A SYSTEM 2000 Database for User Requirements Collection and Analysis according to DATAID Methodology

L. Abba - O. Signore

Rapporto interno C83-10

ASTUTE CONFERENCE Roma 4-6 May, 1983

Reparto basi di dati e sistemi informativi

```
 CCCCCC    NN     NN  UU    UU   CCCCCC    EEEEEEE
CCCCCCCC   NNN    NN  UU    UU  CCCCCCCC   EEEEEEE
CC    CC   NNNN   NN  UU    UU  CC    CC   EE
CC         NN NN  NN  UU    UU  CC         EEEEE
CC         NN  NNNN  UU    UU  CC         EEEEE
CC    CC   NN   NNN  UU    UU  CC    CC   EE
CCCCCCCC   NN    NN  UUUUUUU   CCCCCCCC   EEEEEEE
 CCCCCC    NN    NN   UUUUU     CCCCCC    EEEEEEE
```

CNR - Istituto CNUCE
via S. Maria 36
56100 Pisa

Tel. +39 50 593111
Telex 500371 CNUCE

Table of contents

Abstract

The DATAID-1 manual database design methodology is divided in four phases: user requirements collection and analysis; conceptual design; logical design; physical design.

The overall organization of the first phase and related documents are described.

Subsequently, the information contained in these documents are arranged in a conceptual schema, which is mapped into a hierarchical structure.

The resulting SYSTEM 2000 database is described, and future developments outlined.

In the appendixes, the actual SYSTEM 2000 implementation, and an example of session are shown.

Introduction

In the past few years, the database design has been realized as one of the most important research field, and a lot of effort has been put in this field.

At present, there is a general agreement on the fact that the database design process goes through four phases:

- requirement analysis
- conceptual design
- logical design
- physical design

A complete database design methodology subdivided in this way has been developed as part of the DATAID project, a 5-years project financed by the National Research Council (CNR) of Italy within the Progetto Finalizzato Informatica research project.

At present, 15 working groups are involved in the DATAID research activity (5 from Universities, 3 from CNR and 7 from private industries).

A recently published book [CERI83] collects the contributions from the participants to the project, and describes the results of the first 3 years of the DATAID project, using as case study an exercise that was suggested by IFIP Technical Committee 8 for a "Comparative Review of Information Systems Design Methodologies",in 1980 [OSVT82]. Reference will be made to this book in the rest of this paper.

The DATAID-1 methodology has been released as a manual methodology in 1981, and since then additional work has been done or planned in order to develop and integrate automated tools and to extend the methodology (to incorporate constraints treatment, distribution design etc.).

CNUCE is in charge of extending the methodology to logical and physical design of SYSTEM 2000 databases. In order to develop an integrated, automated tool, a first prototype of an User Requirements Collection and Analysis automated tool was implemented.

User Requirements Collection and Analysis Phase

In the DATAID-1 methodology, it is supposed that some important tasks, namely organization analysis, cost-benefit analysis etc. have already been performed before the actual start of the database design process.

Therefore, the methodology starts from a "contract", either formalized or not (which states what must be done), and ends up with a physical database and procedures which satisfy the contract requirements.

In the recent past, it has been recognized that the user requirements collection phase plays a fundamental role in the attainment of a proper implementation of a database system capable of fulfilling the users' expectations.

For this phase the DATAID-1 methodology provides a series of steps which, starting from the collection of users requirements expressed in natural language, produce a set of documents as input to subsequent phases.

We will briefly summarize the meaning of some keywords and the organization of the phase. For a more complete description, refer to [DeAD83].

"Organization environment" is the part of the organizational structure which is highly homogeneous for what concerns the objectives. This homogeneity implies a corresponding uniformity in the language used and in the functions performed.

"Organization functions" means a set of operations coordinated by conditions requested for the achievement of the goals of an organization environment.

"Operation" means a transformation of and/or inquiry on data of an organization environment.

"Event" is a change of conditions, from pre-conditions to postconditions.

"Entity" is an object of the real world; for application reasons, it is interesting to represent this object in such a way as to be identifiable with respect to other objects of the real world and to represent a certain number of significant properties of this object with logical relationships with other entities.

"Attribute" is a characteristic of an entity, or of an association, interesting to the application.

"Association" is a logical relationship between entities representing facts of the real world, interesting to the application.

It has to be noted that every step of the phase is executed separately for every organization environment.

The users' requirements expressed in natural language constitute the input to the phase.

The following documents are produced:

1. Lists of Function/environment
2. Lists of user/functions
3. Data sentence forms
4. Operation sentence forms
5. Event sentence forms
6. Constraints sentence forms
7. Data glossary
8. Operations glossary
9. Event glossary
10. Relationship glossary
11. Event specification forms
12. Event graphs
13. Skeleton Schema

At first, the environments involved in every function are catalogued in document 1, and the organization function required by every user for the completion of his job within his environment. At a second stage, starting with a description in natural language of the organization operations, a linguistic filter is used to eliminate ambiguities and inaccuracies, and to obtain a division of the sentences into 4 types:

- data sentences
- operation sentences
- event sentences
- constraints sentences;

and at the end the respective forms 3, 4, 5 and 6 are catalogued.

Every datum contained in the sentence is entered together with its characteristics into the data glossary. It has to be noted that a preliminary classification of the data as entity or attribute is already required at this level (this classification will subsequently undergo a control in the conceptual design phase).

Every operation sentence describes one or more operations, which in turn involves one or more data. All useful operations and information characterizing the data are catalogued in the operation glossary provided with a field dedicated to the data concerned. By operation, it is meant to be an action which modifies the state of the data of the system.

Every event sentence describes an event in the sense of

a description of an organizational activity inherent to an organization environment. In the same way also the events are catalogued into the respective glossary.

Every event is described in terms of its own preconditions and of the actions modifying these conditions. The event specification form describes this modification in terms of the events which generate the necessary conditions for a verification of one or more operations.

Finally, all the relationships between interesting data are listed in the relationship Glossary.

Once the glossaries are completed, they are checked for consistence in order to evaluate the completeness of the respective descriptions. Further analyses are done to identify eventual cases of synonymy between data and eventual hierarchical relationships between the concepts expressed.

The skeleton schema constitutes a first nucleus of the conceptual schema of the whole organization. This is designed according to the information organized in the data and operation glossaries, assigning to each datum a "type" (entity, attribute or association). This attempt of classification is subsequently tested for consistence and coherence. In cases where specifications supplied by the users are considered scarcely trustworthy, the methodology provides an alternative step: the skeleton schema is constructed after analyzing the data relationships (relationship glossary) and testing their coherence, the data being classfied by the analysis of the mathematical relations linking the data.

The event graphs are the representation of the network of conditions and actions regulating the organization activities. The formalism adopted for their representation is the Petri Net [PETRI80].

The output documents of the phase will be the source of consultance and basis for further phases envisaged by the methodology.

Figure 1 illustrates the structure of this phase.

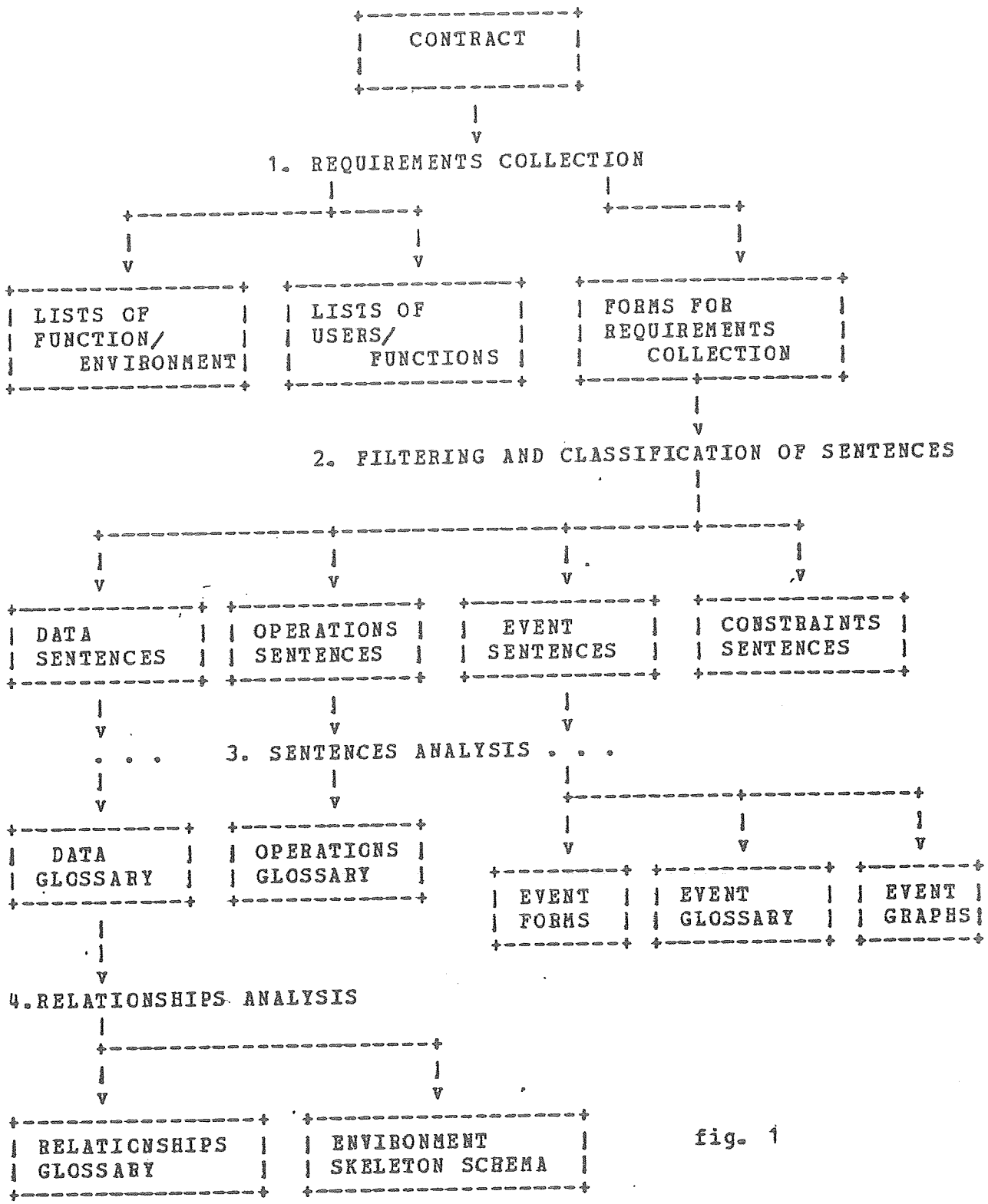

CONTRACT
1. REQUIREMENTS COLLECTION
LISTS OF FUNCTION/ ENVIRONMENT
LISTS OF USERS/ FUNCTIONS
FORMS FOR REQUIREMENTS COLLECTION
2. FILTERING AND CLASSIFICATION OF SENTENCES
DATA SENTENCES
OPERATIONS SENTENCES
EVENT SENTENCES
CONSTRAINTS SENTENCES
3. SENTENCES ANALYSIS . . .
DATA GLOSSARY
OPERATIONS GLOSSARY
EVENT FORMS
EVENT GLOSSARY
EVENT GRAPHS
4. RELATIONSHIPS ANALYSIS
RELATIONSHIPS GLOSSARY
ENVIRONMENT SKELETON SCHEMA

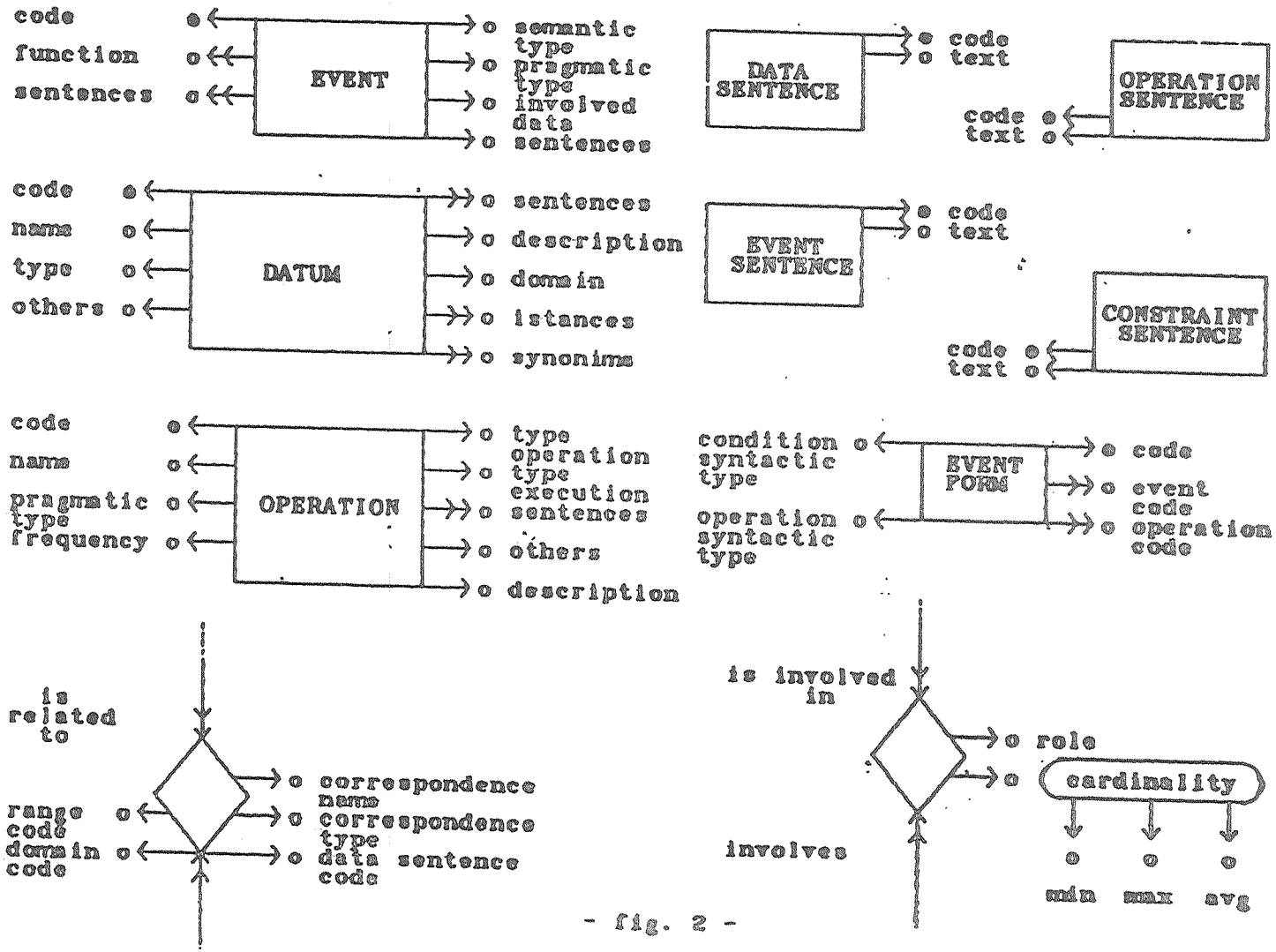
fig. 1

The conceptual schema

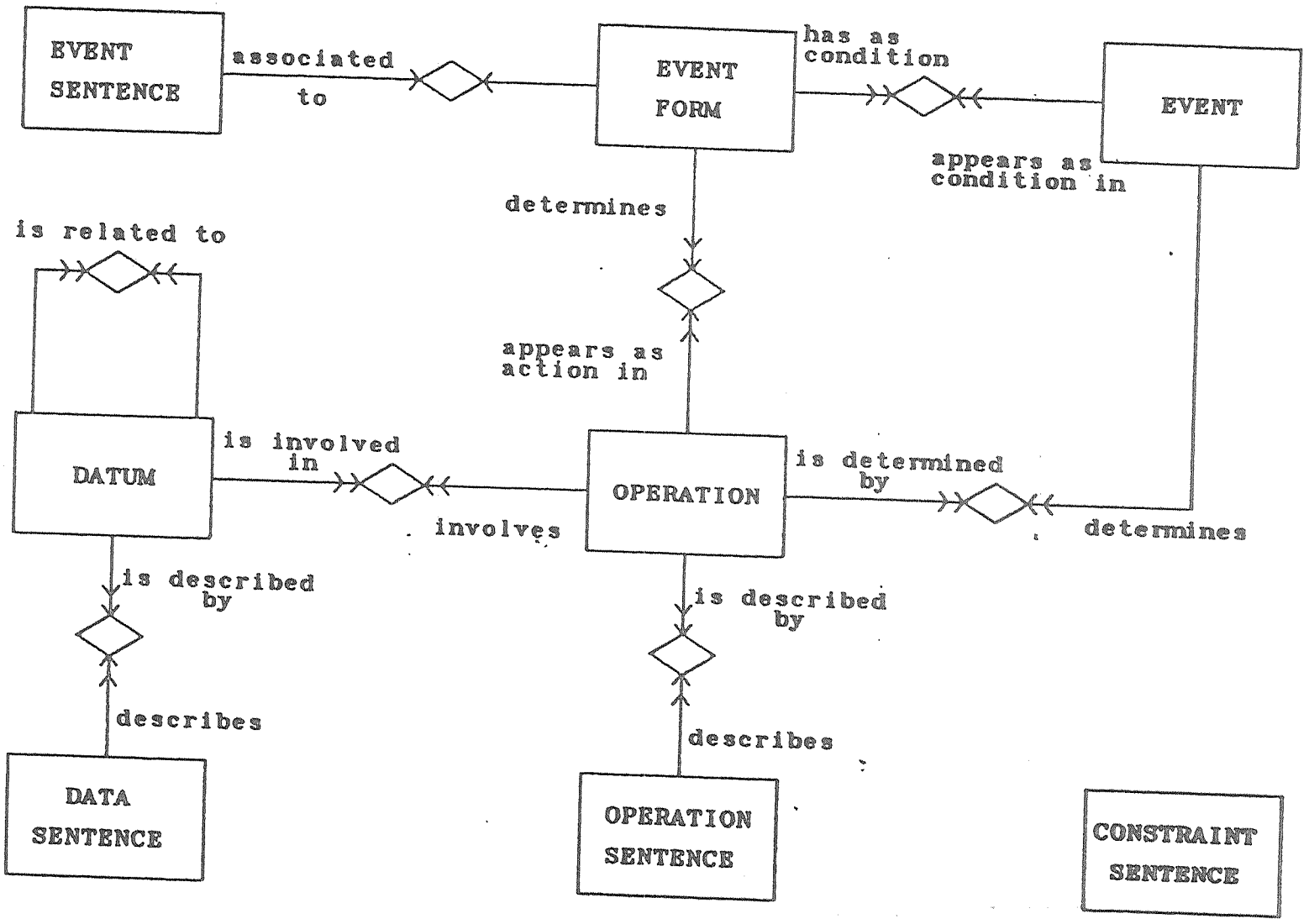
To design a database capable of storing information related to phase 1 of the DATAID methodology, 8 interesting entities have been identified:

- data sentence
- operations sentence
- event sentence
- constraints sentence
- event specification form
- datum
- operation
- event.

Figures 2 and 3 show the relative conceptual schema.
The enriched Entity Relationship model has been used [CHEN76], [BuCe83].


code
function
sentences
EVENT
semantic type
pragmatic type
involved data
sentences
DATA SENTENCE
code
text
OPERATION SENTENCE
code
text
code
name
type
others
DATUM
sentences
description
domain
instances
synonims
EVENT SENTENCE
code
text
CONSTRAINT SENTENCE
code
text
code
name
pragmatic type
frequency
OPERATION
type operation
type execution
sentences
others
description
condition syntactic type
operation syntactic type
EVENT FORM
code
event code
operation code
is related to
range code
domain code
correspondence name
correspondence type
data sentence code
is involved in
involves
role
cardinality
min
max
avg

- fig. 2 -


EVENT SENTENCE
associated to
EVENT FORM
has as condition
EVENT
appears as condition in
determines
is related to
appears as action in
DATUM
is involved in
involves
OPERATION
is determined by
determines
is described by
is described by
describes
describes
DATA SENTENCE
OPERATION SENTENCE
CONSTRAINT SENTENCE

- fig. 3 -

The implementation

One of the fundamental problems encountered in describing a prototype was the definition of the developmental environment of the software tools to be adopted.

It is believed to be advisable to use a DBMS as the support since this appears to be an indispensable tool for the representation of many complex relationships between the entities. Besides, the use of a DBMS offers the advantages of having concurrent access to the data, and guaranteeing the recovery and in general allowing interactivity. These aspects are however not particularly important in the collection of requirements.

Since an automation tool is being designed at CNUCE for the logical and physical design of databases - SYSTEM 2000, we have chosen to use this DBMS for the design of a homogeneous tool.

It appeared so obvious that, regardless of the type of DBMS chosen, it would have been necessary to develop procedural applications for data entry, whereas in general the query language of the DMBS is acceptable for output.

However, in our case of a pilot design intended, at least for the moment, to test the quality of the conceptual schema and the logical schema, we have decided to use directly the query language of SYSTEM 2000 [LSM/SCF] which guarantees a notable speed and flexibility during implementation.

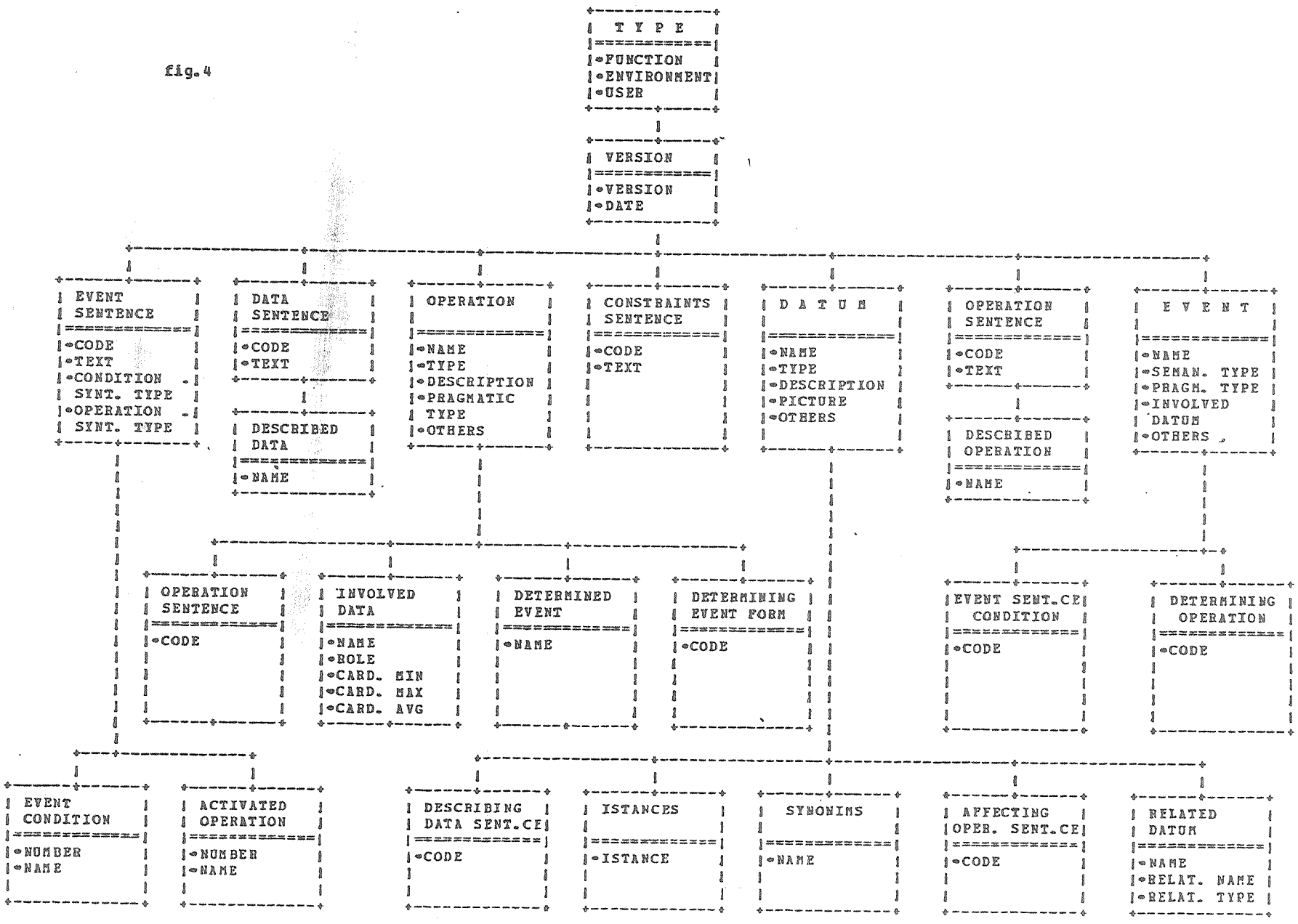
During the translation of the conceptual schema into a hierarchical logical schema, the N:M relationships are solved by introducing an adequate schema record containing the identifiers of the entities.

This solution enables the utilization of LINK commands [LSM-COB] in future procedural developments.

As to the physical definition on SYSTEM 2000 please read the appendix 1.

Figure 4 illustrates the above mentioned logical schema.

fig.4


TYPE
•FUNCTION
•ENVIRONMENT
•USER
VERSION
•VERSION
•DATE
EVENT SENTENCE
•CODE
•TEXT
•CONDITION SYNT. TYPE
•OPERATION SYNT. TYPE
DATA SENTENCE
•CODE
•TEXT
DESCRIBED DATA
•NAME
OPERATION
•NAME
•TYPE
•DESCRIPTION
•PRAGMATIC TYPE
•OTHERS
CONSTRAINTS SENTENCE
•CODE
•TEXT
DATUM
•NAME
•TYPE
•DESCRIPTION
•PICTURE
•OTHERS
OPERATION SENTENCE
•CODE
•TEXT
DESCRIBED OPERATION
•NAME
EVENT
•NAME
•SEMAN. TYPE
•PRAGM. TYPE
•INVOLVED DATUM
•OTHERS
OPERATION SENTENCE
•CODE
INVOLVED DATA
•NAME
•ROLE
•CARD. MIN
•CARD. MAX
•CARD. AVG
DETERMINED EVENT
•NAME
DETERMINING EVENT FORM
•CODE
EVENT SENT.CE CONDITION
•CODE
DETERMINING OPERATION
•CODE
EVENT CONDITION
•NUMBER
•NAME
ACTIVATED OPERATION
•NUMBER
•NAME
DESCRIBING DATA SENT.CE
•CODE
ISTANCES
•ISTANCE
SYNONIMS
•NAME
AFFECTING OPER. SENT.CE
•CODE
RELATED DATUM
•NAME
•RELAT. NAME
•RELAT. TYPE

The interaction primitives as defined in the SYSTEM 2000 provides the designer with the following facilities:

- entry of the sentence into the database;
- entry of the data and all the related characterstics as provided by the methodology;
- possibility of correcting improperly entered data and checking any eventual previous entry of the data;
- printing of the output documents.

During the writing of the interaction commands, the authors realized that the definition of corresponding S2K "strings" was was in same cases difficult in same cases difficult (see appendix 2). Moreover their usage seemed not to be easy for the end user , while it was much simpler to use the natural language [LSM-SCF] for data entry.

The synthesis and the related semantics for every primitive defined are described in the following.

*FRASE<type> (<code>)<sentence text>*end**

A sentence can be entered. the <type> must be one of the following:

DATI
OPERAZIONI
EVENTO
VINCOLI.

As to the <code>, the existing ones can be checked by using the TALLY command [LSM-SCF]. The <sentence text> is any string of characters.

FRASE?

Prints on the terminal the usage specifications for the previous command.

ESISTE DATO<datum name>

Checks the existence in the DB of the datum whose name is specified.

To record a datum, type

INSERISCI DATO (<datum name>,<version>)

where <version> is a decimal which allows the maintenance of the history of updating and the <data name> is the name of the datum to be entered; the date of the operation is automatically registered.

Subsequently the relative characteristics are entered.

Two commands are provided for each characteristic, one to store the new characteristic (the command ends with the letter A) and the other to change the characteristic which already exists (the command ends with the letter C). The synthesis is as follows:

```
*TIPOA(<data type>)*
*TIPOC(<data type>)*

*DESCRIZIONEA(<description>)*
*DESCRIZIONEC(<description>)*

*DOMINIOA**<domain>*end*wh same
*DOMINIOC**<domain>*end*wh same

*ISTANZEA(<instance>)*
*ISTANZEC(<instance>)*

*FRASEDATIA(<data sentence>)*
*FRASEDATIC(<data sentence>)*

*FRASEOPERA(<operation sentence>)*
*FRASEOPERC(<operation sentence>)*

*SINONIMIA(<data synonym>)*
*SINONIMIC(<data synonym>)*

*DATORELA(<related data>)*
*DATORELC(<related data>)*

*RELNOMEA(<relation name>)*
*RELNOMEC(<relation name>)*

*RELTIPOA(<relation type>)*
*RELTIPOC(<relation type>)*

*VARIEA(<string>)*
*VARIEC(<string>)*
```

where:

<data type>:	:E (entity), AT (attribute);
<description>	:a string of characters describing the datum;
<domain>	:is the equivalent of PICTURE COBOL;
<instance>	:the instance of the datum chosen among the significant instances;
<data sentence>	:code of the data sentence describing the datum;

<operation sentence>:code of the operations sentence describing the datum;

<related data> :name of relation range;

<relation name> :name of relationship (in some cases may take special values as: IS ATT OF, or IS SUBSET OF, etc.);

<relation type> :cardinality of the relationship;

<synonym> :name which may be used in substitution to that indicated at the beginning;

<string> :string of characters for particular annotations.

To check exact entry, type:

controllo<data name>.

The commands for output printing are given in the following:

LISTA FRD
LISTA FRO
LISTA FRE
LISTA FRV

which prints respectively the data sentence form, the operation sentence form, the events sentence form and the constraints sentence form.

GLOSSDATI
GLOSSOPER
GLOSSCORR

prints respectively the data glossary, the operations glossary and the relationship glossary.

LISTA COMPLETA

prints all the output documents.

Conclusion and future perspectives

At the first stage of the implementation, even the possibility of using a (modified) version of XDD was considered. The idea was abandoned for the fact that the XDD structure seemed to be greatly oriented toward the physical design stage and suitable for documentation and maintenance, while the extension toward the User Requiremnts Collection phase seemed to be in contrast with its basic structure. Another problem was constituted by the limitation of the total number of components a user can define in a SYSTEM 2000 database, as the total number of XDD components was very near to the maximum allowed of 1000 components.

During the implementation of the logical schema, we encountered the well known problems of implementing N:M relationships in hierarchical DBMSs. The only way out seems to be the implementation of ad-hoc software which makes use of the LINK facility offered by SYSTEM·2000. It should be noted that also QUEX is suitable only for retrieval, and not for insertion and updating, as no user exits are provided.

For further developments, it seems to be particularly useful if an adequate graphical interface is available, at least for displaying the conceptual schema and event graphs. Obviously, a graphic interaction would even be better.

The developments envisaged are, besides the revieving of some particular items, the implementation of PLEX procedures for the automatic entry of schema items which implement the N:M relationship and the consistence control.

Other future developments are the designing of an automated tool for logical and physical design of SYSTEM 2000 which will probably make use of a workload simulator to evaluate the global performances of resulting structures.

References

[CERI83] S. Ceri, editor: Methodology and tools for database design
North Holland

[DeAD83] V. De Antonellis, B. Demo: Requirements Collection and Analysis - in [CERI83]

[BuCe83] U. Bussolati, S. Ceri, V. De Antonellis, B. Zonta: Views conceptual design, in [CERI83]

[CHEN76] Chen P.P., The entity relationship model: toward an unified view of data, ACM TODS, 1(1) (1976) 9-36.

[DATAID] Metodologia per la progettazione manuale di basi di dati - CNR - PFI - Sottoprogetto P2, Obiettivo DATAID

[LSM-COB] SYSTEM 2000-The Language Specification Manual For The COBOL Programming Language Extension (PLEX) For IBM OS/VS.-1979 MRI System Corporation.

[LSM-SCF] SYSTEM 2000-SYNTAX GUIDE-Self Contained Facility For IBM OS/VS.-1979 MRI System Corporation.

[PETRI80] C.A.PETRI-Introdution To General Net Theory,in Lecture Notes in Computer Sciences, n.84, Springer-Verlag, 1980.

[XDD] SYSTEM 2000-Reference Manual for the Extended Data Dictionary for IBM OS/VS - Release 10 - 1980 Intel System Corporation.

[OSVT82] Olle, T.W., Sol, H.G., Verrijn-Stuart, A.A. (eds), Information Systems Design Methodologies: A Comparative Review, North-Holland, Amsterdam, 1982.

Appendix 1: The Database Definition

```
SYSTEM RELEASE NUMBER   10.1
DATA BASE NAME IS       ASTU
DEFINITION NUMBER               2
DATA BASE CYCLE NUMBER          1
     1*  TIPO (CHAR XXX)
     2*  ATTIVITA' (CHAR X(25))
     3*  SETTORE (CHAR X(25))
     4*  UTENTE (CHAR X(25))
    10*  VERSION (RECORD)
      11*  VERSIONE (DECIMAL NUMBER 99.99 IN 10)
      12*  CREATO IL (DATE IN 10)
     100*  FRE (RECORD IN 10)
       101*  FRE CODICE (CHAR X(7) IN 100)
       102*  FRE TESTO (NON-KEY CHAR X(250) IN 100)
       103*  FGE TIPO SINTATTICO CONDIZIONI (CHAR X(25) IN 100)
       104*  FGE TIPO SINTATTICO OPERAZIONI (CHAR X(25) IN 100)
       120*  EVN CONDIZIONE (RECORD IN 100)
         121*  EVN CONDIZIONE NUMERO (NON-KEY INTEGER NUMBER 99 IN 12

         122*  EVN CONDIZIONE NOME (CHAR X(20) IN 120)
       140*  OPR CAUSATA (RECORD IN 100)
         141*  OPR CAUSATA NUMERO (NON-KEY INTEGER NUMBER 99 IN 140)
         142*  OPR CAUSATA NOME (CHAR X(20) IN 140)
    200*  OPR (RECORD IN 10)
      201*  OPR NOME (CHAR X(25) IN 200)
      202*  OPR DESCRIZIONE (NON-KEY CHAR X(250) IN 200)
      203*  OPR TIPO PRAGMATICO (CHAR XXX IN 200)
      204*  OPR TIPO (CHAR X IN 200)
      205*  OPR ESECUZIONE (CHAR X(6) IN 200)
      206*  OPR FREQUENZA (CHAR X(15) IN 200)
      207*  OPR VARIE (NON-KEY CHAR X(250) IN 200)
      220*  DATI COINVOLTI (RECORD IN 200)
        221*  DATI COINVOLTI NOME (CHAR X(20) IN 220)
        222*  DATI COINVOLTI CARDINALITA MIN (INTEGER NUMBER 9(6) IN
              20)
        223*  DATI COINVOLTI CARDINALITA MAX (INTEGER NUMBER 9(6) IN
              20)
        224*  DATI COINVOLTI CARDINALITA AVG (INTEGER NUMBER 9(6) IN
              20)
        225*  DATI COINVOLTI RUOLO (CHAR X IN 220)
      240*  FRO RELATIVA (RECORD IN 200)
        241*  FRO RELATIVA CODICE (CHAR X(20) IN 240)
      260*  EVN DETERMINATO (RECORD IN 200)
        261*  EVN DETERMINATO NOME (CHAR X(20) IN 260)
      280*  FGE DETERMINANTE (RECORD IN 200)
        281*  FGE DTERMINANTE CODICE (CHAR X(20) IN 280)
   300*  FRD (RECORD IN 10)
     301*  FRD CODICE (CHAR X(7) IN 300)
     302*  FRD TESTO (CHAR X(250) IN 300)
     320*  DATI DESCRITTI (RECORD IN 300)
       321*  DATI DESCRITTI NOME (CHAR X(20) IN 320)
```

```
400*  DAT (RECORD IN 10)
  401*  DAT NOME (CHAR X(25) IN 400)
  402*  DAT TIPO (CHAR X(10) IN 400)
  403*  DAT DESCRIZIONE (NON-KEY CHAR X(250) IN 400)
  404*  DAT PICTURE (CHAR X(10) IN 400)
  408*  DAT SOTTOINSIEME DI (CHAR X(25) IN 400)
  409*  DAT VARIE (NON-KEY CHAR X(250) IN 400)
  410*  DAT ISTANZA (RECORD IN 400)
    411*  ISTANZA (NON-KEY CHAR X(30) IN 410)
  420*  FRD DESCRIVENTE (RECORD IN 400)
    421*  FRD DESCRIVENTE CODICE (CHAR X(20) IN 420)
  440*  FRO AFFECTING (RECORD IN 400)
    441*  FRO AFFECTING CODICE (CHAR X(20) IN 440)
  460*  DAT RELATED (RECORD IN 400)
    461*  DAT RELATED NOME (CHAR X(20) IN 460)
    462*  DAT RELATED NOME RELAZIONE (CHAR X(30) IN 460)
    463*  DAT RELATED TIPO RELAZIONE (CHAR XXX IN 460)
  480*  DAT SINONIMI (RECORD IN 400)
    481*  DAT SINONIMI VALORI (CHAR X(25) IN 480)
500*  EVN (RECORD IN 10)
  501*  EVN NOME (CHAR X(25) IN 500)
  502*  EVN TIPO SEMANTICO (CHAR XXXX IN 500)
  503*  EVN TIPO PRAGMATICO (CHAR XXX IN 500)
  504*  EVN VARIE (NON-KEY CHAR X(250) IN 500)
  505*  EVN DATO COINVOLTO NOME (CHAR X(20) IN 500)
  520*  FRE CONDIZIONE (RECORD IN 500)
    521*  FRE CONDIZIONE CODICE (CHAR X(20) IN 520)
  540*  OPR DETERMINANTE (RECORD IN 500)
    541*  OPR DETERMINANTE NOME (CHAR X(20) IN 540)
600*  FRO (RECORD IN 10)
  601*  FRO CODICE (CHAR X(7) IN 600)
  602*  FRO TESTO (NON-KEY CHAR X(250) IN 600)
  620*  OPR DESCRITTA (RECORD IN 600)
    621*  OPR DESCRITTA NOME (CHAR X(20) IN 620)
700*  FRV (RECORD IN 10)
  701*  FRV CODICE (CHAR X(7) IN 700)
  702*  FRV TESTO (NON-KEY CHAR X(250) IN 700)
```

Appendix 2: The Strings

```
2001*  LISTA COMPLETA (STRING (*LISTA FRD*:*LISTA FRO*:*LISTA FRE*:*LIS
       TA FRV*:*GLOSSDATI*:*GLOSSOPER*:*GLOSSCORR*))
2005*  TITOLO (STRING (LI/TITLE   F(25) ))
2006*  A15 (STRING (---------------))
2007*  A8 (STRING (--------))
2008*  A25 (STRING (-------------------------))
2009*  A5 (STRING (-----))
2010*  A9 (STRING (---------))
2011*  A1 (STRING (----------))
2012*  A2 (STRING (--------------------))
2013*  A3 (STRING (----------------------------------------------------
       ------------))
2020*  RIGA (STRING (LI/TITLE-----------------------------------------
       ----------------------------------/))
2050*  AA1 (STRING (L(10)*A1*+| CODICE |+*A1*,B(0),L(68)*A3*+ FRASI DAT
       I
       ))                                                  TESTO    |+*A3*/
2051*  AA2 (STRING (L(10)*A1*+| CODICE |+*A1*,B(0),L(68)*A3*+ FRASI OPE
       RAZIONI
       ))                                                  TESTO    |+*A3*/
2052*  AA3 (STRING (L(10)*A1*+| CODICE |+*A1*,B(0),L(68)*A3*+ FRASI EVE
       NTO
       ))                                                  TESTO    |+*A3*/
2053*  AA4 (STRING (L(10)*A1*+| CODICE |+*A1*,B(0),L(68)*A3*+ FRASI VIN
       COLI
       ))                                                  TESTO    |+*A3*/
2060*  GDATI (STRING (LI/TITLE|                                G L O S S A R I O
                       D A T I                        |/))
2070*  GOPER (STRING (LI/TITLE|                                G L O S S A R I O
                       O P E R A Z I O N I            |/))
2080*  GEVEN (STRING (LI/TITLE|                                G L O S S A R I O
                       E V E N T I                    |/))
2090*  GCORR (STRING (LI/TITLE|                                G L O S S A R I O
                     C O R R I S P O N D E N Z E      |/))
2100*  LISTA FRD (STRING (*TITOLO*,*AA1* C301,C302,OB C301 WH C301 EXIS
       TS:))
2110*  LISTA FRO (STRING (*TITOLO*,*AA2* C601,C602,OB C601 WH C601 EXIS
       TS:))
2120*  LISTA FRE (STRING (*TITOLO*,*AA3* C101,C102,OB C101 WH C101 EXIS
       TS:))
2130*  LISTA FRV (STRING (*TITOLO*,*AA4* C701,C702,OB C701 WH C701 EXIS
       TS:))
2200*  FRASE DATI (STRING (IT ENTRY*0 EQ 1*FRD* 10*300*301* *1* * 302*)
       )
2210*  FRASE OPERAZIONI (STRING (IT ENTRY*0 EQ 1*FRO* 10*600*601* *1* *
        602*))
2211*  FRASE VINCOLI (STRING (IT ENTRY*0 EQ 1*FRV* 10*700*701* *1* * 70
       2*))
2220*  FRASE EVENTO (STRING (IT ENTRY*0 EQ 1*FRE* 10*100*101* *1* * 102
       *))
2300*  FRASE? (STRING (LI/TITLE(80) PER IMMETTERE UNA FRASE FARE,L(50)*D
```

```
        **O***E*/: ))
 2301*  D (STRING (                          FRASE"TIPO"("CODICE")"TESTO FRASE"*E
        ND*))
 2302*  O (STRING (                          TIPO= DATI OPERAZIONI ))
 2303*  E (STRING (                                EVENTO  VINCOLI ))
 2304*  CONTROLLO (STRING (PR/NAME/ENTRY WH SAME:))
 2305*  DATORELA (STRING (IT C460 *0 EQ 461* *1* * END* WH SAME:))
 2306*  DATORELC (STRING (CH C460 EQ 461* *1* * END* WH SAME:))
 2307*  RELNOMEA (STRING (AD C460 EQ 462* *1* * END* WH SAME:))
 2308*  RELNOMEC (STRING (CH C460 EQ 462* *1* * END* WH SAME:))
 2309*  RELTIPOA (STRING (AD C460 EQ 463* *1* * END* WH SAME:))
 2310*  RELTIPOC (STRING (CH C460 EQ 463* *1* * END* WH SAME:))
 3000*  ESISTE DATO (STRING (LI C401 WH C1 EQ DAT AND C401 EQ *1*:))
 3001*  INSERISCI DATO (STRING (IT ENTRY*0 EQ 1*DAT* 10*11* *2* *12**TOD
        AY* *400*401* *1* *END*:*esiste dato*))
 3002*  TIPOA (STRING (AD DAT EQ 402* *1* * END* WH SAME:))
 3003*  TIPOC (STRING (CH DAT EQ 402* *1* * END* WH SAME:))
 3004*  DESCRIZIONEA (STRING (AD DAT EQ 403* *1* * END* WH SAME:))
 3005*  DESCRIZIONEC (STRING (CH DAT EQ 403* *1* * END* WH SAME:))
 3006*  DOMINIOA (STRING (AD DAT EQ 404))
 3007*  DOMINIOC (STRING (CH DAT EQ 404))
 3008*  ISTANZEA (STRING (IT C410 *0 EQ 411* *1* *END* WH SAME:))
 3009*  ISTANZEC (STRING (CH DAT EQ 411* *1* *END* WH SAME:))
 3010*  SOTTOINSIEMEA (STRING (AD DAT EQ 408* *1* *END* WH SAME:))
 3011*  SOTTOINSIEMEC (STRING (CH DAT EQ 408* *1* *END* WH SAME:))
 3012*  VARIEA (STRING (AD DAT EQ 409* *1* *END* WH SAME:))
 3013*  VARIEC (STRING (CH DAT EQ 409* *1* *END* WH SAME:))
 3014*  FRASE DATIA (STRING (IT C420 *0 EQ 421* *1* *END* WH SAME:))
 3015*  FRASE DATIC (STRING (CH DAT EQ 421* *1* *END* WH SAME:))
 3016*  FRASE OPERA (STRING (IT C440 *0 EQ 441* *1* *END* WH SAME:))
 3017*  FRASE OPERC (STRING (CH DAT EQ 441* *1* *END* WH SAME:))
 3018*  SINONIMIA (STRING (IT C480 *0 EQ 481* *1* *END* WH SAME:))
 3019*  SINONIMIC (STRING (CH DAT EQ 481* *1**END* WH SAME:))
 3020*  NOME (STRING (L(25)*A25*+|NOME                          +*A25*))
 3021*  GLOSSDATI1 (STRING (*TITOLO*,*NOME*,B(0),L(5)*A5*+|TIPO+*A5*,B(0
        ),          L(10)*A1*+|DOMINIO |+*A1*,B(0),L(20)*A2*+ISTANZE
                |+*A2*,B(0),        L(9)*A9*+FRDATI  |+*A9*,B(0),L(9)*A9
        *+FROPER  |+*A9*/*GLOSSDATI1X*))
 3022*  GLOSSDATI1X (STRING (BY C400,C401,C402,C404,C411,C421,C441,OB C4
        01          WH C1 EQ DAT:))
 3023*  GLOSSDATI2 (STRING (*TITOLO*,*NOME*,B(0),L(15)*A15*+|SINONIMI
          |+*A15*,    B(0),L(8)*A8*+VARIE  |+*A8*,B(0),L(20)*A2*+DESCRIZI
        ONE          |+*A2*/BY C400,C401,C481,C409,C403,OB C401 WH C1 EQ D
        AT:))
 3024*  GLOSSDATI (STRING (*RIGA*:*GDATI*:*RIGA*:*GLOSSDATI1*:*GLOSSDATI
        2*))
 3025*  GLOSSOPER1 (STRING (*TITOLO*,*NOME*,B(0),L(8)*A8*+|ESEC. |+*A8*,
        B(0),L(5)*A5*+TIPO|+*A5*,B(0),L(15)*A15*+FREQUENZA      |+*A15*,B
        (0),L(25)*A25*+DESCRIZIONE                  |+*A25*/*GLOSSOPER1X*))
 3026*  GLOSSOPER1X (STRING (BY C200,C201,C205,C204,C206,C202,OB C201 WH
         C1 EQ OPR:))
 3027*  DATICOINVOLTI (STRING (*TITOLO*,*NOME*,B(0),L(10)*A1*+|FRASE OPR
        +*A1*,B(0),   L(25)*A25*+|NOME DATI COINVOLTI     |+*A25*,B(0),L(8
        )*A8*+RUOLO  |+*A8*,  B(0),L(8)*A8*+C. MIN |+*A8*,*AA*,B(0),L(8)*
```

```
          A8*+C. AVG |+*A8*/*DATIX*))
3028*     AA (STRING (B(0),L(8)*A8*+C. MAX |+*A8*))
3029*     DATIX (STRING (BY C200,C201,C241,C221,C225,C222,C223,C224,OB C20
          1 WH C1 EQ OPR:))
3030*     GLOSSOPER (STRING (*RIGA*:*GOPER*:*RIGA*:*GLOSSOPER1*:*DATICOINV
          CLTI*))
3031*     GLOSSCORR (STRING (*RIGA*:*GCORR*:*RIGA*:*GLOSSCORR1*))
3032*     GLOSSCORR1 (STRING (*TITOLO*,B(0),L(25)*A25*+| DOMINIO
                +*A25*,B(0),L(20)*A2*+| NOME                    |+*A2*,B(0),L(25)
          *A25*+ CODOMINIO                  |+*A25*,B(0),L(8)*A8*+ TIPO  |+*A8
          */*CORRX*:))
3033*     CORRX (STRING (BY C400,C401,C462,C461,C463,OB C401 WH C462 EXIST
          S:))
4095*     DEFAULT (STRING (ECHO OFF:DATE FORMAT IS DD/MM/YY:))
```

Appendix 3: Session's example

```
user,astu:dbn is astu:
S2K0104/00- SYSTEM 2000 INITIALIZATION PARAMETERS FOLLOW -
S2K0127/01- WARNING- STAE ERROR TRAPPING NOT ENABLED-
S2K0212/01- SYSTEM 2000 RELEASE 10.1-XX -
  05/25/83  16:59:11  BEGIN SYSTEM 2000 - RELEASE 10.1-XX
---
USER,ASTU:DBN IS ASTU:
 -556- OPENED.....ASTU                    2      1  05/25/1983  16:56:18
---
command file is demo:
COMMAND FILE IS DEMO:
---
ACCESS:
---
*FRASE DATI(FRD0001)AN IFIP WORKING CONFERENCE BRINGS TOGETHER EXPERTS
---
FROM ALL IFIP COUNTRIES.*END*:
 -342-       1 SELECTED RECORD(S) -
---

                          . . .

---
 *LISTA FRD*:

* ----------------------------------------------------------------------------
  | CODICE | FRASI DATI                                                TESTO  |
  ----------------------------------------------------------------------------
***
* FRD0001    AN IFIP WORKING CONFERENCE BRINGS TOGETHER EXPERTS FROM ALL IFIP COU
             NTRIES.
* FRD0002    AN IFIP WORKING CONFERENCE CONCERNS TECHNICAL TOPICS OF SPECIFIC INT
             EREST TO ONE OR MORE IFIP WORKING GROUPS.
* FRD0003    AN IFIP WORKING CONFERENCE IS AN INVITED CONFERENCE.
* FRD0004    WORKING GROUPS (TECHNICAL COMMITTEES) ARE INVOLVED IN A CONFERENCE I
             F THEY ARE WITHIN THE SAME TECHNICAL COMMITTEE OF THE CONFERENCE ORG
             ANIZING GROUP OR IF TOPICS OF SPECIFIC INTEREST TO THEM WILL BE DISC
             USSED.
* FRD0005    IT CAN HAPPEN THAT INVITED PEOPLE DO NOT ATTEND THE CONFERENCE.
* FRD0006    AN IFIP WORKING CONFERENCE HAS A FINANCIAL BREAK-EVEN POINT AND THER
             EFORE A LIMITED PARTICIPATION, I.E. A MAXIMUM NUMBER OF ATTENDEES.
* FRD0007    AN ORGANIZING COMMITTEE IS APPOINTED FOR AN IFIP WORKING CONFERENCE.
---
```

```
*FRASE OPERAZIONI(FRO0001)PREPARE THE LIST OF PEOPLE TO INVITE.*END*:
 -342-       1 SELECTED RECORD(S) -
---

                          . . .

---
*LISTA FRO*:

* ------------------------------------------------------------------------------
  | CODICE | FRASI OPERAZIONI                                            TESTO    |
  ------------------------------------------------------------------------------
***
* FRO0001   PREPARE THE LIST OF PEOPLE TO INVITE.
* FRO0002   ISSUE PRIORITY INVITATIONS TO NATIONAL REPRESENTATIVES, WORKING GROU
            P MEMBERS, MEMBERS OF ASSOCIATED WORKING GROUPS.
* FRO0003   PREPARE INVITATIONS TO ALL AUTHORS OF EACH SELECTED PAPER.
* FRO0004   PREPARE INVITATIONS TO ALL AUTHORS OF EACH REJECTED PAPER.
* FRO0005   REGISTER INVITATIONS AND ACCEPTANCE OF INVITATIONS.
* FRO0006   PREPARE THE FINAL LIST OF ATTENDEES.
---

*FRASE EVENTO(FRE0001)==> PREPARE THE LIST OF PEOPLE TO INVITE.*END*:
 -342-       1 SELECTED RECORD(S) -
---

                          . . .

---
*LISTA FRE*:

* ------------------------------------------------------------------------------
  | CODICE | FRASI EVENTO                                                TESTO    |
  ------------------------------------------------------------------------------
***
* FRE0001   ==> PREPARE THE LIST OF PEOPLE TO INVITE.

* FRE0002   ISSUED PRIORITY INVITATIONS ==> ISSUE PRIORITY INVITATIONS.
* FRE0003   ISSUED PRIORITY INVITATIONS ==> ENSURE THAT ALL AUTHORS OF EACH SELE
            CTED PAPER RACEIVE AN INVITATION.
* FRE0004   ENSURED THAT ALL AUTHORS OF EACH SELECTED PAPER RECEIVE AN INVITATIO
            N ==> ENSURE THAT ALL AUTHORS OF EACH SELECTED PAPER RECEIVE AN INVI
            TATION.
* FRE0005   ENSURED THAT ALL AUTHORS OF EACH REJECTED PAPER RECEIVE AN INVITATIO
            N ==> ACTIVATE INVITING FOR THE WHOLE LIST OF PEOPLE TO BE STILL INV
            ITED.
* FRE0006   COMPLETED INVITING ==> GENERATE FINAL LIST OF ATTENDEES
```

```
---
*INSERISCI DATO(IFIP WORKING CONFERENCE,0.01):
 -342-       1 SELECTED RECORD(S) -
* DAT NOME
***
* IFIP WORKING CONFERENCE
---

                          . . .

---
*GLOSSDATI*:
```

GLOSSARIO				DATI	
NOME	TIPO	DOMINIO	ISTANZE	FRDATI	FROPER
AUTHOR	E				FRO0003
COUNTRY	E	X(20)			FRO0004 FRO0002
			ITALY		
EXPERT				FRD0001	
FINANCIAL BREAK-EVEN POINT	A	9(6)		FRD0001	
					FRO0015
				FRD0006	
IFIP WORKING CONFERENCE			$50000		
IFIP WORKING GROUP				FRD0001 FRD0002 FRD0004	
			IFIP WG 8.1		
INVITED CONFERENCE	E				
INVITED PEOPLE	E			FRD0003	
LIMITED PARTICIPATION	A	9(3)		FRD0005	
					FRO0015
				FRD0006	
ORGANIZING COMMITTEE	E		200		
TECHNICAL COMMITTEE				FRD0007	
					FRO0007
				FRD0004	
TOPIC			TG.8		
				FRD0002	

NOME	SINONIMI	VARIE	DESCRIZIONE
***			
* AUTHOR			AUTHOR OF CONTRIBUTED PAPER
* COUNTRY			WHERE EXPERT LIVE OR CONFERENCE IS HELD
* EXPERT			SOMEONE HAVING SPECIFIC KNWOLEDGE OF SOME TOPIC
* FINANCIAL BREAK-EVEN POINT			THE AMOUNT REQUIRED TO COVER ORGANIZ. COSTS
* IFIP WORKING CONFERENCE	INVITED CONFERENCE		INTERN. MEETING TO DISCUSS TECNICAL TOPICS
* IFIP WORKING GROUP			BRINGS TOGETHER EXPERTS
* INVITED CONFERENCE	IFIP WORKING CONFERENCE		CONFERENCE NOT OPEN TO EVERYONE
* INVITED PEOPLE			PEOPLE TO INVITE TO AN INVITED CONFERENCE
* LIMITED PARTICIPATION			MAXIMUM NUMBER OF PEOPLE ADMITTED
* ORGANIZING COMMITTEE			HANDLES FINANCIAL METTERS
* TECHNICAL COMMITTEE			
* TOPIC			TECNICAL SUBJECT EXPERTS ARE WORKING ON

GLOSSOPER:

GLOSSARIO				OPERAZIONI
NOME	ESEC.	TIPO	FREQUENZA	DESCRIZIONE
ISSUE REJECTED PAPERS INVITATIONS	OL	R	2	LIST AUTHORS OF REJECTED PAPERS FOR INVITATION
ISSUE SELECTED PAPERS INVITATIONS	OL	R	2	LIST AUTHORS OF SELECTED PAPERS FOR INVITATION
ISSUE TC PRIORITY INVITATION	OL	R	2	LIST INVITATION TO NATION. REPRESENTATIVES AND INVOLVED T.C. MEMBERS
ISSUE W.G. PRIORITY INVITATION	OL	I	2	LIST INVITATION TO ASSOCIATED WORKING GROUP MEMBERS
SEND INVITATIONS	OL	I	200	

NOME	FRASE OPR	NOME DATI COINVOLTI	RUOLO	C. MIN	C. MAX	C. AVG
ISSUE REJECTED PAPERS INVITATIONS		REJECTED PAPERS	I			
		AUTHORS	O			
	FRO0004					
ISSUE SELECTED PAPERS INVITATIONS		SELECTED PAPERS	I			
		AUTHORS	O			
	FRO0003					
ISSUE TC PRIORITY INVITATION		IFIP WORKING CONFERENCE	I			
		COUNTRY	V			
		IFIP WORKING GROUP	V			
		TECHN. COMM.	V			
	FRO0002					
ISSUE W.G. PRIORITY INVITATION.		TECHN. COMM.	I			
		IFIP WORKING GROUP	V			
		MEMBER PEOPLE	O			
	FRO0002					
SEND INVITATIONS		INVITED PEOPLE	I			
		IFIP WORK. CONF.	I			
	FRO0005					

GLOSSCORR:

G L O S S A R I O		C O R R I S P O N D E N Z E	
DOMINIO	NOME	CODOMINIO	TIPO
***			
* FINANCIAL BREAK-EVEN POINT	IS ATT OF	IFIP WORKING CONFERENCE	1-1
* IFIP WORKING CONFERENCE	IS SUBSET OF	INVITED CONFERENCE	1-1
*	HAS ATT	FIN. BREAK-EVEN POINT	1-1
*	HAS ATT	LIMITED PARTICIPATION	1-1
* INVITED CONFERENCE	IS SUPERSET OF	IFIP WORKING CONFERENCE	1-1

---

exit:
EXIT:
17:02:04 05/25/83 END SYSTEM 2000 - RELEASE 10.1-XX