Directory of HEP Institutions
and its
Application Programs

Onno L. Wiersma
Directory of HEP Institutions
and its Application Programs
Directory of HEP Institutions
and its
Application Programs

Onno L. Wiersma
Hogere Technische School
Amsterdam

CERN TH/SIS-88:05
August 1988

TH/SIS
GENEVA 1988
This system was developed at the CERN Scientific Information Service March - August 1988 as traineeship for the curriculum of the Computer Science program, HTS, Amsterdam

CERN TH/SIS-88:05
CERN Scientific Information Service
1211 Geneva 23, Switzerland
August 1988
Contents

Part I: The actual work .................................................. 1

1 ORACLE database ..................................................... 2
   1.1 The HEP information on ORACLE database .................. 3
   1.2 Indexes on the HEP tables ...................................... 5

2 Application Programs ............................................... 6
   2.1 HEP Addresses Letters .......................................... 7
      2.1.1 The procedure of printing letters ....................... 7
   2.2 Proof-Reading List .............................................. 8
   2.3 An Application Program for the future ...................... 9

Appendix A: MIAFORM .................................................. 10

Appendix B: Listing of GLEIT1 RPT .................................. 42

Appendix C: Listing of GLEIT2 RPT .................................. 45

Appendix D: The MENU EXEC program (REXX) ...................... 50

Appendix E: Listing of SGMI.801 EXEC .............................. 55

Appendix F: Listing of SGMI.802 EXEC .............................. 56

Appendix G: Proof-Reading List Program ............................ 57

Bibliography ........................................................... 61
Figures

1. An example table called: ADDRESS ............................................. 2
2. The screens of the MENUNEW EXEC program .................................... 7
3. How to print the letters ................................................................. 8
Part 1

The actual work

During my six months stay at CERN, I learned a lot of new software features. The first few weeks I had to be familiar with the VM operating system. After that I did research on ORACLE database and the existing HEP\(^7\)-Institute’s data, which is stored on ORACLE. My main task was to develop a new data structure for the HEP Institute’s data. I made new application programs for retrieving, updating and printing of the HEP data. All of these applications can be choosen from a Menu program which is written in REXX. The Scientific Information Service is responsible for maintenance of the HEP-Institute’s data.

\(^7\) HEP – High Energy Physics
Chapter 1

ORACLE database

The ORACLE RDBMS is a software system for organizing, storing, maintaining, calculating, combining, and retrieving of information. A database is an organized collection of information. A database consists of tables, queries, and reports that are based on the tables. Entries in a database may be words, dates, numbers, or pieces of text.

In a relational database system, information is organized in the form of tables. An example of a table is given in figure 1.

![Figure 1: An example table called ADDRESS.](image)

The tables in a database are made up of rows and columns. Each column contains one kind of information. The five columns in the ADDRESS table contain country codes, address numbers, names, streets, and towns. Each row contains one set of information. There are four rows in the ADDRESS table, one for each address. Each row is made up of fields, one field for each column. Each field can hold one value or item of information. For example, the TOWN field in the row DD 12 has the value BERLIN.

Each column can hold one type of value. The most often used types are:

- CHAR

---

8 Relational Database Management System.
1.1 The HEP information on ORACLE database

The information from the HEP Institutes is stored into five tables.

The first table COUNTRY_FRENCH contains the ISO country code for nearly each country in the world and the country name in french.

<table>
<thead>
<tr>
<th>Name</th>
<th>Null?</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUNTRY</td>
<td>NOT NULL</td>
<td>VARCHAR(2)</td>
</tr>
<tr>
<td>NAME</td>
<td></td>
<td>VARCHAR(40)</td>
</tr>
</tbody>
</table>

The country_french table is made by the user "INSTS" and there is a public grant for the table, so every ORACLE user can use that table.

The second table ADDRIHEP is partly made by me. There was an existing table HEPADDR made by the user "INSTS". I added some new columns to the table, and copied it to my ORACLE account "Wiersma". The column country contains the ISO code for countries, together with the column addr_num, they form an unique key, so every Institute can be identified by one unique combination of country code and address number. There are two date type columns which contain respectively the start date of a new record and the date when the last update took place. In the columns Line1 to Line7 the actual name and address of an Institute are stored (Line7 is rarely used). The text in the Line columns is coded in SGML.\(^9\) i.e. SGML entity references to perform national characters like ç and â are used.

\(^9\) Standard Generalized Markup Language.
The columns *tela, tel1, tel2, tel3, telex, cable, tfax1, tfax2, fits, node1,* and *node2* are used for telecommunication addresses of an Institute (hence *twix* and *tel4* are not used).

The third table **RESPROG** contains the two key columns *country* and *addr_num* and five columns *L1* to *L5*. In these five columns the research program of an Institute is stored.

In the fourth table **RESTYP** you find the two key columns *country* and *addr_num* and three columns: *thept*, *chept*, and *archeps*. The table contains information about the Institute's research type. The value of the three last columns can be "X" or "NULL" (a NULL value is an empty field).

---

10 *thept* — theoretical high-energy physics

11 *chept* — experimental high-energy physics

12 *archeps* — activity related to high-energy physics
In the last table `ACCELERATOR` you find three key columns `country`, `addr_num` and `seq_num`. These three columns form together an unique key for each different accelerator. In the column `description` the name, amount of energy and sort of particles of a specific accelerator is stored.

1.2 Indexes on the HEP tables

An index to a table is similar to the index in the back of a book. Like a book’s index, it helps you find information more quickly.

If you want to find all the rows of a table that have a specified value in a certain column, a database tool can “look up” the rows in an index to the table. If there is no index for that column of that table, the database tool must scan through the entire table to find all the rows (records). Indexing a table can reduce the time the database tool takes to perform queries, especially if the table is large, like the `ADDRHEP` table.

The second use of indexes is that they can guarantee that a column or group of columns contain unique values. In all the HEP tables the combination of the columns `country` and `addr_num` must be unique so I created indexes on them. The following indexes are created:

If you try to insert a record with the same country code and address number, the database system will generate an error message.
Chapter 2

Application Programs

To update, retrieve, delete and print the information of a database, you need special application programs. The following programs I made are used to perform these tasks:

**MIFAORM**
A program written with the aid of a database tool called SQL•Forms. This program makes it possible to retrieve, update, and delete information from the HIEP-database. All of these actions can be done automatically using *Function-Keys* within the Form-program. See Appendix A for the listing.

**GLET1 RPF**
This program generates a file containing *User text* and *SGML Tags* needed to create standard letters, see Appendix B for the listing. The addresses printed on the letters are extracted from the HIEP-database.

**GLET2 RPF**
Part 2 of the letter program (see Appendix C for the listing). More information about an Institute is taken from the HIEP-database and put into a file. This file is used to produce the second page of the standard letter.

**MENUNEW EXEC**
Two menu's (figure 2) let you choose which application-program you want to execute. The program is written in REXX (Appendix D), it was an existing program, I made some changes to make it possible to execute my application programs from the menu.

**SGML801 EXEC**
This REXX program (Appendix E) cuts a file containing *User Text* and *SGML Tags* into pieces. Each piece is formatted by the SGML Formatter and after formatting, printed on the IBM3812 laser printer.

**SGML802 EXEC**
Nearly the same program as SGML 801 EXEC, the user text in the input file is different and some input file names. This program is used to print part 2 of the letters, see Appendix F for the listing.
2.1 HEP Addresses Letters

Each year the HEP-database is updated by the Scientific Information Service. To get the information from the Institutes, they send a letter to each of them. To generate part 1 of the letters, you have to choose the "O" option of the MENUNEW program. Due to a disk-space problem, part 2 of the letter must be executed from the account "LOQ", the procedure is; first: logon to LOQ, second: type GIME ABO, and at last: type MENUNEW and choose the "P" option of the menu.

2.1.1 The procedure of printing letters

The Report Generator (RPG) inserts the HEP-database data into a file called SAMP1 RPF (for part 2: SAMP2 RPF), Figure 3 shows the procedure. The SAMP RPF file is formatted by the RPG Formatter. After formatting, the output file (OUT1 DATA for part 1 and OUT2 DATA for part 2) contains User text (data from the database included) and SGML tags. The REXX program SGML801 (for part 2 SGML802) cuts the OUT DATA file in pieces so every piece contains the SGML tags for a letter and database information. Following this, the pieces are send one by one to the SGML Formatter. After a letter is formatted, it will be send to the printer.
2.2 Proof-Reading List

After updating, the database must be checked for errors. The "H" option of the MENUNEW EXEC program executes a program which prints the data of the whole database. Appendix G shows the listing of the proof-reading print program. This program is written in Fortran, and embedded in it is the SQL language. It was an existing program, I made some changes to it which make it possible to print the information stored in the new tables.
2.3 An Application Program for the future

The letters produced by the programs SGM1.801 & SGM1.802 will be send to all HEP-Institutes. The Fill-in Form which is part of the letter has to be replied by all the Institutes. When we do not receive an answer, a second letter must be prepared and send to that Institute. We could do that automatically by adding a boolean field to the ADDRIIEP table, and put a "X" into that field when receiving an answer from an institute. An program written in Fortran with embedded SQL-commands could search in the table for records without a "X" and then generate automatically a second letter. This letter could be send to an Institute, asking again for a reply.
Appendix A

MHAFORM
ADDRESS
ORACLE workspace size:
256
Validation unit:
RECORD
Block name / Description:
**PRE-FORM**
**SQL**
give access to hep db only between 8 am and 6 pm
SELECT 'v'
FROM DUAL
WHERE TO_NUMBER (TO_CHAR(SYSDATE,'HH24')) BETWEEN 8 AND 24

Message if value not found:
The MHEF-address form is available between 8am and 6pm only!! (sorry)
Must value exist Y/N:
Y
Block name / Description:
**KEY-ENTER**
**SQL**
Avemacro nextfld;

Message if value not found:
Must value exist Y/N:
Y
Block name / Description:
**KEY-UPREC**
**SQL**
display message on screen and give beep when PF2 pressed
Avemacro uprec1;

Message if value not found:
Record is locked now, so ready for update.
Must value exist Y/N:
Y
Block name / Description:
**ADDRESS12**
Enter default WHERE and ORDER BY clause:
WHERE RESP='RAHR' ORDER BY COUNTRY,ADDR_NUM

Table name:
ADDRHEP
Check for uniqueness before inserting Y/N:
Y
Display/Buffer how many records:
1 / 2
Field name:
**PRE-INSERT**
**SQL**
SELECT MAX(ADDR_NUM)+1 INTO ADDRESS12.ADDR_NUM FROM ADDRHEP
WHERE COUNTRY='ADDRESS12.COUNTRY AND ADDR_NUM>9888 GROUP BY COUNTRY

Message if value not found:
This country has no MHEF lab. in this database
Must value exist Y/N:
N
SELECT 'RAHR' INTO ADDRESS12.RESP FROM DUAL

Message if value not found:
Default value RAHR not inserted in RESP field
Must value exist Y/N:
Y
SELECT SYSDATE INTO ADDRESS12.STARTED FROM DUAL

Message if value not found:
SYSDATE NOT INSERTED IN CURRENT MHEPAD

Field name:
**PRE-UPDATE**
**SQL**
SELECT SYSDATE INTO ADDRESS12.UPDATED FROM DUAL

Message if value not found:
Sydate not inserted in column UPDATED of current address
Must value exist Y/N:
Y
Field name:
COUNTRY
Type of field:
CHAR
Length of field / Display length / Query length:
2 / 2 / 2
Is this field in the base table Y/N:
Y
Is this field part of the primary key Y/N:
Y
Field to copy primary key from:
Default value:

Page: 1
; Allow field to be entered Y/N : 
; Allow field to be updated Y/N : 
; Allow entry of query condition Y/N : 
; Hide value of field Y/N : 
; SQL> 
; SQL> 
; SELECT * FROM INSTS.COUNTRY_FRENCH WHERE COUNTRY<>ADDRESS2.COUNTRY 
; Message if value not found : 
; Must value exist Y/N : 
; SELECT MAX(ADDR_NUM)+1 INTO ADDRESS2.ADDR_NUM FROM ADDRHEP 
; WHERE COUNTRY<>ADDRESS2.COUNTRY AND ADDR_NUM.<=0 GROUP BY COUNTRY 
; Message if value not found : 
; Country not yet present in ADDRHEP table, give number 1 to ADDR_NUM 
; Must value exist Y/N : 
; Is field mandatory Y/N : 
; Is field fixed length Y/N : 
; Auto jump to next field Y/N : 
; Convert field to upper case Y/N : 
; Help message : 
; Enter value for : COUNTRY 
; Lowest value : 
; Highest value : 
; Field name : 
; ADDR_NUM 
; Type of field : 
; NUMBER 
; Length of field / Display length / Query length : 
; 4 / 4 / 4 
; Is this field in the base table Y/N : 
; Is this field part of the primary key Y/N : 
; Default value : 
; Page : 
; Line : 
; Column : 
; Prompt : 
; Allow field to be entered Y/N : 
; Allow field to be updated Y/N : 
; Allow entry of query condition Y/N : 
; Hide value of field Y/N : 
; SQL> 
; Is field mandatory Y/N : 
; Is field fixed length Y/N : 
; Auto jump to next field Y/N : 
; Convert field to upper case Y/N : 
; Help message : 
; Lowest value : 
; Highest value : 
; Field name : 
; INST_NUM 
; Type of field : 
; NUMBER 
; Length of field / Display length / Query length : 
; 4 / 4 / 4 
; Is this field in the base table Y/N : 
; Is this field part of the primary key Y/N : 
; N
; Default value :
; Page :
; 1
; Line :
; 1
; Column :
; 64
; Prompt :
; Allow field to be entered Y/N :
; Y
; Allow field to be updated Y/N :
; Y
; Allow entry of query condition Y/N :
; Y
; Hide value of field Y/N :
; N
; SQL>
; Is field mandatory Y/N :
; N
; Is field fixed length Y/N :
; N
; Auto jump to next field Y/N :
; N
; Convert field to upper case Y/N :
; Y
; Help message :
; Lowest value :
; Highest value :

; Field name :
; TOWN
; Type of field :
; CHR
; Length of field / Display length / Query length :
; 38 / 38 / 38
; Is this field in the base table Y/N :
; Y
; Is this field part of the primary key Y/N :
; N
; Default value :
; Page :
; 1
; Line :
; 2
; Column :
; 20
; Prompt :
; Allow field to be entered Y/N :
; Y
; Allow field to be updated Y/N :
; Y
; Allow entry of query condition Y/N :
; Y
; Hide value of field Y/N :
; N
; SQL>
; Is field mandatory Y/N :
; N
; Is field fixed length Y/N :
; N
; Auto jump to next field Y/N :
; N
; Convert field to upper case Y/N :
; Y
; Help message :
; Lowest value :
; Highest value :

; Field name :
; RESP
; Type of field :
; CHR
; Length of field / Display length / Query length :
; 4 / 4 / 4
; Is this field in the base table Y/N :
; Y
; Is this field part of the primary key Y/N :
; N
; Default value :
; Page :
; 1
; Line :
; 2
; Column :
; 64
; Prompt :
; Allow field to be entered Y/N :

say
; Allow field to be updated Y/N :
y
; Allow entry of query condition Y/N :
y
; Hide value of field Y/N :
N
; SQL>
; Is field mandatory Y/N :
N
; Is field fixed length Y/N :
N
; Auto jump to next field Y/N :
N
; Convert field to upper case Y/N :
y
; Help message :
Enter value for : RESP
; Lowest value :

; Highest value :

; Field name :
STARTED
; Type of field :
DATE
; Length of field / Display length / Query length :
9 / 9 / 9
; Is this field in the base table Y/N :
Y
; Is this field part of the primary key Y/N :
N
; Default value :

; Page :
1
; Line :
3
; Column :
48
; Prompt :
; Allow field to be entered Y/N :
y
; Allow entry of query condition Y/N :
y
; Hide value of field Y/N :
N
; SQL>
; Field name :
UPDATED
; Type of field :
DATE
; Length of field / Display length / Query length :
9 / 9 / 9
; Is this field in the base table Y/N :
Y
; Is this field part of the primary key Y/N :
N
; Default value :

; Page :
1
; Line :
3
; Column :
64
; Prompt :
; Allow field to be entered Y/N :
y
; Allow entry of query condition Y/N :
y
; Hide value of field Y/N :
N
; SQL>
; Field name :
LINE
; Type of field :
CHAR
; Length of field / Display length / Query length :
50 / 50 / 50
; Is this field in the base table Y/N :
Y
; Is this field part of the primary key Y/N :
N
; Default value :

; Page :
1
; Line :
4
; Column :
20
; Prompt :
; Lowest value:
; Highest value:
; Field name: LINE2
; Type of field: CHAR
; Length of field / Display length / Query length: 50 / 50 / 50
; Is this field in the base table Y/N: Y
; Is this field part of the primary key Y/N: N
; Default value:
; Page: 1
; Line: 5
; Column: 20
; Prompt:
; Allow field to be entered Y/N: Y
; Allow field to be updated Y/N: Y
; Allow entry of query condition Y/N: Y
; Hide value of field Y/N: N
; SOL:
**KEY-DUPFLO**
/
; SOL>
COPY ADDRESS12.LINE6 ADDRESS12.LINE7
/
; Message if value not found:
Not possible to transfer line7 to line6   (sorry)

; Must value exist Y/N: Y
COPY ADDRESS12.LINE6 ADDRESS12.LINE8
/
; Message if value not found:
Not possible to transfer line5 to line6   (sorry)

; Must value exist Y/N: Y
COPY ADDRESS12.LINE4 ADDRESS12.LINE6
/
; Message if value not found:
Not possible to transfer line5 to line4   (sorry)

; Must value exist Y/N: Y

; This trigger inserts a line.
COPY ADDRESS12.LINE2 ADDRESS12.LINE3
/
; Message if value not found:
Not possible to transfer line2 to line3   (sorry)

; Must value exist Y/N: Y
; SOL>
**KEY-LISTVAL**
/
; SOL>
COPY ADDRESS12.LINE3 ADDRESS12.LINE2
/
; Message if value not found:
Not possible to indent line(s)   (sorry)

; Must value exist Y/N: Y
COPY ADDRESS12.LINE4 ADDRESS12.LINE3
/
; Message if value not found:
Not possible to indent line(s)   (sorry)

; Must value exist Y/N: Y
COPY ADDRESS12.LINE5 ADDRESS12.LINE4
/
; Message if value not found:
Not possible to indent line(s)   (sorry)

; Must value exist Y/N: Y
COPY ADDRESS12.LINE6 ADDRESS12.LINE5
/
; Message if value not found:
Not possible to indent line(s)   (sorry)

; Must value exist Y/N: Y

; Indent line(s)
COPY ADDRESS12.LINE7 ADDRESS12.LINE6
; Message if value not found:
Not possible to indent line(s) (sorry)
Must value exist Y/N:
Y
Is field mandatory Y/N:
N
Is field fixed length Y/N:
N
Auto jump to next field Y/N:
N
Convert field to upper case Y/N:
N
Help message:
Enter value for: LINE2
Lowest value: 
Highest value: 
Field name:
LINE3
Type of field:
CHAR
Length of field / Display length / Query length:
50 / 50 / 50
Is this field in the base table Y/N:
Y
Is this field part of the primary key Y/N:
N
Default value:

Page:
1
Line:
6
Column:
28
Prompt:

Allow field to be entered Y/N:
Y
Allow field to be updated Y/N:
Y
Allow entry of query condition Y/N:
Y
Hide value of field Y/N:
N
SQL:
*KEY-DUPTLD
/*
SQL:
This trigger inserts a line.
COPY ADDRESS12.LINE6 ADDRESS12.LINE7
/*
Message if value not found:
Not possible to transfer line 6 to line7 (sorry)
Must value exist Y/N:
Y
COPY ADDRESS12.LINES ADDRESS12.LINE6
/*
Message if value not found:
Not possible to transfer line 5 to line6 (sorry)
Must value exist Y/N:
Y
COPY ADDRESS12.LINE4 ADDRESS12.LINES
/*
Message if value not found:
Not possible to transfer line 4 to line5 (sorry)
Must value exist Y/N:
Y
COPY ADDRESS12.LINE3 ADDRESS12.LINE4
/*
Message if value not found:
Not possible to transfer line 3 to line4 (sorry)
Must value exist Y/N:
Y
SQL:
*KEY-LISTVOL
/*
COPY ADDRESS12.LINE4 ADDRESS12.LINE3
/*
Message if value not found:
Not possible to indent line(s) (sorry)
Must value exist Y/N:
Y
COPY ADDRESS12.LINES ADDRESS12.LINE4
/*
Message if value not found:
Not possible to indent line(s) (sorry)
Must value exist Y/N:
Y
COPY ADDRESS12.LINE6 ADDRESS12.LINES
/*
Message if value not found:
Not possible to indent line(s) (sorry)
Must value exist Y/N:
Y
Indent line(s)
COPY ADDRESS12.LINE7 ADDRESS12.LINE6
/*
Message if value not found:
Not possible to indent line(s) (sorry)
Must value exist Y/N:
Y
Is field mandatory Y/N:
N
Is field fixed length Y/N:
N
Auto jump to next field Y/N:
N
Convert field to upper case Y/N:
N
Help message:
Enter value for: LINE3
Lowest value:

Highest value:

Field name:
LINE
Type of field:
CHAR
Length of field / Display length / Query length: 5B / 5B / 5B
Is this field in the base table Y/N:
Y
Is this field part of the primary key Y/N:
N
Default value:

; Page:
1
; Line:
/
; Column:
20
; Prompt:

; Allow field to be entered Y/N:
Y
; Allow field to be updated Y/N:
Y
; Allow entry of query condition Y/N:
Y
; Hide value of field Y/N:
N
; SOL>
; KEY-DUFLD
/
; SOL>
; This trigger inserts a line.
; COPY ADDRESS12.LINER ADDRESS12.LINE7
/
; Message if value not found:
; Not possible to transfer line6 to line7 (sorry)
; Must value exist Y/N:
Y
; COPY ADDRESS12.LINER ADDRESS12.LINE6
/
; Message if value not found:
; Not possible to transfer line5 to line6 (sorry)
; Must value exist Y/N:
Y
; COPY ADDRESS12.LINER ADDRESS12.LINE5
/
; Message if value not found:
; Not possible to transfer line4 to line5 (sorry)
; Must value exist Y/N:
Y
; SOL>
; KEY-LISTVAL
/
; SOL>
; COPY ADDRESS12.LINES ADDRESS12.LINE4
/
; Message if value not found:
; Not possible to indent line(s) (sorry)
; Must value exist Y/N:
Y
; COPY ADDRESS12.LINE6 ADDRESS12.LINES
/
; Message if value not found:
; Not possible to indent line(s) (sorry)
; Must value exist Y/N:
Y
; Indent line(s)
; ADDRESS12.LINE7 ADDRESS12.LINE6
/
; Message if value not found:
; Not possible to indent line(s) (sorry)
; Must value exist Y/N:
Y
; Is field mandatory Y/N:
N
; Is field fixed length Y/N:
N
; Auto jump to next field Y/N:
N
; Convert field to upper case Y/N:
N
; Help message:
Enter value for: LINE4
; Lowest value:

; Highest value:

; Field name:
LINES
; Type of field:
CHAR
; Length of field / Display length / Query length:
50 / 50 / 50
; Is this field in the base table Y/N:
Y
; Is this field part of the primary key Y/N:
N
; Default value:

; Page:
1
; Line:
6
; Column:
20
/ 
/SOL>
/Indent line
/ADDRESS12.LINE7 ADDRESS12.LINE6
/Message if value not found: 
/Not possible to indent line (sorry)
/Value exists Y/N : Y
/Is field mandatory Y/N : Y
/Is field fixed length Y/N : Y
/Go to next field Y/N : Y
/Convert field to upper case Y/N : Y
/Help message:
/Enter value for : LINE7
/Select value :
/Choose value :
/Field name: 
/LINE7
/Type of field: CHR
/Is field mandatory Y/N : Y
/Is field has display length Y/N : Y
/Is this field in the base table Y/N : Y
/Is this field part of the primary key Y/N : Y
/Default value :

/Page : 
/1
/Line : 
/10
/Column : 
/20
/Prompt :
/Allow field to be entered Y/N : Y
/Allow field to be updated Y/N : Y
/Allow entry of query condition Y/N : Y
/Hide value of field Y/N : Y
/SOL> 
/Is field mandatory Y/N : Y
/Is field fixed length Y/N : Y
/Go to next field Y/N : Y
/Convert field to upper case Y/N : Y
/Help message:
/Enter value for : LINE7
/Select value :
/Choose value :
/Field name: 
/TELA
/Type of field: CHR
/Is field has display length Y/N : Y
/Is this field in the base table Y/N : Y
/Is this field part of the primary key Y/N : Y
/Default value :

/Page : 
/1
/Line : 
/10
N
;SQL>
;Is field mandatory Y/N :
N
;Is field fixed length Y/N :
N
;Auto jump to next field Y/N :
N
;Convert field to upper case Y/N :
N
;Help message :
Enter value for :
TEL2
;Lowest value :

;Highest value :

;Field name :
TEL3
;Type of field :
CHAR
;Length of field / Display length / Query length :
40 / 40 / 40
;Is this field in the base table Y/N :
Y
;Is this field part of the primary key Y/N :
N
;Default value :

;Page :
1
;Line :
14
;Column :
26
;Prompt :
;Allow field to be entered Y/N :
Y
;Allow field to be updated Y/N :
Y
;Allow entry of query condition Y/N :
Y
;Hide value of field Y/N :
N
;SQL>
;Is field mandatory Y/N :
N
;Is field fixed length Y/N :
N
;Auto jump to next field Y/N :
N
;Convert field to upper case Y/N :
N
;Help message :
Enter value for :
TEL3
;Lowest value :

;Highest value :

;Field name :
F15
;Type of field :
CHAR
;Length of field / Display length / Query length :
40 / 40 / 40
;Is this field in the base table Y/N :
Y
;Is this field part of the primary key Y/N :
N
;Default value :

;Page :
1
;Line :
15
;Column :
28
;Prompt :
;Allow field to be entered Y/N :
Y
;Allow field to be updated Y/N :
Y
;Allow entry of query condition Y/N :
Y
;Hide value of field Y/N :
N
;SQL>
;Is field mandatory Y/N :
N
;Is field fixed length Y/N :
N
;Auto jump to next field Y/N :
N
;Convert field to upper case Y/N :
N

- 23 -
Help message:
Enter value for: FTS
Lowest value:

Highest value:

Field name:
Type of field:
Length of field / Display length / Query length:
48 / 48 / 48
In this field in the base table Y/N:
Y
Is this field part of the primary key Y/N:
N
Default value:

Page:
1
Line:
16
Column:
28
Prompt:
Allow field to be entered Y/N:
Y
Allow field to be updated Y/N:
Y
Allow entry of query condition Y/N:
Y
Hide value of field Y/N:
N
SQL:
Is field mandatory Y/N:
N
Is field fixed length Y/N:
N
Auto jump to next field Y/N:
N
Convert field to upper case Y/N:
N
Help message:
Enter value for:
Lowest value:

Highest value:

Field name:
Type of field:
Length of field / Display length / Query length:
48 / 48 / 48
In this field in the base table Y/N:
Y
Is this field part of the primary key Y/N:
N
Default value:

Page:
1
Line:
17
Column:
28
Prompt:
Allow field to be entered Y/N:
Y
Allow field to be updated Y/N:
Y
Allow entry of query condition Y/N:
Y
Hide value of field Y/N:
N
SQL:
Is field mandatory Y/N:
N
Is field fixed length Y/N:
N
Auto jump to next field Y/N:
N
Convert field to upper case Y/N:
N
Help message:
Enter value for:
Lowest value:

Highest value:

Field name:
Type of field:
Length of field / Display length / Query length:
48 / 48 / 48
; Is this field in the base table Y/N:
Y
; Is this field part of the primary key Y/N:
N
; Default value:

; Page:
1
; Line:
19
; Column:
20
; Prompt:

; Allow field to be entered Y/N:
Y
; Allow field to be updated Y/N:
Y
; Allow entry of query condition Y/N:
Y
; Hide value of field Y/N:
N
; SOL>

; Is field mandatory Y/N:
N
; Is field fixed length Y/N:
N
; Auto jump to next field Y/N:
N
; Convert field to upper case Y/N:
N
; Help message:
Enter value for: TFAX1
; Lowest value:

; Highest value:
; Field name:
TFAX2
; Type of field:
CHSR
; Length of field / Display length / Query length:
48 / 48 / 48
; Is this field in the base table Y/N:
Y
; Is this field part of the primary key Y/N:
N
; Default value:

; Page:
1
; Line:
19
; Column:
20
; Prompt:

; Allow field to be entered Y/N:
Y
; Allow field to be updated Y/N:
Y
; Allow entry of query condition Y/N:
Y
; Hide value of field Y/N:
N
; SOL>

; Is field mandatory Y/N:
N
; Is field fixed length Y/N:
N
; Auto jump to next field Y/N:
N
; Convert field to upper case Y/N:
N
; Help message:
Enter value for: TFAX2
; Lowest value:

; Highest value:
; Field name:
MODE1
; Type of field:
CHSR
; Length of field / Display length / Query length:
48 / 48 / 48
; Is this field in the base table Y/N:
Y
; Is this field part of the primary key Y/N:
N
; Default value:

; Page:
1
; Line:
20
COLUMN:

20

Prompt:

ALLOW field to be entered Y/N: 
e
ALLOW field to be updated Y/N: 
Y
ALLOW entry of query condition Y/N: 
Y
HIDE value of field Y/N: 
N
SO

Is field mandatory Y/N: 
N
Is field fixed length Y/N: 
N
Auto jump to next field Y/N: 
N
Convert field to upper case Y/N: 
N
Help message:
Enter value for: NODE1

Lowest value:

Highest value:

Field name:

NODE2

Type of field:

CHAR

Length of field / Display length / Query length: 
48 / 48 / 48

Is this field in the base table Y/N: 
Y
Is this field part of the primary key Y/N: 
N
Default value:

Page:

1

Line:

21

Column:

28

Prompt:

ALLOW field to be entered Y/N: 
e
ALLOW field to be updated Y/N: 
Y
ALLOW entry of query condition Y/N: 
Y
HIDE value of field Y/N: 
N
SO

Is field mandatory Y/N: 
N
Is field fixed length Y/N: 
N
Auto jump to next field Y/N: 
N
Convert field to upper case Y/N: 
N
Help message:
Enter value for: NODE2

Lowest value:

Highest value:

Field name:

Block name / Description:

RCCEL/

Table name:

RODIREP

Check for uniqueness before inserting Y/N: 
N

Display/Buffer how many records:

1 / 2

Field name:

COUNTRY

Type of field:

CHAR

Length of field / Display length / Query length: 
2 / 2 / 2

Is this field in the base table Y/N: 
Y
Is this field part of the primary key Y/N: 
Y

Field to copy primary key from:

ADDRESS12.COUNTRY

Page:

1

Line:

2

Column:
20
;Prompt :
-Allow field to be entered Y/N :
=*Y
-Allow field to be updated Y/N :
=N
-Allow entry of query condition Y/N :
=Y
-Hide value of field Y/N :
=N
;SQL>
**POST-CHANGE**
/
;SQL
SELECT * FROM INSTS.COUNTRY_FRENCH WHERE COUNTRY=ACCEL.COUNTRY
/
;Message if value not found :
This code is not in the COUNTRY_FRENCH table !
Must value exist Y/N :
=Y
SELECT MAX(ADDR_NUM)+1 INTO ACCEL.ADDR_NUM FROM HEPADDR
WHERE COUNTRY=ACCEL.COUNTRY GROUP BY COUNTRY
;
;Message if value not found :
Country not yet present in HEPADDR table, give number 1 to ADDR_NUM
Must value exist Y/N :
=N
-Is field mandatory Y/N :
=Y
-Is field fixed length Y/N :
=Y
-Auto jump to next field Y/N :
=N
-Convert field to upper case Y/N :
=Y
-Help message :
Enter value for : COUNTRY
-Lowest value :
-Highest value :

-Field name :
ADDR_NUM
-Type of field :
NUMBER
-Length of field / Display length / Query length :
4 / 4 / 4
-Is this field in the base table Y/N :
=*Y
-Is this field part of the primary key Y/N :
=N
-Copy field value from :
ADDRESS12.ADDR_NUM
;Page :
2
-Line :
2
-Column :
48
-Prompt :
-Allow field to be entered Y/N :
=*Y
-Allow field to be updated Y/N :
=N
-Allow entry of query condition Y/N :
=Y
-Hide value of field Y/N :
=N
;SQL>
-Is field mandatory Y/N :
=Y
-Is field fixed length Y/N :
=N
-Auto jump to next field Y/N :
=Y
-Convert field to upper case Y/N :
=N
-Help message :
-Lowest value :
-Highest value :

-Field name :
RESP
-Type of field :
CHAR
-Length of field / Display length / Query length :
4 / 4 / 4
-Is this field in the base table Y/N :
=*Y
-Is this field part of the primary key Y/N :
=N
-Copy field value from :
ADDRESS12.RESP
;Page :
;Field name:
;Type of field:
;DATE

;Length of field / Display length / Query length:
S / S / S

;Is this field in the base table Y/N:
Y

;Is this field part of the primary key Y/N:
N

;Default value:

;Highest value:

;Lowest value:

;Prompt:

;Allow field to be entered Y/N:
eY

;Allow field to be updated Y/N:
Y

;Allow entry of query condition Y/N:
Y

;Hide value of field Y/N:
N

;SOL>

;Is field mandatory Y/N:
Y

;Is field fixed length Y/N:
Y

;Auto jump to next field Y/N:
N

;Convert field to upper case Y/N:
Y

;Help message:

;Enter value for:

;RESP

;RESP

;Highest value:

;Lowest value:

;Prompt:

;Allow field to be entered Y/N:
eY

;Allow field to be updated Y/N:
Y

;Allow entry of query condition Y/N:
Y

;Hide value of field Y/N:
N

;SOL>

;Is field mandatory Y/N:
Y

;Is field fixed length Y/N:
Y

;Auto jump to next field Y/N:
N

;Convert field to upper case Y/N:
Y

;Help message:

;Highest value:

;Lowest value:

;Field name:
;Type of field:
;DATE

;Length of field / Display length / Query length:
S / S / S

;Is this field in the base table Y/N:
eY

;Is this field part of the primary key Y/N:
N

;Copy field value from:

;ADDRESS1, UPDATED

;Page:

;Line:

;Column:

;Prompt:

;Allow field to be entered Y/N:
eY

;Allow field to be updated Y/N:
Y
\text{Allow entry of query condition Y/N :} 
Y 
\text{Hide value of field Y/N :} 
N 
\text{SQL} 

; Is field mandatory Y/N : N
; Is field fixed length Y/N : Y
; Auto jump to next field Y/N : N
; Convert field to upper case Y/N : Y
; Help message :
; Lowest value :
; Highest value :
; Field name : LINE1
; Type of field : CHAR
; Length of field / Display length / Query length : 58 / 58 / 58
; Is this field in the base table Y/N : Y
; Is this field part of the primary key Y/N : N
; Copy field value from :
; ADDRESS12.LINE1
; Page :
; 2
; Line :
; 5
; Column :
; 20
; Prompt :
; Allow field to be entered Y/N : ey
; Allow field to be updated Y/N : Y
; Allow entry of query condition Y/N : Y
; Hide value of field Y/N : N
; SQL>
; Is field mandatory Y/N : N
; Is field fixed length Y/N : N
; Auto jump to next field Y/N : N
; Convert field to upper case Y/N : N
; Help message :
; Enter value for : LINE1
; Lowest value :
; Highest value :
; Field name : LINE2
; Type of field : CHAR
; Length of field / Display length / Query length : 58 / 58 / 58
; Is this field in the base table Y/N : Y
; Is this field part of the primary key Y/N : N
; Copy field value from :
; ADDRESS12.LINE2
; Page :
; 2
; Line :
; 6
; Column :
; 20
; Prompt :
; Allow field to be entered Y/N : ey
; Allow field to be updated Y/N : Y
; Allow entry of query condition Y/N : Y
; Hide value of field Y/N : N
; SQL>
; Is field mandatory Y/N : N
; Is field fixed length Y/N : N
; Auto jump to next field Y/N : N
; Convert field to upper case Y/N : N
; Help message :
; Enter value for : LINE2
; Lowest value :
;Highest value:
;Field name: LINE3
;Type of field: CHR
;Length of field / Display length / Query length:
5B / 5B / 5B
;Is this field in the base table Y/N:
Y
;Is this field part of the primary key Y/N:
N
;Copy field value from:
ADDRESS12.LINE3
;Page: 2
;Line: 7
;Column: 20
;Prompt:
;Allow field to be entered Y/N:
Y
;Allow field to be updated Y/N:
Y
;Allow entry of query condition Y/N:
Y
;Hide value of field Y/N:
N
;SQL>
;Is field mandatory Y/N:
N
;Is field fixed length Y/N:
N
;Auto jump to next field Y/N:
N
;Convert field to upper case Y/N:
N
;Help message:
Enter value for: LINE3
;Lowest value:

;Highest value:
;Field name: LINE4
;Type of field: CHR
;Length of field / Display length / Query length:
5B / 5B / 5B
;Is this field in the base table Y/N:
Y
;Is this field part of the primary key Y/N:
N
;Copy field value from:
ADDRESS12.LINE4
;Page: 2
;Line: 8
;Column: 20
;Prompt:
;Allow field to be entered Y/N:
Y
;Allow field to be updated Y/N:
Y
;Allow entry of query condition Y/N:
Y
;Hide value of field Y/N:
N
;SQL>
;Is field mandatory Y/N:
N
;Is field fixed length Y/N:
N
;Auto jump to next field Y/N:
N
;Convert field to upper case Y/N:
N
;Help message:
Enter value for: LINE4
;Lowest value:

;Highest value:
;Field name: LINE5
;Type of field: CHR
;Length of field / Display length / Query length:
5B / 5B / 5B
;Is this field in the base table Y/N:
Y
Is this field part of the primary key Y/N: N
Copy field value from:
ADDRESS12.LINE
Page: 2
Line: 9
Column: 28
Prompt:
Allow field to be entered Y/N: Y
Allow field to be updated Y/N: Y
Allow entry of query condition Y/N: Y
Hide value of field Y/N: N
;SOL>
Is field mandatory Y/N: N
Is field fixed length Y/N: N
Auto jump to next field Y/N: N
Convert field to upper case Y/N: N
Help message:
Enter value for: LINE5
Lowest value:
Highest value:
Field name:
LINE
Type of field:
CHAR
Length of field / Display length / Query length:
50 / 50 / 50
Is this field in the base table Y/N: Y
Is this field part of the primary key Y/N: N
Copy field value from:
ADDRESS12.LINE6
Page: 2
Line: 10
Column: 28
Prompt:
Allow field to be entered Y/N: Y
Allow field to be updated Y/N: Y
 Allow entry of query condition Y/N: Y
Hide value of field Y/N: N
;SOL>
Is field mandatory Y/N: N
Is field fixed length Y/N: N
Auto jump to next field Y/N: N
Convert field to upper case Y/N: N
Help message:
Enter value for: LINE6
Lowest value:
Highest value:
Field name:
LINE7
Type of field:
CHAR
Length of field / Display length / Query length:
50 / 50 / 50
Is this field in the base table Y/N: Y
Is this field part of the primary key Y/N: N
Copy field value from:
ADDRESS12.LINE7
Page: 2
Line: 11
Column: 28
Prompt:
Field name: COUNTRY
Type of field: CHAR
Length of field / Display length / Query length: 2 / 4 / 1
Is this field in the base table Y/N: Y
Is this field part of the primary key Y/N: Y
Field to copy primary key from: ACCEL.COUNTRY
Page: 2
Line: 1
Column: 2
Prompt:
Allow field to be entered Y/N: Y
Allow field to be updated Y/N: N
Allow entry of query condition Y/N: Y
Hide value of field Y/N: N

Field name: ADDR_NUM
Type of field: NUMER
Length of field / Display length / Query length: 4 / 4 / 4
Is this field in the base table Y/N: Y
Is this field part of the primary key Y/N: Y
Field to copy primary key from: ACCEL.ADDR_NUM
Page: 2
Line: 1
Column:

Prompt:

Allow field to be entered Y/N:
Y
Allow field to be updated Y/N:
N
Allow entry of query condition Y/N:
Y
Hide value of field Y/N:
N
SQL>

Is field mandatory Y/N:
Y
Is field fixed length Y/N:
N
Auto jump to next field Y/N:
N
Convert field to upper case Y/N:
N
Help message:

Lowest value:

Highest value:

Field name:
SEQ NUM

Type of field:
NUMBER

Length of field / Display length / Query length:
1 / 1 / 0

Is this field in the base table Y/N:
Y

Is this field part of the primary key Y/N:
Y

Field to copy primary key from:

Default value:

Page:

Line:

Column:
16

Prompt:

Allow field to be entered Y/N:
Y
Allow entry of query condition Y/N:
Y
Hide value of field Y/N:
N
SQL>

Is field fixed length Y/N:
N
Auto jump to next field Y/N:
N
Convert field to upper case Y/N:
N
Help message:

Lowest value:

Highest value:

Field name:
DESCRIPTION

Type of field:
CHAR

Length of field / Display length / Query length:
40 / 40 / 0

Is this field in the base table Y/N:
Y

Is this field part of the primary key Y/N:
N
Default value:

Page:

Line:

Column:
20

Prompt:

Allow field to be entered Y/N:
Y
Allow field to be updated Y/N:
Y
Allow entry of query condition Y/N:
Y
Hide value of field Y/N:
N
SQL>

— 34 —
; Is field mandatory Y/N: N
; Is field fixed length Y/N: N
; Auto jump to next field Y/N: N
; Convert field to upper case Y/N: N
; Help message:
; Lowest value:
; Highest value:
; Field name:
; Block name / Description:
RESEARCH/RESEARCH
; Table name:
RESTYP
; Check for uniqueness before inserting Y/N: N
; Display/Buffer how many records: 1 / 1
; Field name:
COUNTRY
; Type of field:
CHAR
; Length of field / Display length / Query length: 2 / 2 / 2
; Is this field in the base table Y/N: Y
; Is this field part of the primary key Y/N: Y
; Field to copy primary key from:
ADDRESS12.COUNTRY
; Page:
3
; Line:
3
; Column:
26
; Prompt:
; Allow field to be entered Y/N: Y
; Allow field to be updated Y/N: N
; Allow entry of query condition Y/N: Y
; Hide value of field Y/N: N
; ISOL>
; Is field mandatory Y/N: Y
; Is field fixed length Y/N: Y
; Auto jump to next field Y/N: N
; Convert field to upper case Y/N: N
; Help message:
; Lowest value:
; Highest value:
; Field name:
ADDR_NUM
; Type of field:
NUMBER
; Length of field / Display length / Query length: 5 / 5 / 5
; Is this field in the base table Y/N: Y
; Is this field part of the primary key Y/N: Y
; Copy field value from:
ADDRESS12.ADDR_NUM
; Page:
3
; Line:
3
; Column:
58
; Prompt:
; Allow field to be entered Y/N: Y
; Allow field to be updated Y/N: N
; Allow entry of query condition Y/N: Y
; Hide value of field Y/N: N
; ISOL>
$Highest value$

$Field name$
RESE5P

$Type of field$
CHR

$Length of field / Display length / Query length$
1 / 1 / 1

$Is this field in the base table Y/N$
Y

$Is this field part of the primary key Y/N$
N

$Default value$

$Page$
3

$Line$
9

$Column$
51

$Prompt$

$Allow field to be entered Y/N$
Y

$Allow field to be updated Y/N$
Y

$Allow entry of query condition Y/N$
Y

$Hide value of field Y/N$
N

$ISOL$

$Is field mandatory Y/N$
N

$Is field fixed length Y/N$
Y

$Auto jump to next field Y/N$
Y

$Convert field to upper case Y/N$
Y

$Help message$
Enter X or 0

$Lowest value$

$Highest value$

$Block name / Description$
RESEARCH/RESEARCH PROGRAMS

$table name$
RES5PRO

$Check for uniqueness before inserting Y/N$
N

$Display/Buffer how many records$
1 / 1

$Field name$
COUNTRY

$Type of field$
CHR

$Length of field / Display length / Query length$
2 / 2 / 2

$Is this field in the base table Y/N$
Y

$Is this field part of the primary key Y/N$
Y

$Field to copy primary key from$
ADDRESS2.COUNTRY

$Page$
3

$Line$
12

$Column$
26

$Prompt$

$Allow field to be entered Y/N$
Y

$Allow field to be updated Y/N$
N

$Allow entry of query condition Y/N$
Y

$Hide value of field Y/N$
N

$ISOL$

$Is field mandatory Y/N$
Y

$Is field fixed length Y/N$
Y

$Auto jump to next field Y/N$
N

$Convert field to upper case Y/N$
Y

$Help message$

$Lowest value$

- 37 -
;Highest value:
;Field name: L1
;Type of field: CHR
;Length of field / Display length / Query length: 50 / 50 / 50
;Is this field in the base table Y/N: Y
;Is this field part of the primary key Y/N: Y
;Copy field value from: ADDRESS1.DOOR_NUM
;Page: 3
;Line: 16
;Column: 17
;Prompt:
;Allow field to be entered Y/N: Y
;Allow field to be updated Y/N: N
;Allow entry of query condition Y/N: Y
;Hide value of field Y/N: N
;SOL>
;Is field mandatory Y/N: N
;Is field fixed length Y/N: N
;Auto jump to next field Y/N: N
;Convert field to upper case Y/N: N
;Help message:
;Lowest value:
;Highest value:
;Field name: L2
;Type of field: CHR
;Length of field / Display length / Query length: 50 / 50 / 50
;Is this field in the base table Y/N: Y
;Is this field part of the primary key Y/N: Y
CERN, Geneva, Switzerland

; Allow field to be entered Y/N :  
e
; Allow field to be updated Y/N :  
Y
; Allow entry of query condition Y/N :  
Y
; Hide value of field Y/N :  
N
; SQL>

; Is field mandatory Y/N :  
N
; Is field fixed length Y/N :  
N
; Auto jump to next field Y/N :  
N
; Convert field to upper case Y/N :  
N
; Help message :  
Enter value for : LINE4
; Lowest value :  
; Highest value :  

; Field name :  
LS
; Type of field :  
CHAR
; Length of field / Display length / Query length :  
5B / 5B / 5B
; Is this field in the base table Y/N :  
Y
; Is this field part of the primary key Y/N :  
N
; Default value :  

; Page :  
1
; Line :  
20
; Column :  
17
; Prompt :  

; Allow field to be entered Y/N :  
e
; Allow field to be updated Y/N :  
Y
; Allow entry of query condition Y/N :  
Y
; Hide value of field Y/N :  
N
; SQL>

; Is field mandatory Y/N :  
N
; Is field fixed length Y/N :  
N
; Auto jump to next field Y/N :  
N
; Convert field to upper case Y/N :  
N
; Help message :  
Enter value for : LINES
; Lowest value :  
; Highest value :  

; Field name :  
Block name / Description :  

<table>
<thead>
<tr>
<th>COUNTRY</th>
<th>ADOR_NUM</th>
<th>INST_NUM</th>
<th>ADDRNUM</th>
<th>TOWN</th>
<th>STARTED</th>
<th>UPDATED</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINE1</td>
<td>LINE2</td>
<td>LINE3</td>
<td>LINE4</td>
<td>LINE5</td>
<td>LINE6</td>
<td>LINE7</td>
</tr>
<tr>
<td>TEL1</td>
<td>TEL2</td>
<td>TEL3</td>
<td>TEL1</td>
<td>TEL2</td>
<td>TEL3</td>
<td>TEL1</td>
</tr>
<tr>
<td>FITS</td>
<td>TELEX</td>
<td>CABLE1</td>
<td>TFX1</td>
<td>TFX2</td>
<td>NODE1</td>
<td>NODE2</td>
</tr>
</tbody>
</table>

PAGE 2
--------- FORM FOR ACCELERATORS OF INSTITUTES ---------
COUNTRY : ADOR_NUM : RESP : STARTED : UPDATED :

PAGE 5
Description of the accelerators

Research programs and types of HEP institutes

Country: Address number:

Research type: (X = Yes)

Theoretical High-Energy Physics
Experimental High-Energy Physics
Activity related to High-Energy Physics:

Country: Address number:

Research program:
Appendix B

Listing of GLETI RPT
REM REPORT-FILE: GLETI.RPT
REM O.L. VIERSMA TH/815 JUNE '79 SOLRPT ORACLE
REM
REM Generate the letters with the commands:
REM (1) RPT GLETI.RPT SNAP.BPF VIERSMA/passw
REM (2) RPT SNAP1 OUT1.DATA
REM (3) SGML801
REM
REM REPORT PROGRAM FOR GENERATED HEP LETTER
REM
REM
REM This program creates a computer-generated letter for all HEP
REM Institutes. Information from the database: ORACLE is combined with
REM SGML tags to form a SGML file. A REPORT statement is used to drive
REM the generation of multiple SGML files, each with a different
REM Institute address.
REM
REM DECLARE VARIABLES
REM
REM DATABASE ORACLE
DECL L1 L2 L3 L4 L5 L6
DECLARE L2 a50
DECL L3 a50
DECL L4 a50
DECL L5 a50
DECL L6 a50
DECL name a40
DECL country a2
REM
REM DEFINE SELECT ADDRESS AND COUNTRY MACRO
REM
REM DEFINE seladdr
SELECT l1 l2 l3 l4 l5 l6 country
INTO l1 l2 l3 l4 l5 l6 country
FROM addr Hep
ORDER BY country, addr_num
REM
REM DEFINE PROCEDURAL MACRO
REM
REM DEFINE body
.EXECUTE selcou
"GOOD"
"TO"
"FROM"
"DATE"
"OPEN"
"TO"
"FROM"
"DATE"
"OPEN"

"The CERN Directory of High Energy Physics Institutes is in the process
of complete reorganization. Using a new software package, the records
for each Institute will be extended to include all kinds of
telecommunication addresses (telephone, telefax, telex, telegrams),
computer mail nodes, a short characterization of research and a short
description of main research programs.

We would therefore appreciate your cooperation in correcting and
completing the information of your Institute listed on the form enclosed.

With best wishes,
Yours Sincerely,"

S. Schwarz Scientific Information Service

"CLOSE"
"GOOD"

REM
REM PROCEDURE SECTION REPORT
REM
REM
REM
REM *************** DEFINE RPF TABLE ***************
REM
REM
REM 76
REM
REM
MT 1 0 76 #
MT 1
PAGE 0 68
REPORT seladdr body
STOP
Appendix C

Listing of GLET2 RPT
RENAME REPORT-FILE: GLEI2.RPT
RENAME G.J.WIERSMA; TH/SIS; JULY '88; SQL REPORT: ORACLE
RENAME

RENAME Generate the reports with the commands:
RENAME (1) RPF GLEI2.RPT SAMP.RPF WIERSMA/passu
RENAME (2) RPF SAMP OUT2.DAT
RENAME (3) SQL88
RENAME
RENAME
RENAME REPORT PROGRAM FOR GENERATED HEP REPORTS
RENAME
RENAME
RENAME This program creates a computer generated report for all HEP
RENAME Institutes. Information from the database: ORAL is combined with
RENAME SGM, tags to form a SGM file. REPORT statements are used to drive
RENAME the generation of multiple SGM files; each with information from a
RENAME different institute.
RENAME
RENAME
RENAME DECLARE VARIABLES
RENAME
RENAME
RENAME DATABASE oral
RENAME DECLARE country a2
RENAME DECLARE addr_num 9999
RENAME DECLARE L01 a50
RENAME DECLARE L02 a50
RENAME DECLARE L03 a50
RENAME DECLARE L04 a50
RENAME DECLARE L05 a50
RENAME DECLARE L06 a50
RENAME DECLARE name a40
RENAME DECLARE tel_a a40
RENAME DECLARE tel1 a40
RENAME DECLARE tel2 a40
RENAME DECLARE tel3 a40
RENAME DECLARE telex a40
RENAME DECLARE cable a40
RENAME DECLARE fax1 a40
RENAME DECLARE fax2 a40
RENAME DECLARE node a40
RENAME DECLARE node2 a40
RENAME DECLARE tel_a a40
RENAME DECLARE L11 a50
RENAME DECLARE L12 a50
RENAME DECLARE L13 a50
RENAME DECLARE L14 a50
RENAME DECLARE L15 a50
RENAME DECLARE thpp a1
RENAME DECLARE thpp a1
RENAME DECLARE arthpp a1
RENAME DECLARE description a40
RENAME
RENAME
RENAME DEFINE SELECT MACROS
RENAME
RENAME
RENAME DEFINE seladdr
RENAME SELECT line1, line2, line3, line4, line5, line6, country,
RENAME addr_num, tel_a, tel1, tel2, tel3, telex, cable, fax1, fax2,
RENAME node1, node2, node3
RENAME INTO L01, L02, L03, L04, L05, L06, country, addr_num, tel_a, tel1, tel2,
RENAME tel3, telex, cable, fax1, fax2, node1, node2, node3
RENAME FROM addr
RENAME ORDER BY country, addr_num
RENAME
RENAME
RENAME DEFINE select
RENAME SELECT name
RENAME INTO name
RENAME FROM inste.country, french
RENAME WHERE country = &country
RENAME
RENAME
RENAME DEFINE selresp
RENAME SELECT L1 L2, L3, L4, L5
RENAME INTO L1, L2, L3, L4, L5
RENAME FROM resp
RENAME WHERE country = &country and addr_num = &addr_num
RENAME
RENAME
RENAME DEFINE selrest
RENAME SELECT thepp, arthpp, thpp
RENAME INTO thepp, arthpp, thpp
RENAME FROM rest
RENAME WHERE country = &country and addr_num = &addr_num
RENAME
RENAME
RENAME DEFINE selacc
RENAME SELECT description
RENAME INTO description
RENAME FROM accelerator
RENAME WHERE country = &country and addr_num = &addr_num
RENAME ORDER BY seq_num
RENAME
RENAME
RENAME
RENAME DEFINE PROCEDURAL MACROS
RENAME
RENAME
RENAME DEFINE address
RENAME .PRINT L01
RENAME
.IF "L02 IS NULL" THEN label12 ELSE label12
 .Glabel12
 .Glabel12
 .PRINT L02

 .IF "L03 IS NULL" THEN label13 ELSE label13
 .Glabel13
 .Glabel13
 .PRINT L03

 .IF "L04 IS NULL" THEN label14 ELSE label14
 .Glabel14
 .Glabel14
 .PRINT L04

 .IF "L05 IS NULL" THEN label15 ELSE label15
 .Glabel15
 .Glabel15
 .PRINT L05

 .IF "L06 IS NULL" THEN label16 ELSE label16
 .Glabel16
 .Glabel16
 .PRINT L06

 .PRINT nano

DEFINE telecom1
 .PRINT tipo

 .IF "tipo IS NULL" THEN label11 ELSE label11
 .Glabel11
 .Glabel11
 .PRINT tipo

 .IF "tipo IS NULL" THEN label12 ELSE label12
 .Glabel12
 .Glabel12
 .PRINT tipo

 .IF "tipo IS NULL" THEN label13 ELSE label13
 .Glabel13
 .Glabel13
 .PRINT tipo

DEFINE telecom2
 .IF "tipo IS NULL" THEN label11 ELSE label11
 .Glabel11
 .Glabel11
 .PRINT tipo

DEFINE telecom3
 .IF "tipo IS NULL" THEN label11 ELSE label11
 .Glabel11
 .Glabel11
 .PRINT tipo

DEFINE telecom4
 .IF "tipo IS NULL" THEN label11 ELSE label11
 .Glabel11
 .Glabel11
 .PRINT tipo

DEFINE telecom5
 .IF "tipo IS NULL" THEN label11 ELSE label11
 .Glabel11
 .Glabel11
 .PRINT tipo

DEFINE telecom6
 .IF "tipo IS NULL" THEN label11 ELSE label11
 .Glabel11
 .Glabel11
 .PRINT tipo

-- 47 --
### DEFINE research

```assembly
IF "&depth IS NULL" THEN label1 ELSE label11
   .label11
   .PRINT "hiep
   \  \  \ Theoretical High-Energy physics"
   .GOTO label111
   .label111
   .IF "&depth IS NULL" THEN label12 ELSE label122
   .label122
   .PRINT "hiep
   \  \  \ Experimental High-Energy physics"
   .GOTO label1222
   .label1222
   .IF "&depth IS NULL" THEN label13 ELSE label133
   .label133
   .PRINT "hrhep
   \  \  \ Activity related to High-Energy physics"
   .GOTO label1333
   .label1333
```

### DEFINE programs

```assembly
IF "&all1 IS NULL" THEN label11 ELSE label111
   .label111
   .PRINT L11
   .IF "&all2 IS NULL" THEN label12 ELSE label122
   .label122
   .PRINT L12
   .IF "&all3 IS NULL" THEN label13 ELSE label133
   .label133
   .PRINT L13
   .IF "&all4 IS NULL" THEN label14 ELSE label144
   .label144
   .PRINT L14
   .IF "&all5 IS NULL" THEN label15 ELSE label155
   .label155
   .PRINT L15
```

### DEFINE add-code

```assembly
Address code:
   .PRINT country
   /.
   .PRINT addr_num
```

### DEFINE mainbody

```assembly
EXECUTE selcou<CODC>
  <ORTWON>NY</ORTWON>
  <HP2>Telephone number of Institute</HP2>
  <HP2>Address (include area code):</HP2>
```

---

CERN, Geneva, Switzerland
The Institute is engaged in (X = Yes):

---

**Procedure Section**

---

**Define RPF Table**

---

STOP
Appendix D

The MENU EXEC program (REXX)
/*PROCEDURE POUR L'APPLICATION - CENLIB */
DAY(DATEU) : /*EXTRAIT LE NOM DU JOUR EN ANGLAIS*/
M=INDEX('MONDAY,TUESDAY,WEDNESDAY,THURSDAY,FRIDAY'); /*IDENTIFIE LE JOUR SUR 2 CHAR*/
N=INDEX('SUNDAY,MONDAY,TUESDAY,WEDNESDAY,THURSDAY,FRIDAY'); /*IDENTIFIE LE JOUR SUR 2 CHAR*/
COMMON:
CLOSEM
SAY ' '* CENTRAL LIBRARY '
SAY ' ' ACCESS TO ORACLE PROCEDURES '
SAY ' ' JOUR ' DATE(E)
SAY ' ' ' A --- UPDATES OF PUBLICATIONS LIST (LM,PR,BK)
SAY ' ' ' B --- UPDATES OF EXCHANGE LIST (EXCH)
SAY ' ' ' C --- UPDATES OF HEP ADDRESS & ACCELERATOR LISTS
SAY ' ' ' D --- UPDATES OF DICTIONARIES
SAY ' ' ' E --- PRINTING FACILITIES
SAY ' ' F --- ERASE OBSOLETE PUBLICATIONS ADDRESSES AND RENUMBER
SAY ' ' G --- SOME UTILITIES
SAY ' ' Z --- MENU EXIT
SAY ' ' PARSE UPPER PULL REPSONS
IF INDEX('ABCDEFGHIJKLMNOPQRSTUVWXYZ',REPSONS)<8 THEN SAY 'CARACTERE NON PREVU AU MENU' ELSE DO
IF REPSONS='A' THEN CALL LMFORM
IF REPSONS='B' THEN CALL EXCHFORM
IF REPSONS='C' THEN CALL RENUMFORM
IF REPSONS='D' THEN CALL DICTION
IF REPSONS='E' THEN CALL MENU2
IF REPSONS='F' THEN CALL RENUM
IF REPSONS='G' THEN CALL MENU3
IF REPSONS='Z' THEN SIGNAL EXIT
END
SIGNAL MENU
LMFORM:
/* SAISIE DES DONNEES DE LM*/
'10P LA CENLIB/BOOK - C IBM3278U'
RETURN
EXCHFORM:
/* SAISIE DES DONNEES DE EXCH*/
'10P EXCH CENLIB/BOOK - C IBM3278U'
RETURN
RENUMFORM:
/* SAISIE DES DONNEES DE RENUM*/
'10P RENUMFORM UWSMAIL/WSRSOW - C IBM3278U'
RETURN
DICTION:
/* GESTION DES DICTIONNAIRES*/
'10P DICTION CENLIB/BOOK - C IBM3278U'
RETURN
RENUM:
/* ERASE OBSOLETE PUBLICATIONS ADDRESSES AND RENUMBER */
SAY ' Procedure to erase obsolete LM addresses and renumber the ID of '
SAY ' LMSOR and LIMPERS tables '
SAY ' Please confirm: Y/N ?'
SAY ' ' PARSE UPPER PULL REPSONS
IF REPSONS='Y' THEN RETURN
RENUM
RETURN
MENU2:
CLOSEM
SAY ' ' PRINTING FACILITIES '
SAY ' ' A --- LM,PR or BK in 2 parts; printed at the R105 '
SAY ' ' B --- LM,PR or BK sorted by id; printed on vertical A4 (bidg 513) '
SAY ' ' C --- LM,PR or BK sorted by country; town; printed as above '
SAY ' ' D --- LM,PR or BK with list of country as selection criteria (R105) '
SAY ' ' E --- All EXCH sorted by country; town; printed as above '
SAY ' ' F --- All EXCH sorted by country; town; printed as above '
SAY ' ' G --- EXCH with selection criteria (R105) '
SAY ' ' H --- Proof-reading of HEP address list (3812 bidg 53) '
SAY ' ' I --- Listing for White Book '
SAY ' ' J --- Nailing list of HEP addresses (R105) '
SAY ' ' K --- List of countries as used in HEP addresses (R105) '
SAY ' ' L --- List of towns as used in HEP addresses (R105) '
SAY ' ' D --- Front page of HEP addresses letters (3812 bidg 53) '
SAY ' ' F --- Second page of HEP addresses letters (3812 bidg 53) '
SAY ' ' Z --- MAIN MENU '
PARSE UPPER PULL REPSONS
IF INDEX('ABCDEFGHIJKLMNOPQRSTUVWXYZ',REPSONS)<8 THEN SAY 'CARACTERE NON PREVU AU MENU' ELSE DO
IF REPSONS='A' THEN CALL LMLM
IF REPSONS='B' THEN CALL LMSOR
IF REPSONS='C' THEN CALL LMSOR
IF REPSONS='D' THEN CALL LMSOR
IF REPSONS='E' THEN CALL EXCH
IF REPSONS='F' THEN CALL EXCHI
IF REPSONS='G' THEN CALL EXCHI
IF REPSONS='H' THEN CALL PROD
IF REPSONS='I' THEN CALL WHI
IF REPSONS='J' THEN CALL LISTHEP
IF REPSN-"K" THEN CALL PAYSHEP
IF REPSN-"L" THEN CALL VILHEP
IF REPSN-"D" THEN CALL LETTER1
IF REPSN-"P" THEN CALL LETTER2
IF REPSN-"Z" THEN SIGNAL MENU1
END
SIGNAL MENU2
LMLA1:
/\ IMPRESSION DE LM/1
SAY 'LM list will be produced ! Please confirm. Y/N ?'
SAY ''
PARSE UPPER PULL REPSN
IF REPSN-"Y" THEN CALL MENU2
CALL SUBJECT
SAY 'Air mail or surface mail please ? Enter A or S'
SAY ''
PARSE UPPER PULL PARAM
'FILEDEF 2I DISK LMLA LISTING AI (LRECL 137)
LMLA1 PARAM SUBJECT
 XPARM LSMLA LISTING AI (OE 5B DIST X2)
'ERASE LMLA LISTING AI'
RETURN
LSMLA1:
CALL SUBJECT
SAY 'List of country codes requested please ? '
SAY ' eg. MEMBER_STATES, IRR, as stated in COUNTRY_LISTS dictionary'
SAY ''
PARSE UPPER PULL PARAM1
SAY 'Your report name please ? default is: A GIVEN LIST OF COUNTRIES'
PARSE UPPER PULL PARAM2
IF PARAM2-"" THEN PARAM2="A GIVEN LIST OF COUNTRIES"
SAY 'Your list of selection criteria : '
PARSE UPPER PULL PARAM2
SAY ''
PARSE UPPER PULL REPSN
IF REPSN-"OK" THEN SIGNAL MENU2
'FILEDEF 2I DISK LSMLA LISTING'
'LMLA1 LSMLA SUBJECT PARAM2
 XPARM LSMLA LISTING AI (OE 5B DIST X2)
'ERASE LSMLA LISTING AI'
RETURN
LSMLA2:
SAY 'All LM list will be produced ! Please confirm. Y/N ?'
SAY ''
PARSE UPPER PULL REPSN
IF REPSN-"Y" THEN SIGNAL MENU2
CALL SUBJECT
'FILEDEF 2I DISK LSMLA LISTING'
'LMLA2 SUBJECT
 XPARM LSMLA LISTING AI (PRINTER 8700 CLASS C DIST X2)
'ERASE LSMLA LISTING AI'
RETURN
LSMLA3:
SAY 'All LM list will be produced ! Please confirm. Y/N ?'
SAY ''
PARSE UPPER PULL REPSN
IF REPSN-"Y" THEN SIGNAL MENU2
CALL SUBJECT
'FILEDEF 2I DISK LSMLA LISTING'
'LMLA3 SUBJECT
 XPARM LSMLA LISTING AI (PRINTER 8700 CLASS C DIST X2)
'ERASE LSMLA LISTING AI'
RETURN
SUBJECT:
SAY 'Which subject do you want ? (LM, FR or BF)'
SAY ''
PARSE UPPER PULL SUBJECT
IF SUBJECT-"LM" THEN RETURN
IF SUBJECT-"FR" THEN RETURN
IF SUBJECT-"BF" THEN RETURN
SIGNAL SUBJECT
EXCHID:
/\ IMPRESSION DE EXCH. triees par ID/1
SAY 'All EXCH list will be produced ! Please confirm. Y/N ?'
SAY ''
PARSE UPPER PULL REPSN
IF REPSN-"Y" THEN CALL MENU2
'FILEDEF 2I DISK EXCHID LISTING AI'
EXCHID
 XPARM EXCHID LISTING AI (PRINTER 8700 CLASS C DIST X2)
'ERASE EXCHID LISTING AI'
RETURN
EXCHPSY:
/\ IMPRESSION DE EXCH. triees par PAYS-VILLE/1
SAY 'All EXCH list will be produced ! Please confirm. Y/N ?'
SAY ''
PARSE UPPER PULL REPSN
IF REPSN-"Y" THEN CALL MENU2
'FILEDEF 2I DISK EXCHPSY LISTING AI'
EXCHPSY
 XPARM EXCHPSY LISTING AI (PRINTER 8700 CLASS C DIST X2)
'ERASE EXCHPSY LISTING AI'
RETURN
EXCHSEL:
/\ PRINTING OF EXCH WITH SELECTION CRITERIA/1
SAY 'Paper or microfilm : P or M ?'
SAY ''
PARSE UPPER PULL PARAM1
CERN, Geneva, Switzerland

SAY 'List of subjects requested please?  eg. YYYY'
SAY 'List of country codes requested please?'
SAY 'eg. ALL-RMEMBER,STATES, RPP, as stated in COUNTRY_LISTS dictionary'
SAY 'Your report name please?  default is CERN BN-88'
SAY 'Your list of selection criteria:  "PARAM1"
SAY 'PARAM2'
SAY 'PARAM3'
SAY 'PARAM4'
SAY 'Enter OK or press RETURN to restart'

FILEDEF 21 DISK EXCEL LISTING
'EXCEL 'PARAM1 PARAM2 PARAM3 PARAM4
FILEDEF 21 DISK EXCEL LISTING
'EXCEL 'PARAM1 PARAM2 PARAM4
FILEDEF 21 DISK EXCEL LISTING

Do you want to cut your file into two parts?  If > 4996 lines,  Y or N?

FILEDEF 21 DISK MAPPRINT LIST A1(RECL 137)

FILEDEF 21 DISK WHITEBOOK SCRIPT A1
WHITEBK
'RESCRIBE WHITEBOOK SCRIPT A1 (APA6678 BAT 513 3 7)'
'RESCRIBE WHITEBOOK SCRIPT A1'
'RESCRIBE WHITEBOOK PEPRI NT A1'
'RESCRIBE WHITEBOOK LISTAMP A1'

FILEDEF 21 DISK ListHep LIST A1(RECL 137)
'LISTHEP'
'RESCRIBE LISTHEP LISTING A1 (DE SB DIST X2)
'RESCRIBE LISTHEP LISTING A1'

FILEDEF 21 DISK PAYSHEP LIST A1(RECL 137)
'PAYSHEP'
'RESCRIBE PAYSHEP LISTING A1 (DE SB DIST X2)
'RESCRIBE PAYSHEP LISTING A1'

FILEDEF 21 DISK VILLEHEP LIST A1(RECL 137)
'VILLEHEP'
'RESCRIBE VILLEHEP LISTING A1 (DE SB DIST X2)
'RESCRIBE VILLEHEP LISTING A1'

LETTER1

/* PRODUCTION OF LETTERS TO HEP INSTITUTES (FRONTPAGE) */
'RPT GLE1,RPT SMP1,RPP WIERSM,HTSHOU'
'RPP SMP1 OUTL.DAT'
'SGM.882'
SAY ' Press RETURN:'
PARSE PULL REPONS
RETURN
LETTER2:
/* PRODUCTION OF LETTERS TO HEP INSTITUTES (SECOND PAGE) */
'REP GLIT2.RPT SAMP2.RPT WIESSMA/HISARW'
'REP SAMP2 OUT2.DUNI'
'SGM.882'
SAY ' Press RETURN:'
PARSE PULL REPONS
RETURN
MENU3:
CLRSCRN
SAY ' ACCESS TO SOME UTILITIES'
SAY ' A --> Simple calculator'
SAY ' B --> Simple accumulator'
SAY ' Z --> MAIN MENU'
SAY ' PARSER UPPER PULL REPONS'
IF INDEX('aabZ',REPONS)=0 THEN SAY 'CARACTERE NON PREVU AU MENU'
ELSE DO
  IF REPONS='A' THEN CALL calc
  IF REPONS='B' THEN CALL accu
  IF REPONS='Z' THEN SIGNAL MENU1
END
SIGNAL MENU3
CALC:
CALCUL
RETURN
ACCU:
ADDITION
RETURN
EXIT:
'CP SET MSG ON'
CLRSCRN
EXIT
Appendix E

Listing of SGML801 EXEC

/* how to sgml about 800 letters out of one file */
address command
endline=
EOF=0
SYS "About 800 pages will be produced! Please confirm: Y/N ?"
SY"Y"
PARSE UPPER PULL REPS
IF REPS = "Y" THEN EXIT
SYS "Are you sure ? (Y/N)"
PARSE UPPER PULL REPS
IF REPS = "Y" THEN EXIT
"CP SET MSG OFF"
CLS
SYS "All letters will be printed on the 3812 printer in building 53"
SYS "This printing job can take a long time!"
do while EOF=0
queue "SET AUTOSAVE OFF"
queue "<!DOCTYPE SGMLetl>
""EXECDG DISK OUTI DATA 6 line ""STEM TLINE. LOCATE /<GDC>"" FINIS"
if rc=0 then parse var tline.2 . startline .
else alert.
""EXECDG DISK OUTI DATA 6 startline ""STEM TLINE. LOCATE $<GDC>$ FIN
if rc=0 then parse var tline.2 . endline .
else next
begin
endline+endline=startline+1
""EXECDG totLine ""DISK OUTI DATA 6 startline ""STEM R. FINIS"
do x=1 to totline
queue "I "fil.x
end
queue "FILE"
""EDIT LETTER TEMP 41"
set cmstype ht
""EXEC SGML LETTER TEMP 41 (3812 NAME esp4x NORTINE PRINT NEW"
set cmstype rt
""ERASE LETTER TEMP 41""
""ERASE LETTER PENPRINT R1"
""ERASE LETTER LIST3APP R1"
""ERASE LETTER 13620 R1"
end
""ERASE SIMPL RFP 41"
""ERASE OUTI DATA 41"
CLS
SYS "All letters are sent to the printer!"
SYS "Loss from time to time if there is enough paper in the printer!"
Appendix F

Listing of SGML802 EXEC

/* how to sgml about 800 reports out of one file */
address command
endline1
$off8
SRY"About 800 pages will be produced ! Please confirm, Y/N ?"
SRY"
PARSE UPPER PULL REPONS
IF REPONS ~"Y" THEN EXIT
SRY"ARE YOU SURE ? (Y/N)"
PARSE UPPER PULL REPONS
IF REPONS ~"Y" THEN EXIT
"CP SET MSG OFF"
SRY"--------------------------------------------------------------------------"
SRY"801i reports will be printed on the 3812 printer in building 53"
SRY"This printing job can take a long time!"
do while $off8
queue "set autosave off"
queue "i <IDOCTYPE SGMLclet>"
"EXEC10 = DISKR OUT2 DATA a * endline "STEM TLINE. LOCATE /<GDOC>/ FINIS"
if rc# then parse var tline.2 , startline .
else end=1
"EXEC10 = DISKR OUT2 DATA a * startline "STEM TLINE. LOCATE $</GDOC>$ FIN
if rc# then parse var tline.2 , endline .
else nap
tolines=endline-startlines!
"EXEC10" tololine "DISKR OUT2 DATA a * startline "STEM LI. FINIS"
do $x1 to tolinline
queue "i "fi.x
and
queue "FILE"
"XEDIT REPORT TEMP R1"
sctstate ht
"EXEC SGML REPORT TEMP R1(I3812 NAME ep84e NODROUSE PRINT NEW"
sctstate ri
"ERASE REPORT TEMP R1"
"ERASE REPORT FEPAPP R1"
"ERASE REPORT LISTAPP R1"
"ERASE REPORT I3820 R1"
and
"ERASE SMP2 PF R1"
"ERASE OUT2 DATA R1"
CLOSER
SRY"801i reports are send to the printer"
SRY"Look from time to time if there is enough paper in the printer!"
Appendix G

Proof-Reading List Program
PROGRAM HEDADD
C APPLICATION BIBLIOTHEQUE CENTRALE
C IMPRESSION DE LA LISTE HEP ADDRESS
C EXEC SOL INCLUDE SOLCA.FOR
C EXEC SOL BEGIN DECLARE SECTION
C INTEGER4 ADDR
CHARACTER*8 LI1,LI2,LI3,LI4,LS1,LS2,LS3,LS4,LS5
CHARACTER*4 TR,T1,T2,T3,TX,CA,TF1,TF2,F1,N01,N02,DESCR,NAM
CHARACTER*4 DB
CHARACTER*4 ID
CHARACTER*4 RESP,ADDR,RAHAY
CHARACTER*2 PHYS
CHARACTER*1 TH,EH,AR
C EXEC SOL END DECLARE SECTION
C INTEGER4 OUT,TAB,TAB2,TAB3,NL,(2),ADDR
CHARACTER*132 LINE(34)
CHARACTER*11 ACCEL
CHARACTER*1 BLK,ONE
C PARAMETER(OUT=21,BLK=' ',ONE='I')
C DB = 'UIERM/HSDOU'
RAHAY = 'RAHAY'
ACCEL = 'ACCELERATOR'
C PRINT * , 'Programme running, please be patient !'
C EXEC SOL CONNECT :DB
C EXEC SOL WHENEVER SOLERROR GOTO 10000
C OPEN (UNIT=OUT,FMT='FORMATTED',STATUS='NEW')
C EXEC SOL DECLARE SELI CURSOR FOR
1 SELECT ADDR,N01,N02,LINE1,LINE2,LINE3,LINE4,LINE5,LINE6,
1 LINE7,TR,EH,LI1,LI2,LI3,LI4,LS1,LS2,LS3,LS4,LS5,TEL,TEL2,TEL3,TEL4,
1 TELEX,CABLE,TF1,TF2,FTF1,FTF2,F1,N01,N02,DESCR,NAM,PHYS
1 FROM ADADDR WHERE RESP=RAHAY AND COUNTRY=PHYS
1 ORDER BY IDN
C EXEC SOL DECLARE SEL2 CURSOR FOR
2 SELECT COUNTRY FROM INST,COUNTRY ENGLISH
2 WHERE COUNTRY IN (SELECT COUNTRY FROM ADADDR GROUP BY COUNTRY)
2 ORDER BY NAME
C EXEC SOL DECLARE SEL3 CURSOR FOR
3 SELECT DESCRIPTION FROM ACCELERATOR
3 WHERE COUNTRY=PHYS AND ADDR=ADDR
C EXEC SOL DECLARE SEL4 CURSOR FOR
4 SELECT TEL,EH,TEL,TEL FROM RESPROG
4 WHERE COUNTRY=PHYS AND ADDR=ADDR
C EXEC SOL DECLARE SEL5 CURSOR FOR
5 SELECT LI1,LI2,LI3,LS4,LS5 FROM RESPROG
5 WHERE COUNTRY=PHYS AND ADDR=ADDR
C EXEC SOL DECLARE SEL6 CURSOR FOR
6 SELECT NAME FROM INST,COUNTRY ENGLISH
6 WHERE COUNTRY=PHYS
C NLINE = 34
DO 10 I=1,NLINE
10 LINE(I) = BLK
LINE(I+1) = ONE
TAB = -38
IPRO = 0
NL(I) = 0
NL(I+1) = 0
ID1 = 0
ADDR = 0
C EXEC SOL OPEN SEL2
EXEC SOL WHENEVER NOT FOUND GOTO 44
C EXEC SOL FETCH SEL2 INTO :PHYS
C EXEC SOL OPEN SEL1
EXEC SOL WHENEVER NOT FOUND GOTO 44
C CONTINUE
LI = BLK
L2 = BLK
L3 = BLK
L4 = BLK
L5 = BLK
L6 = BLK
L7 = BLK
NAM = BLK
TR = BLK
T1 = BLK
T2 = BLK
T3 = BLK
TX = BLK
CH = BLK
TF1 = BLK
TF2 = BLK
FT = BLK
N01 = BLK
N02 = BLK
LRI = BLK
LR2 = BLK
LR3 = BLK
LR4 = BLK
LR5 = BLK
TH = BLK
EH = BLK
AR = BLK

C EXEC SOL FETCH SEL1 INTO :ADDNR1, L1, L2, L3, L4, L5,
    (L6, L7, TH, TF1, TF2, TF3, TX, CH, TF1, TF2, FT)
    :N01, N02
C EXEC SOL OPEN SEL1
EXEC SOL WHENEVER NOT FOUND GO TO 222
EXEC SOL FETCH SEL2 INTO :TH, :EH, :AR
222 EXEC SOL OPEN SEL2
EXEC SOL WHENEVER NOT FOUND GO TO 223
EXEC SOL FETCH SEL3 INTO :LR1, LR2, LR3, LR4, LR5
C 223 EXEC SOL CLOSE SEL3
EXEC SOL CLOSE SEL5
C EXEC SOL OPEN SEL6
EXEC SOL FETCH SEL6 INTO :MAN
EXEC SOL CLOSE SEL6
C NADDR = NADDR + 1
S TAB = TAB + 76
IF (TAB .GT. 83) GO TO 2
TAB2 = TAB + 50
IDX = IDX + 1
C Integer ADDR is transferred into character type variable ADDR
WRITE (ADDR, 103) ADDR
LINE(16) (TAB1, TAB2) = 'PHYS//'
    ' ' // ADDR
LINE(27) (TAB1, TAB2) = 'L(1:50)
LINE(38) (TAB1, TAB2) = 'L(1:50)
LINE(49) (TAB1, TAB2) = 'L(1:50)
LINE(50) (TAB1, TAB2) = 'L(1:50)
LINE(61) (TAB1, TAB2) = 'L(1:50)
LINE(72) (TAB1, TAB2) = 'L(1:50)
LINE(83) (TAB1, TAB2) = 'L(1:50)
LINE(94) (TAB1, TAB2) = 'MAN(1:49)
TAB3 = TAB - 8
TAB4 = TAB2 - 18
LINE(18) (TAB3, TAB4) = 'TELA : //TA
LINE(19) (TAB3, TAB4) = 'TELI : //TI
LINE(20) (TAB3, TAB4) = 'TEL3 : //T3
LINE(21) (TAB3, TAB4) = 'TELEX : //TX
LINE(22) (TAB3, TAB4) = 'CABLE : //CA
LINE(23) (TAB3, TAB4) = 'TFRN1 : //TF1
LINE(24) (TAB3, TAB4) = 'TFRN2 : //TF2
LINE(25) (TAB3, TAB4) = 'FTS : //FT
LINE(26) (TAB3, TAB4) = 'MODE1 : //MO1
LINE(27) (TAB3, TAB4) = 'MODE2 : //MO2
LINE(28) (TAB3, TAB4) = 'THEP : //TH
LINE(29) (TAB3, TAB4) = 'THEP : //THE
LINE(30) (TAB3, TAB4) = 'BHEP : //BHEP
LINE(31) (TAB3, TAB4) = 'LR1(1:50)
LINE(32) (TAB3, TAB4) = 'LR2(1:50)
LINE(33) (TAB3, TAB4) = 'LR3(1:50)
LINE(34) (TAB3, TAB4) = 'LR4(1:50)
LINE(35) (TAB3, TAB4) = 'LR5(1:50)
C I = 30
EXEC SOL OPEN SEL3
EXEC SOL WHENEVER NOT FOUND GO TO 55
S DESCR = BLK
EXEC SOL FETCH SEL3 INTO :DESCR
IF (I = 1) LINE(1) (TAB3, TAB4) = DESCR
GO TO 6
C 55 CONTINUE
IF (I .GT. 30) LINE(30) (TAB3, TAB4) = ACCEL
DO 28 I = 1, NLINE
    IF (LINE(I) (TAB3, TAB2) .NE. BLK) GO TO 28
    NL(0) = NL(0) + 1
28 CONTINUE
L = NLINE + 1
DO 16 K = 1, L
    DO 14 J = 1, L
        IF (LINE(J) (TAB3, TAB2) .NE. BLK) GO TO 14
14 CONTINUE
16 CONTINUE
15 CONTINUE
14 CONTINUE
C DO 56 I = 2, L
    IF (LINE(I) (TAB3, TAB4) .EQ. ACCEL) LINE(I) (TAB3, TAB2) = BLK
56 CONTINUE
C
C
2 II = MIN(NL(1),NL(2))
ILEN = NLINE - II
IPAG = IPAG + ILEN + 1
IF ( IPAG .GT. 56 ) GO TO 3
DO 11 I=1,ILEN
WRITE (OUT,109) LINE(I)
11 CONTINUE
GO TO 12 +NL

12 LINE(I) = BLK
WRITE (OUT,109) LINE(I)
TOB = -58
IDX = 0
NL(1) = 0
NL(2) = 0
GO TO 3

3 CONTINUE
LINE(I)(1:I) = ONE
IPAG = 0
GO TO 2

C
4 CONTINUE

WRITE (OUT,182) WADDR
EXEC SOL COMMIT WORK RELEASE
STOP 'Fin normale de la liste HEP'

C
10000 PRINT 110, SOLCODE
PRINT 111, (SOLENCE(I),I=1,70)
STOP 'Fin anormale de la liste HEP'

C
100 FORMAT (6)
182 FORMAT (/////////50X,'** ',I4,' Addresses printed **')
183 FORMAT (4)
110 FORMAT (' ORACLE ERROR : CODE IS ',I5)
111 FORMAT (' MESSAGE IS : ',FNL)
END
Bibliography


