] to create a new record, then shift-[F4] to duplicate the last record.

10 COPY SERVICE CHARGES

This screen accesses the copy service charges which are used to calculate the costs which appear on all other screens. The cost types should not be altered without consulting the Computing or Database branch. The costs, however, can be changed at will, and the copy costs appearing on other screens are then be altered automatically.